MACHINE LEARNING MODEL DEPLOYMENT IN CLOUD

Project report submitted in partial fulfilment of the requirement for the degree of Bachelor of Technology

In

Computer Science and Engineering

By

Shivek Gupta (181323)

Vardhan Chambial (181425)

UNDER THE SUPERVISION OF

Dr. Rajni Mohana

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Waknaghat,

173234, Himachal Pradesh

CERTIFICATE

Candidate's Declaration

I hereby declare that the work presented in this report entitled **"Machine Learning Model Deployment in Cloud"** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August 2021 to December 2021 under the supervision of **Dr. Rajni Mohana,** Associate Professor Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat.

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Shivek Gupta (181323) Vardhan Chambial (181425)

This is to certify that the above statement made by the candidate is true to the best of my knowledge

Dr. Rajni Mohana
Associate Professor
Computer Science Engineering and Information Technology
Jaypee University of Information Technology, Waknaghat,

AKCNOWLEDGEMENT

Firstly, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible to complete the project work successfully.

I am really grateful and wish my profound indebtedness to Supervisor **Dr. Rajni Mohana,** Associate Professor of CSE Jaypee University of Information Technology,Waknaghat. Deep Knowledge & keen interest of my supervisor in the field of **Machine Learning and NLP** to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

I would like to express my heartiest gratitude to **Dr. Rajni Mohana,** Department of CSE, for his kind help to finish my project.

I would also generously welcome each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

Shivek Gupta
Vardhan Chambial

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

- **EDA:** Exploratory Data Analysis
- **ML:** Machine Learning
- **i.e:** that is
- **RAM:** Random Access Memory
- **CPU:** Central Processing Unit
- **GPU:** Graphics Processing Unit
- **SVM:** Support Vector Machines
- **NLTK:** Natural Language Toolkit (NLTK)
- **MNB:** MultinomialNB
- **SVC:** Support vector classifier
- **POS:** Parts of Speech

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The web can be a huge visual aid when it can be accurate and shared testing, certain health influences, which have an impact on advertising and voice communication alike. A social media unit that influences clients' opportunities on how to shape their attitudes and behavior. Chasing communications services can be a great way to live the loyalty of customers, to restore their sense of perspective about products or products. A social network is the next logical advertising space.

Web Portals receive a large amount of feedback from users. Getting all the answers can be a daunting task. You should separate the ideas expressed in the response forums. This can be used for the feedback management system. We classify individual comments / updates and determine the overall rating based on individual comments / updates. So that company can get a complete view of the feedback given to customers and can take care of those specific areas. This makes Customers more loyal to the company, business growth, reputation, product value, profitability.

An independent technology is used in language that understands the meaning of the text. Indicates the opinion or attitude that a person has toward the subject or object. Emotional analysis is widely used in survey reviews and responses, in online and social media, and in-app therapy products ranging from marketing to customer services to medical treatment.

With the help of Sentimental Analysis, we often help an organization better understand their customer feedback so that they can focus on customer problems.

Chapter 1 INTRODUCTION

1.1 Introduction

Machine learning is the part of artificial intelligence that provides a computer the ability to automatically learn with experience without the need of external programming. It is one of the hot ,most in demand research topics in computer science engineering. It can provide intelligence to the machines with the help of various tools and techniques. ML uses programs that can access data and learn for themselves using the same. In today's world machine learning has become an important part of everything we are a part of. It helps various enterprises for the development of new products and also helps to have an idea about the customer trends. Many MNCs have made ML an important part of their organisation.

The learning process for an ML algorithm begins and ends with data. Therefore, data is the most important part of a model. This data has an effect on future predictions and decisions. The primary objective of ML is that a computer system learns automatically without human assistance.

Machine Learning is categorized mainly into 4 approaches based on how an algorithm goes through the process of learning.

- **Supervised machine learning**: Analysis of labelled data and then training on the same to predict future events. This system can provide output for new inputs if sufficient training is done.
- **Unsupervised machine learning:** The information used in training is not labelled and neither classified. This system is used to study different patterns and connections in data.
- **Semi-supervised machine learning**: Falls in between the above two as it uses small amounts of labelled data and larger amounts of the unlabelled data.
- **Reinforcement machine learning**: In this system the program learns to behave in a particular environment by performing actions and then seeing the results.

As online markets have become popular in recent decades, online retailers and retailers are inviting their customers to share their thoughts on the products they have purchased. Every day millions of updates are made across the Internet about different products, services and locations. This has made the Internet a valuable resource for getting ideas and opinions about a particular product or service. However, as the number of reviews available for a product grows, it becomes more and more difficult for a potential consumer to make a good decision as to whether to buy the product. Different opinions about the same product on the one hand and incomprehensible reviews on the other hand make customers more confused

to get the right decision. Here the need to analyze this content seems to be relevant to all commerce businesses.

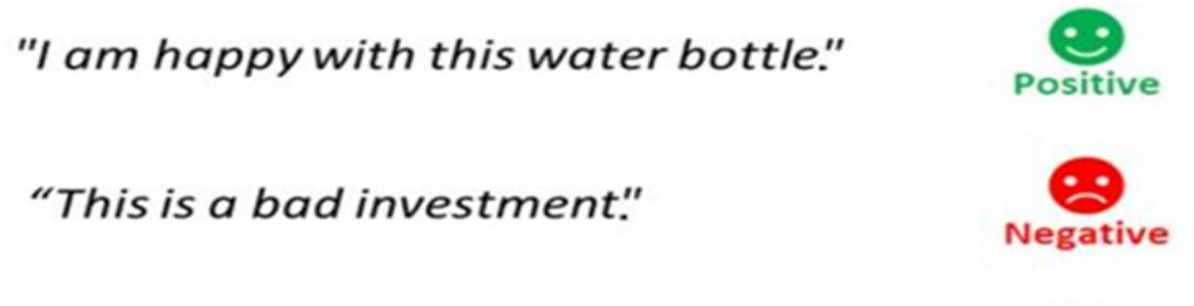


Figure 1.1: Sentiment

Sentiment Analysis is a part of Natural Language Processing which uses ML algorithms to identify the emotional tone behind an expression. This is used by organizations to determine the opinions about a product or service. Most Sentiment analysis engines can easily predict the sentiments behind a simple expression but struggle to accurately predict emotions behind a complex one, such as “This is a very good product but it will not be very useful.” or “It is quite sunny outside, let me fetch an umbrella” etc.

Emotional analysis and classification is a computer study that attempts to solve this problem by extracting automated data from texts provided in the natural language, such as ideas and emotions. Various methods have been used to address this problem from natural language analysis, text analysis, computer language learning, and biometric. In recent years, electronic learning methods have become increasingly popular in semantic analysis and reviews of their simplicity and accuracy. We use sentiment analysis for a given expression so that our project is capable of accurately predicting the emotions behind simple sentences such as “It is such a lovely day today”, “You look amazing” and hopefully predict the sentiments behind complex ones with a good accuracy.

1.2 Problem Statement

In today’s world there are thousands of people shopping online and reviewing products so that someone else can benefit from it and not end up buying the wrong product. Due to this there are positive as well as negative reviews for the same product based on the user experience. We can classify the reviews based on positive feedback or negative feedback. But some comments have both mixed words positive

as well as negative for eg. " I didn't like the product", this statement consists of both words and this is where negation handling comes into play. With the technological advancements and the load increase in online shopping and reviews, sentiment analysis without negation handling does not give the desired results and offers low accuracy. Customers on the other hand do not have a tool to have an idea about the reviews that could help them in planning accordingly. Also many customers totally rely on the reviews provided by the various buyers having no or little knowledge of the product. Sentiment analysis can mix this comment into negative as well as positive reviews so to solve this we use negation handling which helps the systems accuracy to be improvised in sorting the various reviews.

So, a machine learning model would be used for the same problem to get the reviews of the products sorted on the basis of negative or positive. This was a brief into the problem which will be discussed below in the report and a possible solution to this problem will be provided.

1.3 Objectives

Every day we come across a variety of products in our lives, digitally we swipe hundreds of product selection items under one category. It will be boring for a customer to choose something. Here comes the 'updates' when customers find the product leaving a rating after using it and summarizing their information by providing updates. As we know the ratings can be easily sorted and judged on whether the product is good or bad. But when it comes to sentence review we need to read the whole line to make sure the review conveys a good or bad idea.

With the advent of artificial intelligence, things like that are becoming easier with Natural Language Processing (NLP) technology. And the fact that we can measure, predict and empower learning tools alone is powerful and unlimited in terms of application opportunities. Both consumers and manufacturers highly value the "customer ideas" about products and service. The accuracy offered by the model should be high so that it could be trusted and anyone doesn't end up buying a wrong product.

The main purpose of our project is to classify individual comments or updates and determine the overall rating based on individual them. Build the model which has highest accuracy in classifying the feedback as Positive, Negative. Once the model classification is done, then we deploy our Model in Cloud Platform. This will help the company to get a complete view of the feedback given to customers and can take care of those specific areas.

1.4 Methodology

The main methodology of this project revolves around using sentimental analysis along with negation handling for classification of the reviews. First and foremost we begin our project with the dataset from kaggle collecting the required data and applying the exploratory data analysis. This is our initial investigation on data to look for any patterns or to check if there is something wrong with it with the help of statistics and graphical ways. After this we move on to data preprocessing. Data preprocessing is the step where data is changed or encoded so that the machine could easily understand it. It also involves the process of dealing with missing, duplicate or inconsistent data. We have used the SentiWordnet of NLTK module to label our pre-processed dataset as positive or negative. Then by using various ML models we check which model gives us the results with highest accuracy.

