REVIEWS USING SENTIMENTAL ANALYSIS

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

SHARVIRALA KETHAN (Reg.No - 39110936)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade “A” by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI - 600119

APRIL - 2023

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **SHARVIRALA KETHAN(Reg.No-39110936)** who carried out the Project Phase-2 entitled REVIEWS USING SENTIMENTAL ANALYSIS under my supervision from Jan 2023 to April 2023.

Internal Guide

Dr. A. Mary Posonia, M.E., Ph.D.,

Head of the Department

Dr. L. LAKSHMANAN, M.E., Ph.D.

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY ★
DEPARTMENT OF CSE

Submitted for Viva voce Examination held on 19.04.2024

Jany

J. Refonaa

Internal Examiner **External Examiner**

DECLARATION

I, **SHARVIRALA KETHAN (Reg.No- 39110936),** hereby declare that the Project Phase-2 Report entitled “**REVIEWS USING SENTIMENTAL ANALYSIS”** done by me under the guidance of **Dr. A. Mary Posonia,M.E.,Ph.D** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

Kethan.

DATE: 19-04-2023

PLACE: Chennai **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph. D**, **Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.,** Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. A. Mary Posonia,M.E.,Ph.D,**for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-2 project work.

I wish to express my thanks toall Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

In today’s digitalized world, e-commerce is taking the ascendancy by making products available within the reach of customers thereby eliminating the need for them to step out of their homes. As customers are buying products online, product reviews are the most reliable way for them to decide whether to buy a product or not. Therefore, sentiment analysis is essential to understand a product’s popularity among buyers. Sentiment analysis also known as opinion mining is a classification process where machine learning techniques are applied on text-driven datasets in order to extract the polarities involved in it i.e., positive or negative. In this project, four different machine learning algorithms i.e., Multinomial Naive Bayes (MNB), Bernoulli Naive Bayes (BNB), Support Vector Machine (SVM) and Random Forest Classifier which is an ensemble technique are compared. The data used in this study contains Amazon reviews.

In day to day life the Reviews has became like part of cooperative industry. Because by using the reviews only the cooperative company come to now how there products are reaching to the customer and their reactions to the products. Using these reviews only they come to now how their feedback is given to the products so using the reviews only they can decide to change the products according to their customers.so in this project the user need to know that reviews of products form customer to do changes or to remove from the Product roaster so for that we are going to take the dataset where it has the all the products with their unique id and process start with the data exploration and then we go for the correlation and for the nature of reviews we use sentimental analysis and the find the which algorithm is best for the process using the accuracy and then we classify the algorithm and we get the classification report for the model and then we get to now the reviews of customers.
Keywords— Machine learning, Python, Dataset, Data Exploration, Correlations, Sentimental Analysis, Naive Bayes, Logistic Regression Classifier, Support Vector Machine Classifier, Decision Tree Classifier, Random Forest Classifier.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

People buy goods from multiple e-commerce websites when the entire world's commercial sites are virtually on the internet. Reviews of products before purchases are frequently a matter of prerogative. Consumers are more likely to purchase a good after reading reviews. Online retailers and vendors solicit feedback from their customers on their goods. Every day, millions of reviews of products, facilities, and locations are posted online. People buy goods from multiple e-commerce websites when the entire world's commercial sites are virtually on the internet. Reviews of products before purchases are frequently a matter of prerogative. Consumers are more likely to purchase a good after reading reviews. Online retailers and vendors solicit feedback from their customers on their goods. Every day, millions of reviews of products, facilities, and locations are posted online. In the age of artificial intelligence, it takes time to polarize a sample of reviews into particular categories in order to assess a company's appeal to consumers around the world. Analyzing data from specific consumer comments is an important area today.