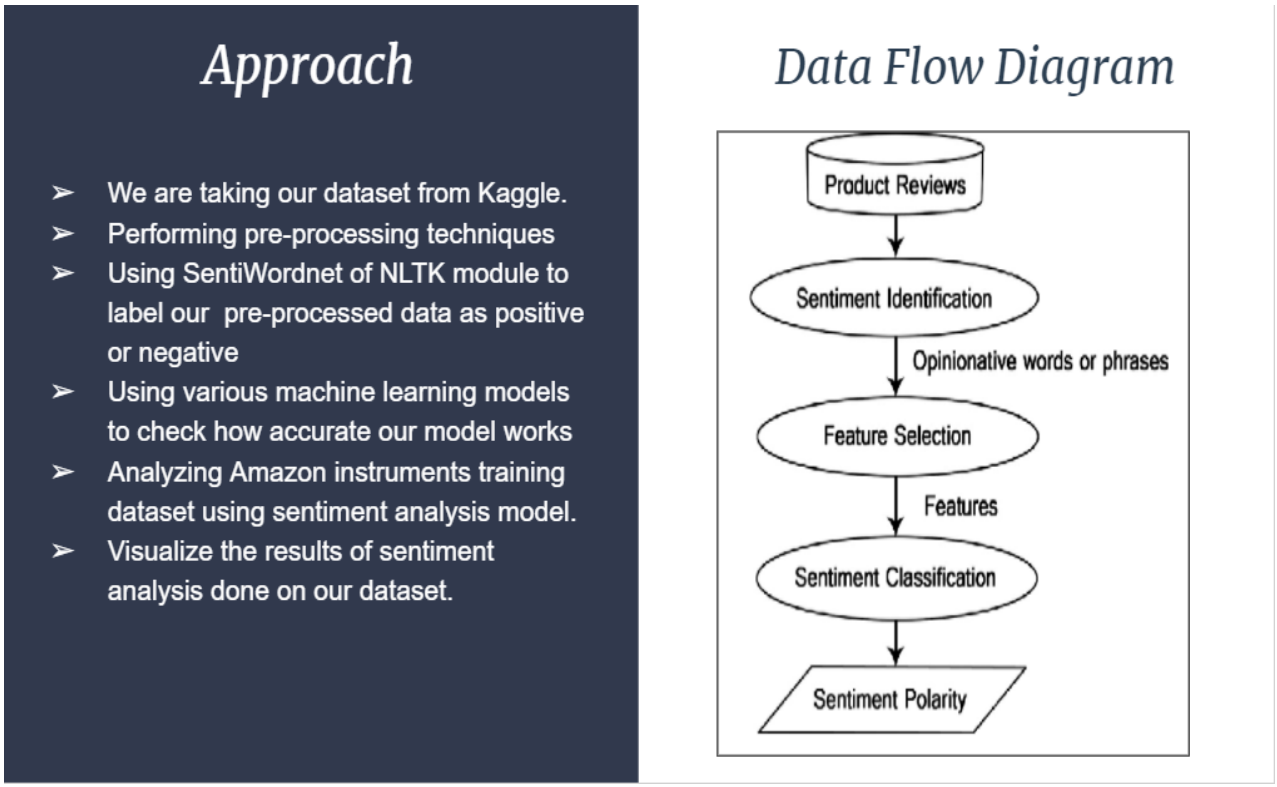


Figure 1.2: Data Flow Approach

As we can see in the diagram, the flow chart shows the steps that are used to proceed in the project.

First we select the reviews, then we identify the review on the basis of positive and negative with the help of sentiment analysis. Once this is done the reviews that consist of both positive and negative words are sorted by negation handling thus, helping us to get the positive and negative reviews more clearly. This completes the aim of our project.

1.5 Organization

In this part we will explain the organization of the chapter wise formatting of this whole report.

In chapter one we have already seen the Introduction, Problem Statement, objectives and the methodology of the report. Now we are in the Organization section that will explain the chapters and topics involved further down in this report.

Chapter 2 consists of the literature survey that gives us the basic overview of the project, the languages used, some algorithms used and also explains the process briefly. This will also include some of the resources we have used in the research to gain the necessary knowledge for this project and all the other work that we came across through the research that helped us in moving forward to reach at this stage.

In Chapter 3 we will be discussing how the model is designed and developed. First we will discuss the dataset that we used in making this report and see some basic features.We will discuss about the basic framework which help us lead to random forest algorithm that include decision tree and feature selection. We would take a look at the various models we applied to draw comparisons with the random forest and their respective accuracies. We have explained them a bit to give the reader a better understanding of them. After that we have also listed some of the mathematical formulas used in this report.

In chapter 4 we have shown how our system performed and also compared them with the other systems used earlier like linear regression ,decision tree etc. Then we begin to showcase the outputs at various stages of our project work that include various graphs and figures. This would conclude with the final output of our model.

Finally, in chapter 5 we started the conclusion part and explained about what we managed to achieve in the project and how this model was the perfect way for us to learn and explore in this field. After that we proposed some future plans that can be achieved through this project and made an extension to it.

Chapter 2 LITERATURE SURVEY

Sentiment Analysis of Reviews using Machine Learning (2021)

Preprocessing:

Start by checking for null values, duplicate rows, and removing unnecessary information. It also removes all the patient conditions that have no meaning at all. The data is cleaned for further analysis.

Methodology:

Start by **cleaning the data** and performing the **data exploration** process using **visualization**. There was a scope of **feature engineering** and **feature extraction**. Once the data is ready for further analysis, perform a **test train split**. Use **SMOTE** for balancing the data. Further using **classifiers** and check the **performance metrics** and then accordingly create a recommendation system.

Conclusion:

In this work, each review was classified as positive or negative, depending on the user's star rating. Ratings above five are classified as positive, while negative ratings are from one to five-star ratings.

Sentiment Analysis of User-Generated Content on Drug Review Websites(2015)

Preprocessing:

The grammatical relationship of words in a clause was processed using the Stanford NLP library.

Methodology:

Sentiment Lexicons were created to collect the negative and positive phrases from the data and then the dependency tree was applied to the data for finding the grammatical relationship between the words i.e. if it is the governor or the dependent. SVM and Linguistic Approach was also used to check the accuracy.

Conclusion:

The accuracy of the model came out to be 62% according to the first SVM and 66% according to the second SVM but the highest accuracy that we came across came from the Linguistic Approach i.e. 69%.

Having various reviews in any product makes the buying decision easy and fast. But when the review count goes up and the difference between the positive and negative comments is not much then, the decision becomes a bit difficult. In order to make things easier, Artificial Intelligence technique called Machine Learning provides a suitable and convenient platform to the users for selecting the right type of information by classifying the views in positive or negative side.

Various Machine Learning algorithms have been used to predict the reviews as sometime there are both positive as well as negative words in the comments. A number of datasets have also been used for the same although the main essence of prediction is the same for all which deals with proper preprocessing and feature selection. Various algorithms like MultinomialMB, Decision tree, Logistic Regression and Support vector classifier (SVC) are commonly used for building this model. Among all the algorithms mentioned above, MultinomialMB gives the best accuracy.

Here is a figure given below giving the Algorithms used and the accuracy and precision of the algorithms used.

	MLA Name	MLA Test Accuracy	MLA Precision
1	MultinomialNB	0.9226	0.877540
2	SVC	0.9071	0.860232
3	DecisionTreeClassifier	0.8889	0.832995
0	LogisticRegression	0.7480	0.687474

Figure 2.1: Accuracy Table

Python:

Python is deciphered, structured, highly developed by dynamic systems semantics. Its elevated level of certain data structures, combined with an exciting build and sure, get involved in Fast Performance Improvement, similarly related use as writing or pasting language to link existing parts together. Python

is clear, simple learning the structure of language emphasizes undeniable and appropriate quality reduces program funding costs. Supports parts and packages, viz renews system protection and code reuse

Machine Learning(ML) :

Machine learning enables PCs to learn without being drawn to the obvious (Arthur Samuel, 1959) This is an area under programmatic structure. Machine learning in all aspects of scale development can read and create opportunities in data. Such controls adhere to altered rules, but they can also choose data predictions or decisions based on data. Build models from test inputs. Machine learning is done when it is unthinkable to masterminding and programming computations. Models include spam filtering.

Natural Language Processing(NLP) :

Sentiment analysis is a process of NLP used to classify confidential information in a text or human language. The purpose of emotional analysis is to classify the feelings of public opinion by classifying them as constructive, neutral, and negative. And Python is often used in NLP activities as an emotional analysis because there is a large collection of NLP tools and libraries to choose from. There are many things about Python that make it an excellent choice for an NLP project plan. Simple syntax and clear semantics make it an excellent choice for projects that include Natural Language Processing tasks.

Natural Language Toolkit (NLTK) :

NLTK is an important library that supports functions such as segmentation, conquest, marking, segmentation, semantic consultation, and Python tokens. Basically your main tool for processing natural language and machine learning. Today we serve as the educational foundation for Python developers who are putting their toes in the field. This library is excellent, but we must admit that it is very difficult to use it for processing Indigenous Languages with Python.

MultinomialNB:

MultinomialNB stands for Multinomial Naive Bayes algorithm is a learning method that is basically used in NLP. This algorithm is based on the bayes theorem that is easily scalable and it can easily handle big datasets. Some of the applications that MultinomialNB can be used for are Face Recognition, Spam Detection, Language Identification as well as Sentiment Analysis.

Flask:

Flask is a framework in python that helps in the web development application easily. Flask is basically a collection of libraries and modules that creates user friendly experience and also is easy to use that makes it a good choice for anyone. It is a very small and light framework used in python and it gives the user very flexible and vast usage of easy coding that can be used in a single file for creating a web application.

Chapter 3 SYSTEM DEVELOPMENT

3.1 Design and Algorithm

The main algorithm used in this report is Naive Bayes. But before explaining that we would explain little bits about the other stages of our model too, i.e exploratory data analysis , data preprocessing and feature selection. Also we would try to customize the data we want, a simple code was written in python to remove unnecessary features. Many features have been removed without all, rtexts, centi_score, emotions. Scores generated by the reviewer include multiple scores on a scale of 1 to 5. Revisions rated by one, two or three are considered negative and those with four or five numbers are considered negative. Three-star reviews usually have a lot of reviews included and it's hard to label them in the positive or negative category.

In this study, two activities were performed.In the first task all data was used. Since the number of updates was sufficient to obtain reasonable results from the classifiers, the three-star review was not issued to avoid any problem while training the algorithms.

However, in the second work due to the small amount of data the three-star review was also considered negative. The same code was used to label the data. The revised rating received a score of “1” and the rest received a score of “0”.

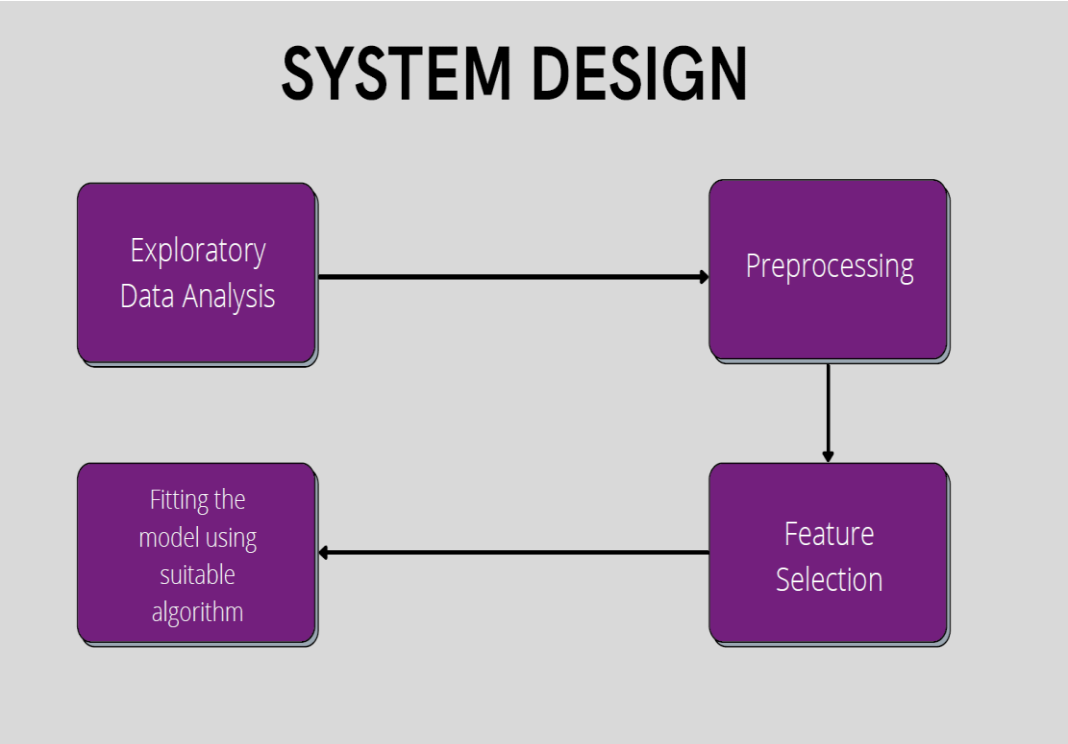


Figure 3.1: System Design

3.1.1 Exploratory Data Analysis

Exploratory data analysis is the process of critical inspection of data so that we can have an insight on the patterns of the dataset, to check for any anomalies with the help of graphs and statistical analysis. It is a very great step to understand the dataset and gather as much information from it. It helps to determine how to manipulate data to get the answers we need. EDA was originally developed by the mathematician John Turkey in the 1970's.

Some tools and techniques that can be used for the same are :

SENTIMENT SENTENCE EXTRACTION & POS TAGGING:

Tokenization of reviews after removal of STOP words which mean nothing related to sentiment is the basic requirement for POS tagging. After proper removal of STOP words like "am, is, are, the, but" and so on the remaining sentences are converted into tokens. These tokens take part in POS tagging.

In natural language processing, part-of-speech (POS) taggers have been developed to classify words based on their parts of speech. For sentiment analysis, a POS tagger is very useful because of the following two reasons:

➢Words like nouns and pronouns usually do not contain any sentiment. It is able to filter out such words with the help of a POS tagger.

➢A POS tagger can also be used to distinguish words that can be used in different parts of speech.

LEMMATIZATION

He aims to reduce the name to its inferior structure and to combine different forms of the same name. For example, verb words in the past are changed to present (for example "went" changed to "go") and the same is combined (for example "best" is changed to "acceptable"), making common words have the same significance in their root. Apart from the fact that it seems to be firmly entrenched with the prevention process, ice formation uses another method of coping with the arrival of word types. Lemmatization places words in a word reference structure (known as lemma) that requires clues of the words used where the calculation can search and link words by comparing lemmas.

We can use different functions available in the python pandas library to begin our EDA.

The ".head()" and the ".tail()" functions show the first five and the last five rows from the dataset respectively. It is a good practice to do so as it gives an idea about the pattern of data.

We can also know about the total of rows and columns in any dataset using the ".shape" function in pandas. We can also use the ``.info()" function which helps to know the columns and their data types, and also helps to find out whether any column contains null values or not. The "describe()" is like the most handy one among all these. It returns various information about the data like count, mean, minimum and maximum values ,standard deviation etc.

```
[ ]  data.info()

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 10261 entries, 0 to 10260
     Data columns (total 9 columns):
      #   Column          Non-Null Count  Dtype
     ---  ------          --------------  -----
      0   reviewerID      10261 non-null  object
      1   asin            10261 non-null  object
      2   reviewerName    10234 non-null  object
      3   helpful         10261 non-null  object
      4   reviewText      10254 non-null  object
      5   overall         10261 non-null  float64
      6   summary         10261 non-null  object
      7   unixReviewTime  10261 non-null  int64
      8   reviewTime      10261 non-null  object
     dtypes: float64(1), int64(1), object(7)
     memory usage: 721.6+ KB
```

Figure 3.2: Example of Info function

Data visualization also comes under the EDA as it helps the user to get a graphical insight into the data. This is basically divided into three categories:

- Univariate analysis: This displays all the observations in data around a single data variable. Ex: line plot, histogram etc
- Bivariate analysis: Reveals relationship between two data variables. Ex: heat maps, box plots etc
- Multivariate analysis: reveals relationship between more than two sets of variables. Ex: violin plots, box plots, histograms etc.

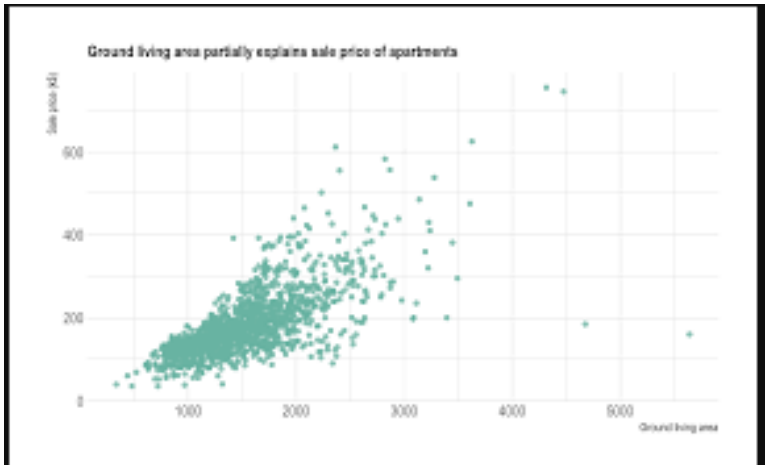


Figure 3.3: Scatter plot

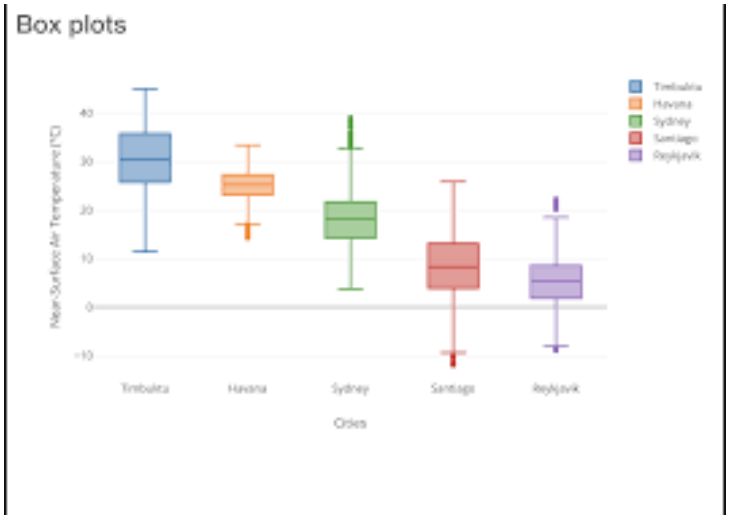


Figure 3.4: Box plots

3.1.2 Data Preprocessing:

Whenever we think of datasets , a large number of rows and columns pop into our heads. But this is not always the case because data can be in structured ,unstructured, tables, images, audio files, videos etc. Now as we all know our computers do not understand anything other than the language of 1's and 0's. So it will utterly make no sense to pass a set of videos or images to our model. Also a dataset may contain missing values, duplicates that are of no use to the model or can cause errors. So here comes the part of preprocessing. EDA and preprocessing are closely related terms and sometimes used in the same way.

In any ML procedure, data preprocessing is the step where our data is transformed or in complex terms encoded ,to bring it to a state so that the computer program can understand it. The columns or the attributes of the data can be consumed or understood by the algorithm after preprocessing. Also it helps in cleaning the data i.e getting rid of the various null or missing values and it helps in increasing the accuracy and efficiency of the model.

- **Tokenization**: The process by which the running text that is meaningful is segmented into terms and phrases i.e. random string of characters called tokens. The key task is to divide a text into tokens while throwing off other characters like dots.
- **Stop word**: This approach filters out and omits some very common words that seem to offer little to no meaning to the NLP target. This is done due to the fact that these words exist in so much abundance that eventually end up offering no unique information that might be used by our model in classification or clustering. It excludes the generic terms which are not insightful about the relevant material.
- **Handling Negative Adjectives**: While eliminating stop words it is possible that we might eliminate words that will wipe out the relevant details and change the meaning of our reviews. So the aim is to avoid the elimination of these words in a given sentence that might change the meaning. Under these conditions, we can pick a minimum to stop word lists or tests of this form (not good / not so good, not bad / not so bad) in our data under certain conditions and delete them from stop words.
- **Stemming**: Refers to the slicing process of the end or beginning of words to remove affixes (lexical additions to the word's root). It basically aims at bringing the words to their base form. It increases the retrieval accuracy along with reducing the size of the index.
- **Lemmatization**: It is the process of bringing the words to their base form properly by using vocabulary analysis of words. It also groups the various forms of the same words together. The base form returned is known as the lemma.

This whole process can be divided into the following sub categories:

- **Getting the dataset:** The collection of data for a particular use is called the dataset. As a machine learning model completely depends on data so we have to make sure we acquire the best possible data out there.
- **Importing libraries**: Some python libraries that we need in our project for some specific jobs have to be integrated. This is called importing libraries. Mainly there is use of three python libraries

  Numpy: Used for mathematical calculations in the code.

  Pandas: Most common and useful library. Used for data manipulation and analysis of dataset.

  Matplotlib: Used for plotting various charts using python. Works with the help of a sub library pyplot.

- **Importing datasets:** The step where we import required datasets for the model

- **Handling missing/inconsistent/duplicate data:** It is a common thing to have missing values in a dataset. The cause to it can be anything ,whether it was by default in the dataset or happened during data collection. But it is our responsibility to handle these missing values.

  1. Eliminate the rows having missing data: It is a simple and effective strategy. This could also fail if many objects have missing values. Sometimes the feature has to be eliminated if its has mostly missing values.
  2. Estimation of the missing values: If only some values are missing then various interpolation methods can be used to fill up these missing values. But in general dealing with these missing values is done by filling them with median ,mode or mean from the respective column.

Figure 3.5: Illustration of missing values

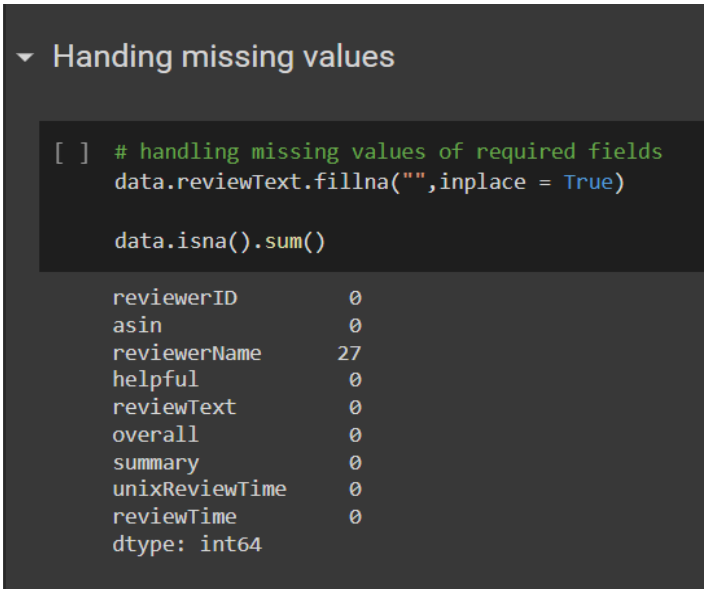


Figure 3.6: Handling of missing values

3. Duplicate values: Deduplication is an often used term that refers to the process of dealing with duplicate values. Duplication of values is also another problem with data. In real world example it can happen when a person fills in his student details twice when asked by the company. The reason behind removing of duplicates is that it should not give a particular data object an edge over the other or in simple words a bias while learning algorithms.