Customers can write product or service reviews in areas of e-commerce websites to provide comments, suggestions, and thoughts on products. Therefore, e-commerce enterprises absolutely require the study of this review. The number of online reviews is enormous today, and the number of forums is outpacing growth. Several web formats for consumer analysis are available, including products of a particular type (like video cameras), articles from publications like Rolling Stone and customer reports, industry articles on businesses like Amazon, and technical and pages for user analysis in a variety of domains. The websites and forums of websites like Blogstreet.com, AllConsuming.net, and onfocus.com also feature customer reviews of products. Amazon is one of the e-commerce pioneers that consumers use daily for online shopping and receives hundreds of reviews from customers about their favorite products. Consumers rate products and services. As a result of individual ratings determining exact values during the analysis, reviews are grouped with incompatible ratings.

Having said that, finding and monitoring opinion sites on the Web, as well as distilling the information contained in them, remains a difficult task due to the proliferation of diverse sites. Each site typically contains a large amount of opinionated text that is not always easy to decipher in lengthy forum posts and blogs. The average human reader will struggle to identify relevant sites and accurately summaries the information and opinions contained within them. Furthermore, teaching a computer to recognize sarcasm is a complex and difficult task because computers cannot think like humans at the moment.

When the world's commercial sites are virtually on the internet web, people purchase products from numerous e-commerce websites. It is often a prerogative situation where products are reviewed before they are purchased. Consumers are more inclined to buy a commodity from feedback. Online sellers and dealers ask their buyers to share their views on their products. Millions of feedbacks regarding goods, facilities and locations are generated daily on the internet. This makes the internet the primary source of information for a product or service. Therefore, reviews provide useful views on a company, including its location, pricing and advice that allow consumers to understand almost every aspect of the company. This is not only good for customers, but it also allows retailers to consider buyers and their desires that make their goods entirely. As the number of comments available for a company increases, for a prospective customer, it becomes more difficult to consider whether to buy it or not. In this era of artificial intelligence, reading thousands of feedbacks to understand a company requires time to polarise a sample in specific categories, in order to consider its appeal among consumers around the world. A significant area today is to analyse the data from specific consumer feedback.

E-Commerce websites have sections where customers can post product or service reviews for critical feedback, recommendations, and opinions on goods. Thus, the need for the study of this review is essential for e-commerce firms. Today, there are a significant amount of reviews on the internet, and the list of forums is overgrowing. Consumer analysis can be seen in several online forms; items of a specified kind (such as video cameras), newspapers or magazines with articles (such as Rolling Stone and customer reports), industry articles on companies (such as Amazon), technical and user analysis pages in a selection on fields. Pages such as Blogstreet.com, AllConsuming.net and onfocus.com also have consumer feedback on items on their websites and forums. Amazon is one of the pioneers for e-commerce used by consumer's every day for online shopping and receives thousands of ratings about their favourite items by the consumer. The Amazon ranking system ranges from 1 to 5, where the consumers offer the product or service with a rating, the worst being

"1" and the highest being "5". Reviews are categorised with incompatible ratings as individual ratings determine exact values during the analysis.

The purpose of this paper is to classify customers' positive and negative reviews of various products and to build a supervised learning model to polarise large amounts of reviews. Our dataset is made up of customer reviews and ratings obtained from Amazon's Consumer Reviews. We extracted the features from our dataset and used them to build several supervised models.

Sentiment is an attitude, thought, or judgment prompted by feeling. Sentiment analysis [1-8], which is also known as opinion mining, studies people's sentiments towards certain entities. Internet is a resourceful place with respect to sentiment information. From a user's perspective, people are able to post their own content through various social media, such as forums, micro-blogs, or online social networking sites. From a researcher's perspective, many social media sites release their application programming interfaces (APIs), prompting data collection and analysis by researchers and developers. For instance, Twitter currently has three different versions of APIs available [9], namely the REST API, the Search API, and the Streaming API. With the REST API, developers are able to gather status data and user information; the Search API allows developers to query specific Twitter content, whereas the Streaming API is able to collect Twitter content in realtime. Moreover, developers can mix those APIs to create their own applications. Hence, sentiment analysis seems having a strong fundament with the support of massive online data.