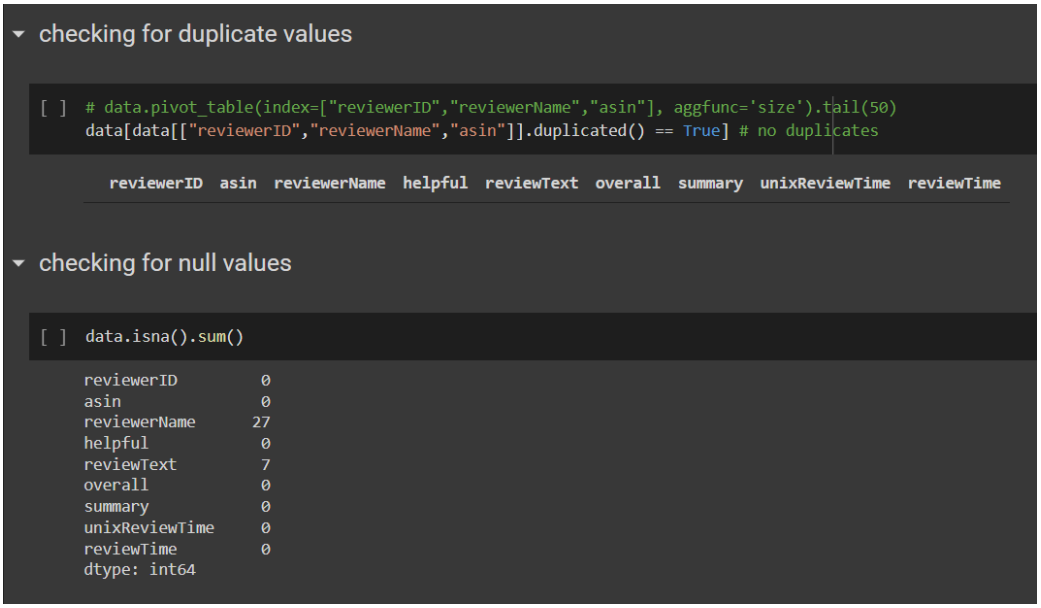


Figure 3.7: Example of isna().sum() function

4. Inconsistent values: Like duplicate values, finding inconsistent data in a dataset is highly common. For example the flight number in some rows could be sometimes written under the passenger phone number. The cause to this could simply be a human error or the information was not read correctly at the time of filling the entries.

- **Encoding categorical data:** Categorical data is that whose values are taken from a pre defined set of values. A simple example can be months of a year :{January, February, March……} as its values is always taken form this set. Now as an algorithm works on number and maths it is necessary to convert these into suitable format and can be done using various methods like LabelEncoder(), OneHotEncoder etc.

- **Splitting of data set into training and test set:** If we train our model on a dataset but when it comes to testing we use a completely different dataset we would create difficulties for the model. Doing this would also decrease the performance of our model. So we make a model that does well on the training and on the test set. So the test set can be defined as a subset to the training set to test our model on. Validation data is also another term used in ML. We use this data to improve hyperparameter tuning. The model does not learn on this validation data set.

  Split Ratio- It is largely dependent on the model we our building and on the dataset. If a lot of training is required then we use a larger portion of the data for training purposes ,example image data as it contains millions of features.

Figure 3.8: Split ratio

3.1.3 Feature Selection

Before jumping onto feature selection first lets take a look at what do we mean by the term features.

A dataset is a collection of various objects that are defines by a number of features, which simply tell us about the different characteristics of our data. Features are also called attributes, fields or variables. Like for example the mass, height of a person can be features of a dataset regarding lifestyles of various people. A feature is an individual characteristic of a phenomenon. We have to choose unbiased and

independent features for ML models so that we can have accurate regression and classification. Features are predominantly numerical but in some cases they can be graphical or strings. For example, for a scooter, colour, weight, mileage cane be observed as features. In a speech recognition system noise,sounds,power can be used as features. In algorithms using spam detection certain email headers, structures, frequency of words can be used as features. Also in character recognition, the length of a particular letter, the boldness of the letters, the spaces between words are used as or can be used as features.

Now features can be:

1. **Categorical**: These are the features whose values are chosen from a predefined set of values. Example dates in a month {1,2,3,4,5,……..} is a categorical value as it is chosen always from this set. The Boolean set is also an example of categorical data.
    i. Nominal data: These are the values that don't have any specific order. Example a new scooter comes in three color- green ,blue, red.
    ii. Ordinal data: These are the values that have a natural order within them but the difference on scale is not defined. Example when we go for buying clothes the sizes have a natural order small<medium<large but this does not ensure that the difference among them would also be the same.

2. **Numerical**: These are the features whose values are either discrete or continuous. They represented in the form of numbers and mostly posses their properties. Example speed of a plane or the number of steps we walk in a day.
    i. Interval data: This contains a defined unit of measurement and the difference between the data values is meaningful. Example include dates of calender ,temperature in C or F.
    ii. Ratio : This also contains defined unit of measurement but both differences and the ratio are meaningful. Examples Age, mass , height etc.

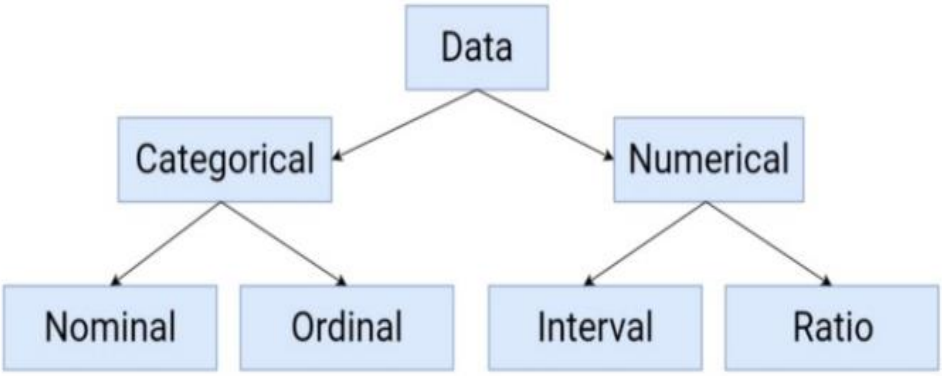


Figure 3.9: Types of features

Features can also be represented as feature vectors. In machine learning a feature vector is an n dimensional vector that represent some object. Many ML models require numerical equivalent of an object to perform statistical processing. Example in order to represent images we need to have numerical representations and that is provided y feature vector ,which contain the pixels of on image. Feature vectors usually combine with a set of weights and use dot product to produce a linear predictor function in order to determine a score for making a prediction. The space vector associated with these are called feature space. Also we can develop new features or higher level features from the already available features and adding it to the feature vector. This is called feature construction.

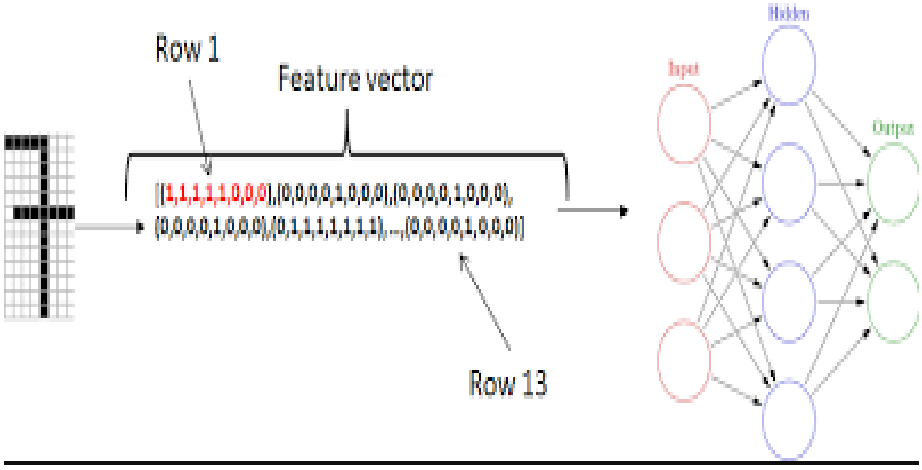


Figure 3.10: Feature vectors

In many scenarios using all the features of a dataset is not an ideal approach as some features do not contribute in increasing the efficiency of a program and also the data size would become very large. The three main goals of feature selection are to

- Reduce the cost of computation
- To improve the overall accuracy of the model
- Producing a more understandable model

Feature selection can be performed both before or after training ,it can be performed manually or by automated methods. Below are some methods.

Correlation plot: It is one of the manual techniques to perform feature selection. This creates a visualization that plots the correlation measure for every feature in the data. We can then observe which features are closely related and therefore can remove some of those , and also some variables might have a low correlation with the output variable , so it is recommended to remove some of those.

As there is no sentiment given for any review present in the dataset, we have to give the sentiment to the reviews and ratings given by the patients.It's key role is to identify the emotional tone behind the body of any text. To make the public opinion summaries useful for both pharmacists and the clinicians, this proposed sentiment analysis or what is commonly called opinion mining, for the given reviews is very helpful and important as it makes the given data more valuable in terms for the use of various parties involved seeking for the recommended medicinal drug through our model. This is a popular way to categorize the medicinal drug as useful or not taken by the respective patient.

Emotion analysis will be done using word dictionary sentiment analysis or using SentiWordnet of NLTK module, to label our pre-processed reviews data as positive or negative.

The formula that we will be using is :

Positiv_ratio = the number of positive words / (the number of positive words+the number of negative words)

If the Positiv_ratio comes out be :

1. Less than 0.5 then it is classified as negative
2. Greater than 0.5 then it is classified as positive
3. If it comes out to be exactly 0.5 then it is classified as neutral which includes the sentence without either positive or negative words.

3.1.4 Naïve Bayesian Classifier Algorithm

The Naïve Bayesian classifier works as follows:

Suppose that there exists a set of training data, D, in which each tuple is represented by an n-dimensional feature vector, X=x 1,x 2,..,x n , indicating n measurements made on the tuple from n attributes or features. Assume that there are m classes, C 1,C 2,...,C m . Given a tuple X, the classifier will predict that X belongs to C i if and only if: P(C i |X)>P(C j |X), where i,j∈[1,m]a n d i≠j. P(C i |X).

Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable. Bayes' theorem states the following relationship, given class variable *y* and dependent feature vector x_1 through x_n

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

Using the naive conditional independence assumption that

$$P(x_i|y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(x_i|y),$$

for all i, this relationship is simplified to

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}$$

Since $P(x_1, \ldots, x_n)$ is constant given the input, we can use the following classification rule:

$$P(y \mid x_1, \ldots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y)$$

$$\Downarrow$$

$$\hat{y} = \arg\max_y P(y) \prod_{i=1}^{n} P(x_i \mid y),$$

and we can use Maximum A Posteriori (MAP) estimation to estimate $P(y)$ and $P(x_i \mid y)$; the former is then the relative frequency of class y in the training set.

Multinomial Naive Bayes :

`MultinomialNB` implements the naive Bayes algorithm for multinomially distributed data, and is one of the two classic naive Bayes variants used in text classification (where the data are typically represented as word vector counts, although tf-idf vectors are also known to work well in practice). The distribution is parametrized by vectors $\theta_y = (\theta_{y1}, \ldots, \theta_{ym})$ for each class y, where n is the number of features (in text classification, the size of the vocabulary) and θ_{yi} is the probability $P(x_i \mid y)$ of feature i appearing in a sample belonging to class y.

The parameters θ_y is estimated by a smoothed version of maximum likelihood, i.e. relative frequency counting:

$$\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha n}$$

where $N_{yi} = \sum_{x \in T} x_i$ is the number of times feature i appears in a sample of class y in the training set T, and $N_y = \sum_{i=1}^{n} N_{yi}$ is the total count of all features for class y.

The smoothing priors $\alpha \geq 0$ accounts for features not present in the learning samples and prevents zero probabilities in further computations. Setting $\alpha = 1$ is called Laplace smoothing, while $\alpha < 1$ is called Lidstone smoothing.

3.2 Model Development

3.2.1 **Dataset**

The dataset was collected from Kaggle.com. Kaggle is a google owned subsidiary, web community of information scientists and ML practitioners. Kaggle lets customers to find and upload record units, discover and construct models via internet based record science surroundings, engaging with other data

scientists and also participate in competitions to solve any data science problems. We are assured that the dataset is original and authentic because kaggle is a trusted website that is used by over a million users worldwide.

Our dataset is *Amazon Update Database contains Musical Instrument* updates from Amazon. Database is available from Kaggle, we have processed it from Kaggle itself. This data was created by Chetan Gadge. It contains the following files: Update including Reviewer ID,User ID, Update Name, Reviewer text, assistant, Summary (found in Reviewer text), Total rating on the scale 5, Review period. This can help the organization understand customer backlogs.

K-cores (i.e., dense subsets): These details are reduced to extract the k-core, so that each of the remaining users and objects are updated individually.

Ratings only: These data sets do not include metadata or updates, but only (user, item, rating, timestamp) sounds. They are therefore suitable for use with mymedialite (or similar) packages.

Amazon is reviewing the full scores of data generated by random sampling of 10,000 samples for each review rating from 1 to 5. The downloaded file format was one update per JSON line. The file has been converted to Comma Separated Values (CSV) format, as it is much easier for python to manage this type of file.

DATA FORMAT:

The following is an example review in Json file:

"reviewText": string "Not much to write about here, but it does exactly what it's supposed to. filters out the pop sounds. now my recordings are much more crisp. it is one of the lowest prices pop filters on amazon so might as well buy it, they honestly work the same despite their pricing,"
"overall": int 5
"summary": string "good"
"unixReviewTime": int 1393545600
"reviewTime": string "02 28, 2014"

1. reviewerID - ID of the reviewer, e.g. A2SUAM1J3GNN3B
2. asin - ID of the product, e.g. 0000013714
3. reviewerName - name of the reviewer
4. helpful - helpfulness rating of the review, e.g. 2/3
5. reviewText - text of the review
6. overall - rating of the product
7. summary - summary of the review
8. unixReviewTime - time of the review (unix time)
9. reviewTime - time of the review (raw)

Figure 3.11: Dataset Features

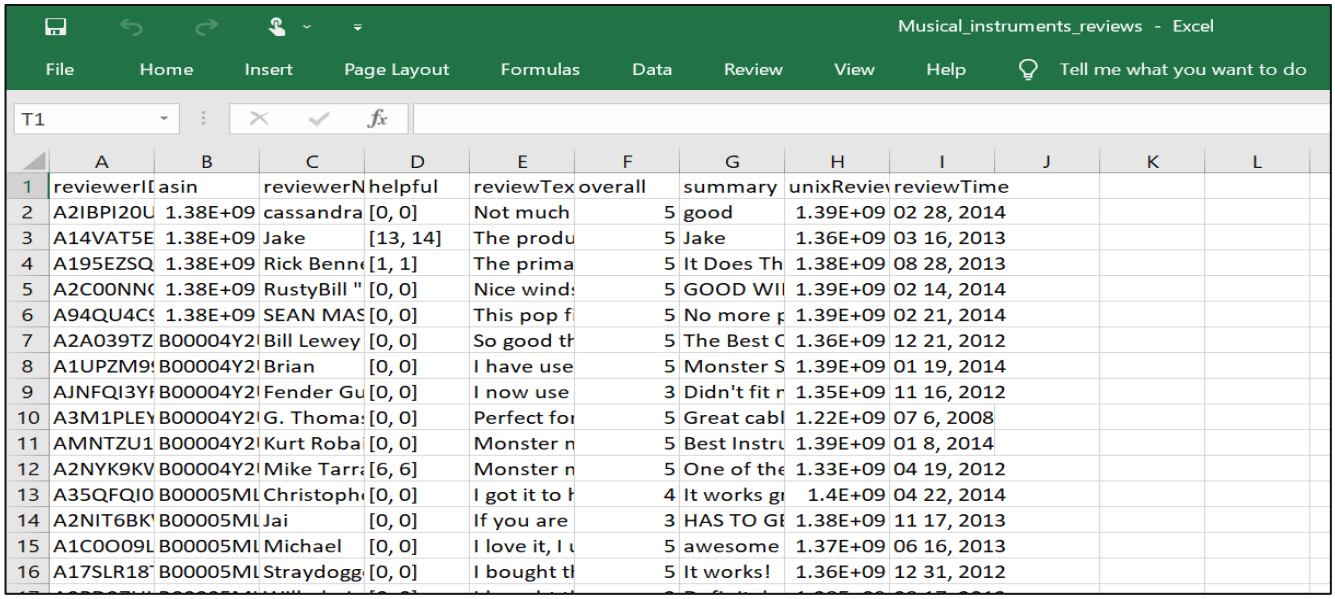
Types of Data Set :

In Statistics, we have different types of data sets available for different types of information. They are:

- *Numerical Data Sets* : A set of all numerical data. It deals only with numbers.
- *Bivariate Data Sets* : A data set that has two variables is called a Bivariate data set. It deals with the relationship between the two variables.
- *Multivariate Data Sets* : A data set with multiple variables.
- *Categorical Data Sets* : Categorical data sets represent features or characteristics of a person or an object.
- *Correlation Data Sets* : The set of values that demonstrate some relationship with each other indicates correlation data sets. Here the values are found to be dependent on each other.

Here, our dataset Amazon Musical Instruments Reviews from Kaggle is a Multivariate dataset.

Table 3.1: Dataset Information



	A	B	C	D	E	F	G	H	I	J	K	L
1	reviewerI	asin	reviewerN	helpful	reviewTex	overall	summary	unixRevie	reviewTime			
2	A2IBPI20U	1.38E+09	cassandra	[0, 0]	Not much	5	good	1.39E+09	02 28, 2014			
3	A14VAT5E	1.38E+09	Jake	[13, 14]	The produ	5	Jake	1.36E+09	03 16, 2013			
4	A195EZSQ	1.38E+09	Rick Benn	[1, 1]	The prima	5	It Does Th	1.38E+09	08 28, 2013			
5	A2C00NN0	1.38E+09	RustyBill "	[0, 0]	Nice winds	5	GOOD WI	1.39E+09	02 14, 2014			
6	A94QU4C9	1.38E+09	SEAN MAS	[0, 0]	This pop f	5	No more p	1.39E+09	02 21, 2014			
7	A2A039TZ	B00004Y2	Bill Lewey	[0, 0]	So good th	5	The Best C	1.36E+09	12 21, 2012			
8	A1UPZM9	B00004Y2	Brian	[0, 0]	I have use	5	Monster S	1.39E+09	01 19, 2014			
9	AJNFQI3Y	B00004Y2	Fender Gu	[0, 0]	I now use	3	Didn't fit r	1.35E+09	11 16, 2012			
10	A3M1PLEY	B00004Y2	G. Thomas	[0, 0]	Perfect for	5	Great cabl	1.22E+09	07 6, 2008			
11	AMNTZU1	B00004Y2	Kurt Roba	[0, 0]	Monster n	5	Best Instru	1.39E+09	01 8, 2014			
12	A2NYK9KV	B00004Y2	Mike Tarra	[6, 6]	Monster n	5	One of the	1.33E+09	04 19, 2012			
13	A35QFQI0	B00005ML	Christoph	[0, 0]	I got it to h	4	It works gr	1.4E+09	04 22, 2014			
14	A2NIT6BK	B00005ML	Jai	[0, 0]	If you are	3	HAS TO GE	1.38E+09	11 17, 2013			
15	A1C0O09L	B00005ML	Michael	[0, 0]	I love it, I u	5	awesome	1.37E+09	06 16, 2013			
16	A17SLR18	B00005ML	Straydogg	[0, 0]	I bought th	5	It works!	1.36E+09	12 31, 2012			

3.2.2 Computational

For this project work we used the machine with the following specs at the time of training.

CPU: The computer we used had the following specs:

Table 3.2 CPU specifications

Parameter	Specifications
CPU Model name	Intel(R)Core(TM)
CPU frequency	2.3 Ghz
No of CPU cores	4
Available Ram	7.86 GB
Disk Space	400 GB

These are the CPU specs of the machine we used to do computations. Most of the computational work mostly happens on the GPU. But CPU takes care of most of the preprocessing. The large amount of RAM did not put loads of pressure and made it easier for the whole dataset to be loaded in time and we did not have to worry about any system crashes occurring. The clock speed of the CPU mentioned 2.3 Ghz is the basic clock speed which if needed can go upto 5 Ghz. But no overclocking was needed as the system was able to do the work in its normal 4 cores.

Table 3.3: GPU Specifications

Parameter	Specifications
GPU	NVIDIA GeForce GTX 1060
GPU Memory	10 GB
GPU Memory Clock	1.40 Ghz
GPU Release Year	2016
Cores	2
Available RAM	6 GB
Disk Space	400 GB

Tools Used: We used the following tools in making our model.