However, those types of online data have several flaws that potentially hinder the process of sentiment analysis. The first flaw is that since people can freely post their own content, the quality of their opinions cannot be guaranteed. For example, instead of sharing topic-related opinions, online spammers post spam on forums. Some spam are meaningless at all, while others have irrelevant opinions also known as fake opinions [10-12]. The second flaw is that ground truth of such online data is not always available. A ground truth is more like a tag of a certain opinion, indicating whether the opinion is positive, negative, or neutral. The Stanford Sentiment 140 Tweet Corpus [13] is one of the datasets that has ground truth and is also public available. The corpus contains 1.6 million machine-tagged Twitter messages. Each message is tagged based on the emoticons (☺as positive, ☹as negative) discovered inside the message.

Data used in this paper is a set of product reviews collected from Amazon [14], between February and April, 2014. The aforementioned flaws have been somewhat overcome in the following two ways: First, each product review receives inspections before it can be posted a. Second, each review must have a rating on it that can be used as the ground truth. The rating is based on a star-scaled system, where the highest rating has 5 stars and the lowest rating has only 1 star.

This paper tackles a fundamental problem of sentiment analysis, namely sentiment polarity categorization . Figure 2 is a flowchart that depicts our proposed process for categorization as well as the outline of this paper. Our contributions mainly fall into Phase 2 and 3. In Phase 2: 1) An algorithm is proposed and implemented for negation phrases identification; 2) A mathematical approach is proposed for sentiment score computation; 3) A feature vector generation method is presented for sentiment polarity categorization. In Phase 3: 1) Two sentiment polarity categorization experiments are respectively performed based on sentence level and review level; 2) Performance of three classification models are evaluated and compared based on their experimental results.

The rest of this paper is organized as follows: In section ‘Background and literature review’, we provide a brief review towards some related work on sentiment analysis. Software package and classification models used in this study are presented in section ‘Methods’. Our detailed approaches for sentiment analysis are proposed in section ‘Background and literature review’. Experimental results are presented in section ‘Results and discussion’. Discussion and future work is presented in section ‘Review-level categorization’. Section ‘Conclusion’ concludes the paper.

There is a vigorous improvement in the micro blogging websites as well as social networks. One of the major web destinations to the users is micro blogging websites, which are helpful for expressing the user's attitudes, opinions, and thoughts regarding various contexts .The most used social networking services and the micro blogging platform is twitter, which provides more data. At present, for the sentiment analysis of the user's opinions on the product, event, or context, researchers make use of social data. Moreover, the other name for sentiment analysis is opinion mining, which is the significant NLP task. This sentiment analysis defines orientation of sentiments related to text as either neutral, positive, or negative [23] [24]. Moreover, sentiment analysis represents the text analytics, computational linguistics, and NLP implementations for recognizing and categorizing the opinions of the user. In general, the main intention of the sentiment analysis is to define the author's point of view concerning the similar context or the entire document's contextual polarity. The view can be either a user's judgment or assessment, affective state or the deliberated communication of emotion. In general, the classification of text expressions in source materials into facts and opinions is done by the sentiment analysis. Facts are the objective expressions regarding the events and their attributes as well as entities. The opinions are the subjective expressions of sentiments, emotions, feelings, events and attributes, and attitudes. This must be specified that not all the objective sentences include no opinions and not all subjective sentences include opinions. Thus, for sentiment analysis, it is significant for recognizing and extracting the facts and opinions from source materials. However, this seems to be quite complex for attaining precisely. In recent times, important approaches are related to machine learning, rule-based, and the combination of both techniques. Machine learning models consist of conventional approaches like deep learning and conditional random field approaches, whereas the rule-based models consist of lexicon-based approach. Object detection ,network optimization , image recognition , system security ,sensor networks and transportation are based on deep learning methods, which are mostly utilized in different fields. Several researchers have combined deep learning as well as machine learning algorithms into text sentiment analysis by sentiment lexicon formulation and best results are obtained .