- Python
- Matplotlib
- Seaborn
- Jupyter Notebook
- Pandas
- Plotly
- Numpy
- Scikit Learn
- Flask Framework

These packages mentioned above were used in their latest upto date editions. The code works properly and would not cause any issue until any further updates in them.

3.2.3 Experimental

Now we tried a few algorithms just to check them in comparison to what we used.

i. Logistic Regression
ii. Decision Trees
iii. Support Vector Machines

We are also providing with some basic definitions of the above algorithms so as to explain easily and look why those algorithms did not perform the way desired.

Logistic Regression

Logistic regression predicts the probability of an outcome that can only have two values (i.e. a dichotomy). Prediction is based on the use of one or more predictions (numbers and categories). Line reversal is incorrect to predict the value of binary fluctuations for two reasons:

➢A linear regression will predict values outside the acceptable range (e.g. predicting probabilities outside the range 0 to 1)

➢Since the dichotomous experiments can only have one of two possible values for each experiment, the residuals will not be normally distributed about the predicted line.

On the other hand, the reversal of order produces a curve of objects, which is limited to values between 0 and 1. The reversal of order is similar to the reversal of the line, but the curve is constructed using the natural logarithm of the "constraints" of the target, rather than the probability. In addition, forecasters do not need to be widely distributed or have equal variations in each group.

An asset deficit is used as the maximum probability (MLE) to obtain model coefficients that correlate a prediction in a target. After limiting the initial operation, the process is repeated until the LL (Log Likelihood) does not change significantly.

$$Accuracy = 75\%$$

Decision trees:

This is an algorithm we already talked about . It uses a tree approach to get to the results. In this algorithm we calculate entropy of every feature and the one with the least is selected as the root node and we keep on splitting for the remaining columns. The reason behind the failure of this algorithm is that he model works on data in a sequential manner and the feature that has more domination outlasted the other feature.

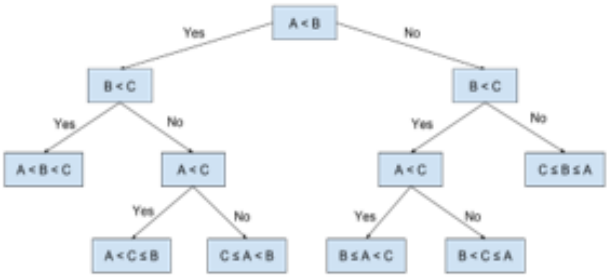


Figure 3.12: Decision Tree

$$Accuracy = 89\%$$

Support Vector Machine (SVM)

It is a method for the classification of both linear and nonlinear data. If the data is linearly separable, the SVM searches for the linear optimal separating hyperplane (the linear kernel), which is a decision boundary that separates data of one class from another. Mathematically, a separating hyperplane can

be written as: W·X+b=0, where W is a weight vector and W=w1,w2,...,w n. X is a training tuple. b is a scalar. In order to optimize the hyperplane,

The problem essentially transforms to the minimization of ‖ W‖ , which is eventually computed as:

$\sum_{i=1}^{n} \alpha_i y_i x_i$, where α_i are numeric parameters, and y_i are labels based on support vectors, X_i .

That is: if y_i=1 then

$$\sum_{i=1}^{n} w_i x_i \geq 1;$$

if y_i=−1 then

$$\sum_{i=1}^{n} w_i x_i \geq -1.$$

Figure 3.13: SVM Parameters

$Accuracy = 91\%$

3.2.4 Mathematical

Down below we have listed the formulas used by us in the report.

Logistic Regression:

y = e^(b0 + b1*x) / (1 + e^(b0 + b1*x))

Where y is the predicted output, b0 is the bias or intercept term and b1 is the coefficient for the single input value (x). Each column in your input data has an associated b coefficient (a constant real value) that must be learned from your training data.

Naive Bayes Formula:

P(c|x) = P(x|c) * P(c) / P(x)

Bayes theorem calculates probability P(c|x) where c is the class of the possible outcomes and x is the given instance which has to be classified, representing some certain features.

Decision Tree:

Entropy:

$$\mathbf{E(S)} = -p_{(+)} \log p_{(+)} - p_{(-)} \log p_{(-)}$$

Here p+ is the probability of positive class

p– is the probability of negative class

S is the subset of the training example

Chapter 4 PERFORMANCE ANALYSIS

4.1 Analysis of System

All these steps include the sentiment analysis of our data and then using n-gram to make the most sense out of the reviews, followed by a LightGBM model that allows us to calculate the final predicted value and recommend the appropriate drug for each condition according to the order of the value. Hence making the process of recommending first grade medicine not only for doctors or any health care workers but also for the patients themselves easier and effective. We find the accuracies and the confusion matrices for the models for better understanding of the results.

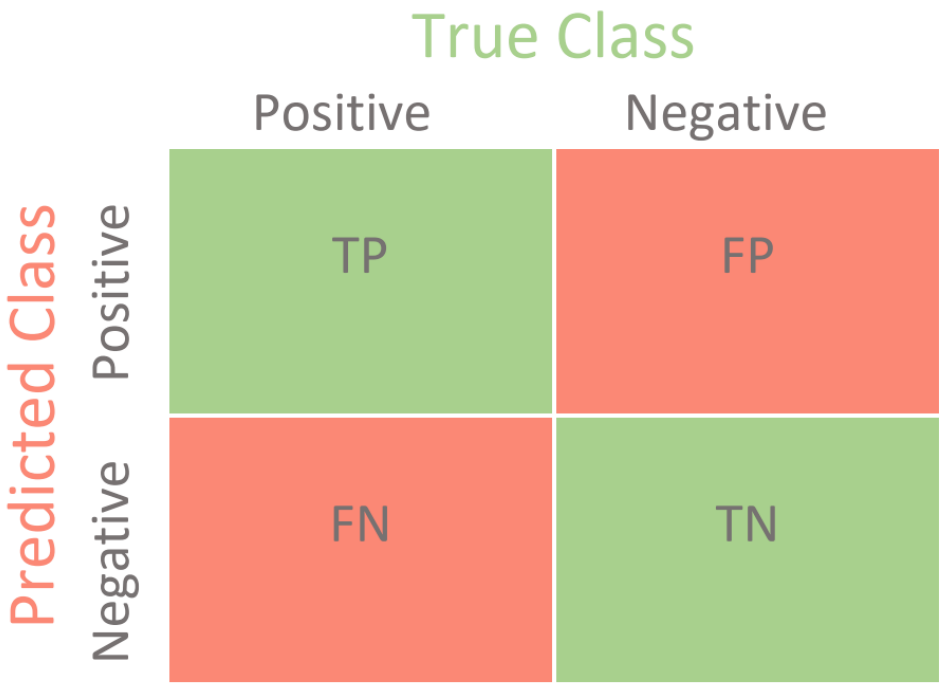
Accuracy is defined as the percentage of correct predictions for the test data. It can be calculated easily by dividing the number of correct predictions by the number of total predictions.

$$\text{accuracy} = \frac{\text{correct predictions}}{\text{all predictions}}$$

Classification metrics

When performing classification predictions, there's four types of outcomes that could occur.

- **True positives** are when you predict an observation belongs to a class and it actually does belong to that class.
- **True negatives** are when you predict an observation does not belong to a class and it actually does not belong to that class.
- **False positives** occur when you predict an observation belongs to a class when in reality it does not.
- **False negatives** occur when you predict an observation does not belong to a class when in fact it does.



Analysis of the system is the thing where we compare our system with the previous implementations and check the performance of the system. Sentiment analysis and natural language processing can present opportunities to improve customer experience, reduce employee profits, build better products, and more. Many companies manually analyze customer feedback details. It makes sense to others, especially if there are small amounts of data available. However, at a high volume level, there are many human-enabled analytics problems — the main one of which is speed. In addition, a person tends to have the ability to perform analysis once a week, which means they do not have a real-time understanding of customer complaints that can help them to be able to firmly resolve an unusual source of customer problems. This type of customer information plays an important role in making business decisions. Allowing firms to respond to the needs of their customers and thus reduce customer stress and keep them competitive.

Given below are the results of different models used on this task before which includes the models used by us.

Table 4.1: Models and Accuracy

Model Name	**Accuracy**

MultinomialNB	0.9226
SVM	0.9071
Decision Tree Classifier	0.8889
Logistics Regression	0.7480

After testing some arbitrary reviews, it seems that our features is performing correctly with Positive, Neutral, Negative results We also see that after running the search, our MultinomialNB Machine Classifier has improved to 92.26% accuracy level.

Below are shown some of the output at various stages of this project.

```
[ ] data = pd.read_csv("Musical_instruments_reviews.csv")
    data.head()
```

	reviewerID	asin	reviewerName	helpful	reviewText	overall	summary	unixReviewTime	reviewTime
0	A2IBPI20UZIR0U	1384719342	cassandra tu "Yeah, well, that's just like, u...	[0, 0]	Not much to write about here, but it does exac...	5.0	good	1393545600	02 28, 2014
1	A14VAT5EAX3D9S	1384719342	Jake	[13, 14]	The product does exactly as it should and is q...	5.0	Jake	1363392000	03 16, 2013
2	A195EZSQDW3E21	1384719342	Rick Bennette "Rick Bennette"	[1, 1]	The primary job of this device is to block the...	5.0	It Does The Job Well	1377648000	08 28, 2013
3	A2C00NNG1ZQQG2	1384719342	RustyBill "Sunday Rocker"	[0, 0]	Nice windscreen protects my MXL mic and preven...	5.0	GOOD WINDSCREEN FOR THE MONEY	1392336000	02 14, 2014
4	A94QU4C90B1AX	1384719342	SEAN MASLANKA	[0, 0]	This pop filter is great. It looks and perform...	5.0	No more pops when I record my vocals.	1392940800	02 21, 2014

Figure 4.1 : Original dataset

Here we check for duplicate values as well as handling of missing values :