The main aim of the sentiment lexicon-based model is to develop a sentiment lexicon, which is done by choosing suitable negative words, sentimental words, and degree adverbs. For the constructed sentiment lexicon, sentimental polarity and intensity are marked. Once the text is given as input, the words are matched with the sentiment words present in the sentiment lexicon, and those words are weighted and added for acquiring the input text's sentiment value, thus the determination of sentimental polarity is done as per the sentiment value. However, there are few approaches for acquiring the features of word vector related to the text like Glove, Word2Vec, and FastText automatically. However, the conventional machine learning models still require the emotional feature extraction of the structured information from the input text by text vectorization, human intervention, and later that algorithms are utilized for categorizing the sentiment of the text features . The main contributions of this paper are portrayed as follows.

- To undergo a critical review of sentiment analysis under different applications.

- To carry out the detailed review of various sentiment analysis models based on the machine learning algorithms, types of data, tools, and different performance measures.
- To formulate the valuable research gaps and challenges based on the existing contributions under sentiment analysis.

The review on sentiment analysis classification is designed in the following manner: Section II specifies the literature review on conventional sentimental analysis in social media. Section III describes various machine learning algorithms for sentiment analysis along with performance measures. The analysis on different types of data used and tools for sentiment analysis is given in Section IV. The research gaps and challenges of sentiment analysis using machine learning algorithms are shown in Section V. Section VI specifies the conclusion of the entire paper.

CHAPTER 2

LITERATURE SURVEY

Sentiment analysis used recently gained a lot of popularity in the fields of text mining and computational linguistics. Classifying the data is a fundamental effort in sentiment analysis. the document, phrase, or feature/aspect level polarity of a given text. Five key steps make up the sentiment analysis method. They are data gathering, text preparation, sentiment categorization, sentiment detection, and output display.

- Fang and Zhan (2015) has conducted sentimental analysis on the data from product reviews, using a Naive Bayes classifier to extract subjective content and address the problem of polarity categorisation. Amazon's online product reviews dataset has been used in this study's research.The disadvantage of this method is a Naive Bayes does not produce satisfying accuracy.
- Goyal and Parulaker (2015) analysed text-based movie reviews by counting the occurrences of each term using a random forest classifier. Sentiment analysis can be used to determine the reviewer's emotional state, such as whether they were "happy," "sad," "angry," or soon. Examine the tone of a few critics' evaluations of different movies to ascertain if they believed the movie was good or awful. the relationships among the words in the review to predict its overall polarity."
- Rahul Wadbude and his team (Wadbude et al., 2016) used The field of natural language processing has recently paid a lot of attention to fine-grained sentiment analysis of text reviews. The majority of the early work was concentrated on developing effective feature representations of the classification-related text reviews. During fine-grained sentiment categorization, the algorithms typically disregard other common criteria like user identity, product identity, and help fullness rating.
- According to Sharma, Chakraborti, and Jha (2019), online shopping has gained global popularity over the past decade. This dramatic trend's primary reasons include the ease of internet access, availability of smartphones, increased awareness of e-commerce, and increased access to online shopping applications. Online shopping platforms such as Amazon enable customers to shop with convenience, save time, and get their products delivered at home within the shortest time possible. Competitive businesses prefer e-commerce

over physical stores for the potential to reach more customers around the world (Sharma, Chakraborti and Jha, 2019).