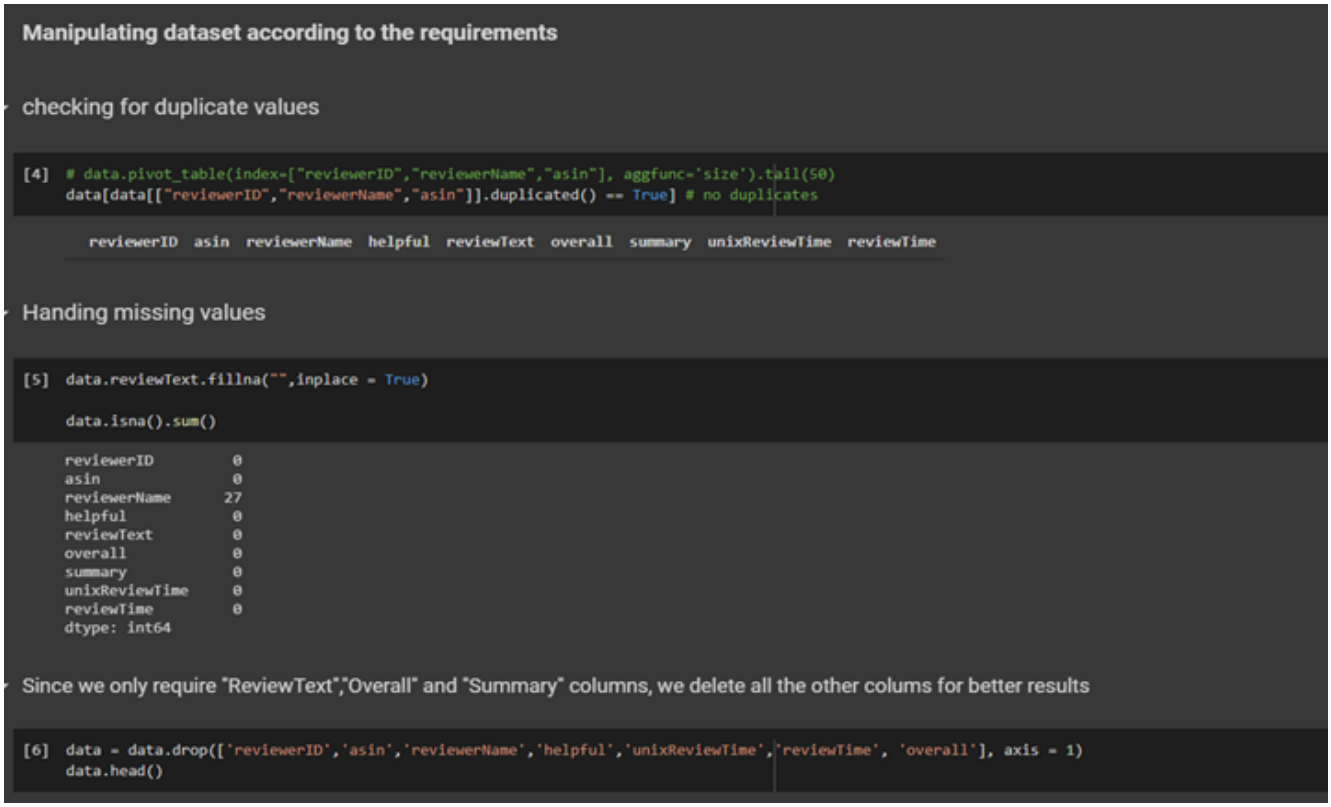


Figure 4.2: Data Manipulation :

Data Preprocessing and Visualization:

Here we converted all the categorical data and did some visualizations.

Data Preprocessing

```
[8]  def preprocess_Reviews_data(data,name):
         # Proprocessing the data
         data[name]=data[name].str.lower()
         # Code to remove the Hashtags from the text
         data[name]=data[name].apply(lambda x:re.sub(r'\B#\S+','',x))
         # Code to remove the links from the text
         data[name]=data[name].apply(lambda x:re.sub(r"http\S+", "", x))
         # Code to remove the Special characters from the text
         data[name]=data[name].apply(lambda x:' '.join(re.findall(r'\w+', x)))
         # Code to substitute the multiple spaces with single spaces
         data[name]=data[name].apply(lambda x:re.sub(r'\s+', ' ', x, flags=re.I))
         # Code to remove all the single characters in the text
         data[name]=data[name].apply(lambda x:re.sub(r'\s+[a-zA-Z]\s+', '', x))
         # Remove the twitter handlers
         data[name]=data[name].apply(lambda x:re.sub('@[^\s]+','',x))

     # Function to tokenize and remove the stopwords
     def rem_stopwords_tokenize(data,name):

         def getting(sen):
             example_sent = sen

             filtered_sentence = []

             stop_words = set(stopwords.words('english'))

             word_tokens = word_tokenize(example_sent)

             filtered_sentence = [w for w in word_tokens if not w in stop_words]

             return filtered_sentence
         # Using "getting(sen)" function to append edited sentence to data
         x=[]
         for i in data[name].values:
             x.append(getting(i))
         data[name]=x
```

Figure 4.3: Data Preprocessing

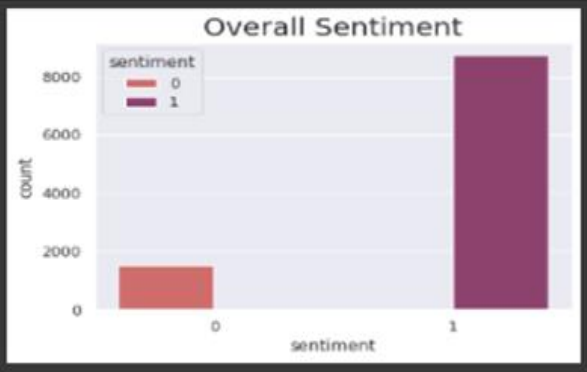
This is a vital part of training the dataset. Here Words present in the file are accessed both as a solo word and also as pair of words. Because, for example the word “bad” means negative but when someone writes “not bad” it refers to as positive. In such cases considering single word for training data will work otherwise. So words in pairs are checked to find the occurrence to modifiers before 19 any adjective which if present which might provide a different meaning to the outlook.

Sentiment Classification :

```
[16] overall=[]
     for i in range(len(data)):
         if data['sentiment'][i]>= 0.05:
             overall.append('Positive')
         elif data['sentiment'][i]<= -0.05:
             overall.append('Negative')
         else:
             overall.append('Neutral')
     data['sentiment']=overall

[17] def convert_sentiment(sentiment):
         if(sentiment=='Negative'):
             return 0
         else:
             return 1

[18] data.sentiment = data.sentiment.apply(convert_sentiment)

[19] data.head(30)
```

Figure 4.4: Sentiment Classification

```
[20] sns.set(style="darkgrid")
     bar = sns.countplot(x = 'sentiment' , hue = 'sentiment' , data = data, palette="flare")
     bar.set_title("Overall Sentiment", size = 20)
     plt.show()
```



we can tell by this plot, that the data is unbalanced

Figure 4.5: Overall Sentiment

Word Cloud :

Creating Word Cloud

(visualize the frequent words)

```
[21] all_words = " ".join([sentence for sentence in data['feedback']])

     from wordcloud import WordCloud
     wordcloud = WordCloud(width=800, height=500, random_state=42, max_font_size=100).generate(all_words)

     # plot the graph
     plt.figure(figsize=(15,8))
     plt.imshow(wordcloud, interpolation='bilinear')
     plt.axis('off')
     plt.show()
```

Figure 4.6: World Cloud

Different ML Models :

Applying Maching Learning Models

```
[25] from sklearn.model_selection import train_test_split
     x_train1,x_test1,y_train1,y_test1 = train_test_split(X_res,y_res,test_size = 0.2 , random_state = 0)
```

```
[26] ### Logistic Regression
     from sklearn.linear_model import LogisticRegression
     ### Naive Bayes Classifier - MultinomialNB
     from sklearn.naive_bayes import MultinomialNB
     ### Support Vector Machine
     from sklearn import svm
     ### Decision Trees Classifier
     from sklearn.tree import DecisionTreeClassifier
     ### Random Forest Classifier
     from sklearn.ensemble import RandomForestClassifier
     ### Classification Report
     from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
```

```
[27] from sklearn.metrics import mean_squared_error,confusion_matrix, precision_score, recall_score, auc,roc_curve
     MLA_columns = []
     MLA_compare = pd.DataFrame(columns = MLA_columns)
     row_index = 0
```

Figure 4.7: ML Models

Performance Metrics from Various ML Models:

	precision	recall	f1-score	support
0	1.00	0.43	0.61	1319
1	0.69	1.00	0.81	1641
accuracy			0.75	2960
macro avg	0.84	0.72	0.71	2960
weighted avg	0.83	0.75	0.72	2960

Logistics Regression

	precision	recall	f1-score	support
0	1.00	0.83	0.90	1319
1	0.88	1.00	0.93	1641
accuracy			0.92	2960
macro avg	0.94	0.91	0.92	2960
weighted avg	0.93	0.92	0.92	2960

Multinomial NB

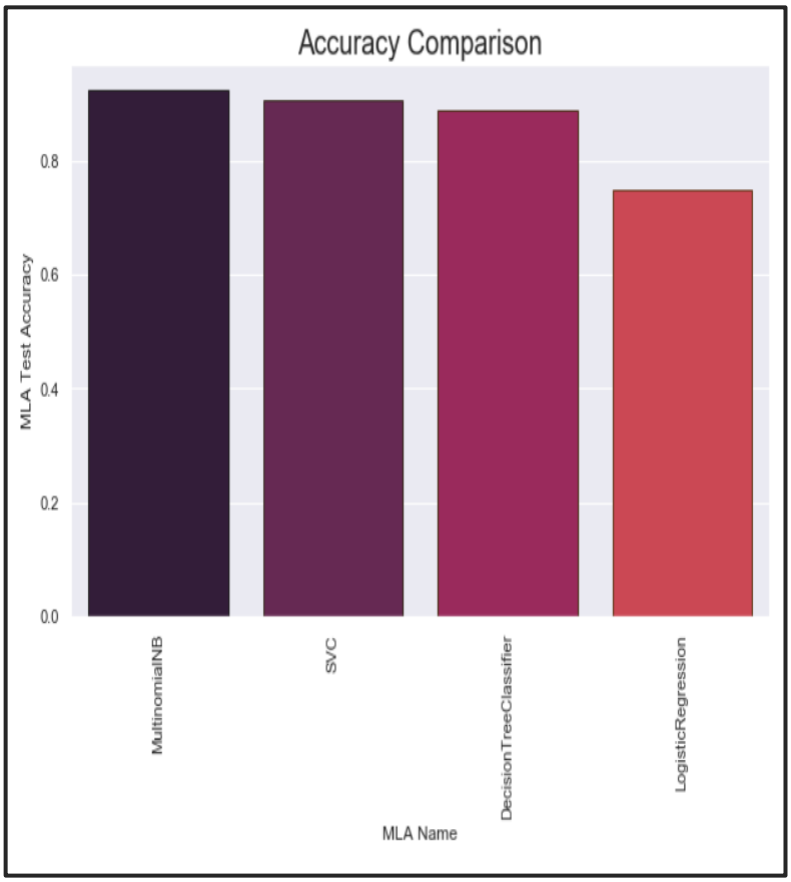
	precision	recall	f1-score	support
0	0.99	0.80	0.88	1319
1	0.86	0.99	0.92	1641
accuracy			0.91	2960
macro avg	0.93	0.90	0.90	2960
weighted avg	0.92	0.91	0.91	2960

SVM

	precision	recall	f1-score	support
0	1.00	0.75	0.86	1319
1	0.83	1.00	0.91	1641
accuracy			0.89	2960
macro avg	0.92	0.88	0.88	2960
weighted avg	0.91	0.89	0.89	2960

Decision Tree

Accuracy Comparisons :



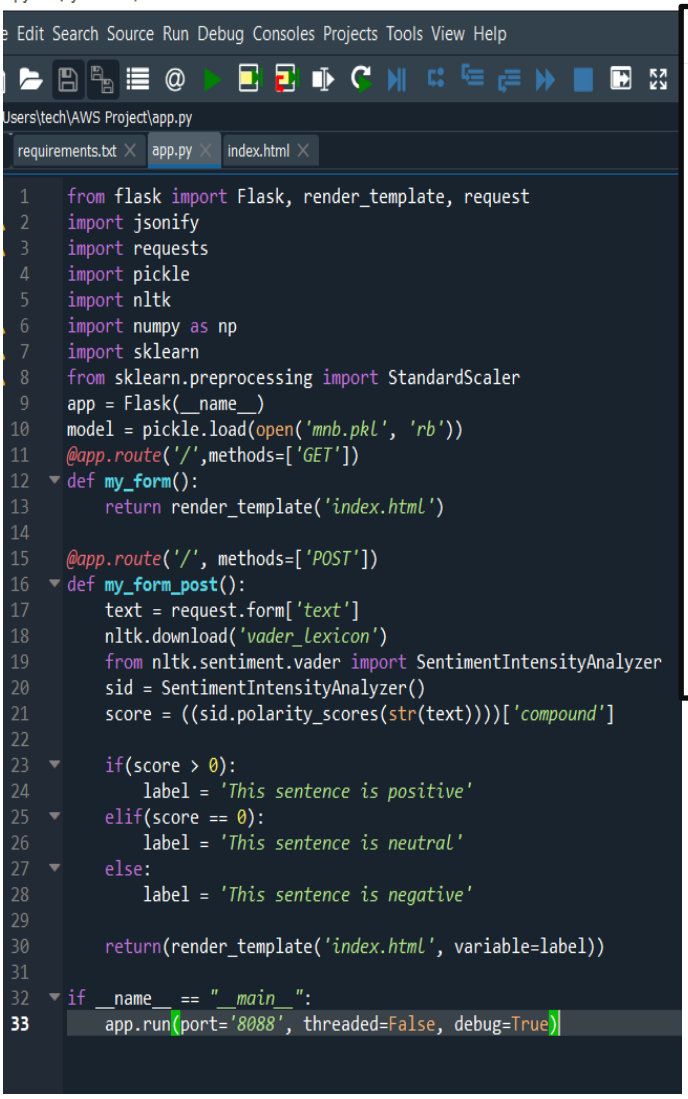
	MLA Name	MLA Test Accuracy	MLA Precision
1	MultinomialNB	0.9226	0.877540
2	SVC	0.9071	0.860232
3	DecisionTreeClassifier	0.8889	0.832995
0	LogisticRegression	0.7480	0.687474

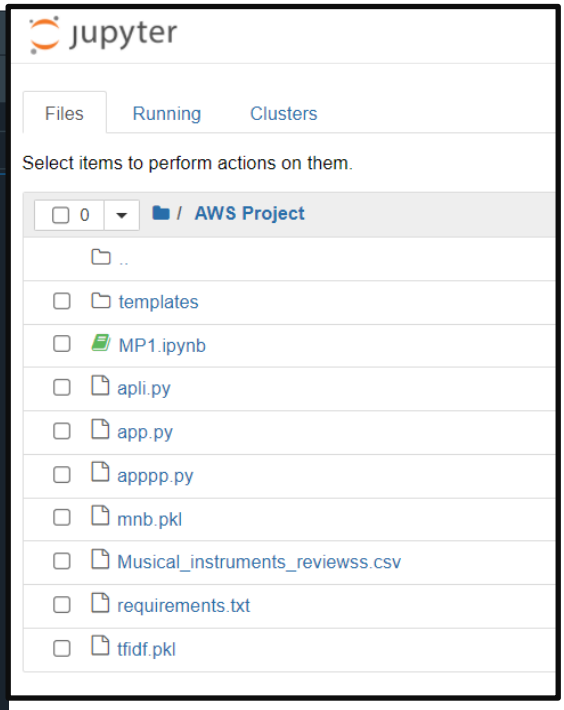
Build a Flask Website to serve a Model :

Flask is a framework in python that helps in the web development application easily. Flask is basically a collection of libraries and modules that creates user friendly experience and also is easy to use that makes it a good choice for anyone. It is a very small and light framework used in python and it gives the user very flexible and vast usage of easy coding that can be used in a single file for creating a web application.

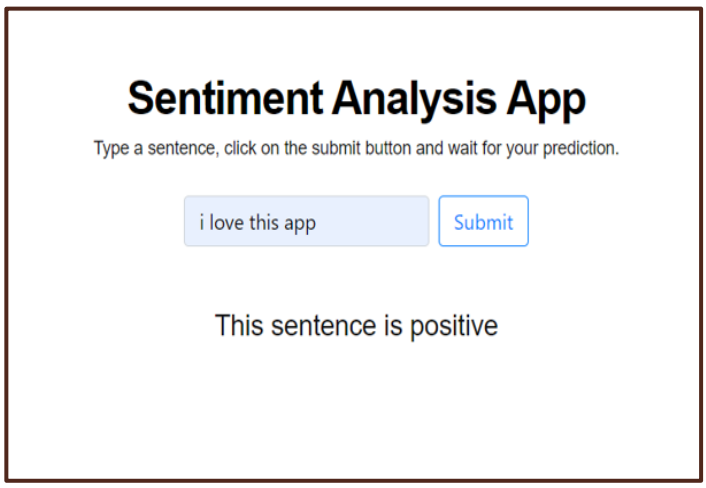
Here, our best accurate model i.e Multinomial NB pickle file extracted through Flask framework inside which several libraries are imported along with requirements.

Spyder (Python 3.7)





Predictions :



For a Positive Feedback, The model returns with 'Positive' and vice-versa.

Deploy the Flask website on AWS EC2 :

In order to deploy our model, we first created a flask website above. After that, to deploy our best fitted model, we will use AWS EC2 Instance which will give us a server where we can deploy our flask website and hence it can be used by anyone with the link of the website.

An Amazon EC2 instance is a virtual server in Amazon's Elastic Compute Cloud (EC2) for running applications on the AWS infrastructure. Amazon EC2 is used to create and run virtual machines in the cloud. AWS EC2 Instance is free of cost and easy to use.

We have to create an Ubuntu instance using EC2 and after completing all the necessary requirements we will be able to deploy our model correctly.

Steps :

1. Create an AWS EC2 instance and edit the security group.
2. Then download the keygen (pem file) and similarly download and install Putty and WinSCP
3. Upload Flask website to EC2 using WinSCP
4. Install packages on EC2 using Putty.

Chapter 5 CONCLUSIONS

5.1 Conclusions

These days online shopping has taken a step-up to a level that almost everything is available online easily which gives us a lot of choices but, buying any product needs a lot of research and comparison. This comparison can be done easily if we can have the existing users' experiences. This can be a=made easier by reading the user rivews. This project helps us in sorting the reviews in two parameters i.e positive and negative. Based on the data we used and the models and algorithms we have used provided us a satisfactory result for the comparison of reviews that made our model precise and accurate in sorting the positive and negative all together.

This report further shows that ML predictors(models) are a more than satisfactory option to sort the reviews and makes the buying experience a bit easy. To the level of our understanding , most of the preceding research work on sentiment analysis focused on traditional statistical procedures, which have their own obstacles of estimating and prediction. Also we came to know that proper data and feature extraction and selection are an important part of this process and helped us to come out with some helpful insights. Many features were extracted from the data that make sentiment analysis with negation handling easy to understand. From this project we can get information about the various factors that play a main role in the prediction of the reviews in terms of positive and negative.

We can now with full conviction draw a conclusion that if the model implemented is done in a proper manner can be of great use in saving money of many people by providing them with the information about the product also give them an idea about the competing product in the same segment which would help them to decide easily what will be the best for them.

Also various ML models are studied and their efficiency and performance are compared so as to get better results. With the help of this project we can conclude that sentiment analysis with negation handling is of great use for anyone and thus helping them to pick the best of all the products available in the market making it of a great use.

5.2 Future Scope

Sentiment analysis is an emerging research area in text mining and computational linguistics, and has attracted considerable research attention in the past few years. Future research shall explore sophisticated methods for opinion and product feature extraction, as well as new classification models that can address the ordered labels property in rating inference. Applications that utilize results from sentiment analysis is also expected to emerge in the near future.

Our work is currently limited to just english language data exclusively, and we focus to expand our work to many other languages,

Another work that we wish to do in future is to detect sarcasm where someone might rate the product more and give bad reviews or vice versa in order to achieve better results for custom feedback.

We will deploy our trained model in one of the main cloud platforms i.e AWS SageMaker or Microsoft Azure.

5.3 Application of Our Project

➢Sentiment analysis and natural language processing can present opportunities to improve customer experience, reduce employee profits, build better products, and more. The most common applications for natural language processing fall into three broad categories: Social Media Monitoring, Customer Experience Management and Voice of Customer, and People Analytics and Voice of Employee.

➢Our Project Itself is based on one of the Applications of Sentiment Analysis -- Customer Feedback

➢Customer surveys, reviews and support tickets contain information that can drive retention, increase conversion rates, and improve the quality of life of customers. This project provides the entry and exit of customer feedback analysis so that we can turn the response into customer feedback.

➢Many companies manually analyze customer feedback details. It makes sense to others, especially if there are small amounts of data available. However, at a high volume level, there are many human-enabled analytics problems — the main one of which is speed. In addition, a person tends to have the ability to perform analysis once a week, which means they do not have

a real-time understanding of customer complaints that can help them to be able to firmly resolve an unusual source of customer problems.

➢Text analytics (read our full guide to text analytics here), on the other hand, eliminates the need for human-enabled analysis. Emotional analysis, a subset of text analytics, classifies this text as constructive, negative, or neutral and can do up to 100,000 reviews per minute. While NLP can convert unedited text into unmeasured data - for example, the frequency of updates related to specific topics.

➢This type of customer information plays an important role in making business decisions. Allowing firms to respond to the needs of their customers and thus reduce customer stress and keep them competitive.

REFERENCES

[1] Umar Farooq, Yacine Ouzrout, Muhammad Abdul Qadir, May 21, 2016 DISP Laboratory, University Lumiere Lyon 2, Lyon, France, Negation Handling in Sentiment Analysis at Sentence Level

[2] Tanjim Ul Haque, Nudrat Nawal Saber, Faisal Muhammad Shah, Department of Computer Science & Engineering, Ahsanullah University of Science & Technology, Dhaka, Bangladesh, 12 May 2018, Sentiment analysis on large scale Amazon product reviews

[3] Ansari Fatima Anees, Arsalaan Shaikh, Arbaz Shaikh, Sufiyan Shaikh, January 15 2020, Survey Paper on Sentiment Analysis: Techniques and Challenges

[4] Sanjay Dey, Sarhan Wasif, Subrina Sultana, Monisha Dey, 27 April 2020, A Comparative