- In 2019, Afzaal et al. have recommended a novel approach of aspect-based sentiment classification, which recognized the features in a precise manner and attained the best classification accuracy. Moreover, the scheme was developed as a mobile application, which assisted the tourists in identifying the best hotel in the town, and the proposed model was analyzed using the real-world data sets. The results have shown that the presented model was effective in both recognition as well as classification.
- In 2019, Feizollah et al. have concentrated on tweets related to two halal products such as halal cosmetics and halal tourism. By utilizing Twitter search function, Twitter information was extracted, and a new model was employed for data filtering. Later, with the help of deep learning models, a test was performed for computing and evaluating the tweets. Moreover, for enhancing the accuracy and building prediction methods, RNN, CNN, and LSTM were employed. From the outcomes, it was seemed that the combination of LSTM and CNN attained the best accuracy.
- In 2018, Mukhtar et al. have performed the sentiment analysis to the Urdu blogs attained from several domain with Supervised Machine learning and Lexicon-based models. In Lexicon-based models, a well-performing Urdu sentiment analyzer and an Urdu Sentiment Lexicons were employed, whereas, in Supervised Machine learning algorithm, DT, KNN, and SVM were employed. The data were combined from the two soruces for performing the best sentiment analysis. Based on the tests conducted, the outcomes were shown that the Lexicon-based model was superior to the supervised machine learning algorithm.
- In 2020, Kumar et al. have presented a hybrid deep learning approach named ConVNet-SVMBoVW that dealt with the real-time data for predicting the fine-grained sentiment. In order to measure the hybrid polarity, an aggregation model was developed. Moreover, SVM was used for training the BoVW to forecast the sentiment of visual content. Finally, it was concluded that the suggested ConvNet-SVMBoVW was outperformed by the conventional models.
- In 2018, Abdi et al. have proffered a machine learning technique for summarizing the opinions of the users mentioned in reviews. The suggested

method merged multiple kinds of features into a unique feature set for modelling accurate classification model. Therefore, a performance investigation was done for four best feature selection models for attaining the best performance and seven classifiers for choosing the relevant feature set and recognized an effective machine learning algorithm. The suggested method was implemented in various datasets. The outcomes have demonstrated that the combination of IG as the feature selection approach and SVM-based classification approach enhanced the performance.

- In 2019, Ray and Chakrabarti [have introduced a deep learning algorithm for extracting the features from text and the user's sentiment analysis with respect to the feature. In opinionated sentences, a seven layer Deep CNN was employed for tagging the features. In order to enhance the performance of sentiment scoring and feature extraction models, the authors merged the deep learning methods using a set of rule-based models. Finally, it was seen that the suggested method achieved the best accuracy. In 2019, Zhao et al. have offered a novel image-text consistency driven multi-modal sentiment evaluation model, which explored the correlation among the text and image. Later, a multi-modal adaptive sentiment analysis model was implemented. By using the traditional SentiBank model, the mid-level visual features were extracted and those were employed for representing the visual theories by integrating the different characteristics like social, textual, and visual features for introducing a machine learning model. The suggested model has attained best performance when compared over traditional models.
- In 2019, Park et al. have developed a semi-supervised sentiment-discriminative objective for resolving the issue by documents partial sentiment data. The suggested model not only reflected the partial data, but also secured the local structures obtained from real data. The suggested model was evaluated on real time datasets. The results have shown that the suggested model was performing well. In 2019, Vashishtha and Susan have calculated the sentiment related to social media posts by a new set of fuzzy rules consisting of many datasets and lexicons. The developed model combined Word Sense Disambiguation and NLP models with a new unsupervised fuzzy rule-based model for categorizing 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 outcomes have shown that the introduced method was attaining the best results.

- In 2019, Yousif et al. have presented a multi-task learning method on the basis of CNN and RNN. The structure of the suggested method was helpful for denoting the citation context and feature extraction was done in an automatic way. By considering two freely accessible datasets, the suggested technique was analyzed. The outcomes have shown that the proposed model was improved than conventional models. In 2020, Hassonah et al. have recommended hybrid machine learning algorithm for improving the sentiment analysis, because a classification approach was built on the basis of "Positive, Negative, and Neutral" classes with SVM classifier, at the same time two feature selection methods were merged by the MVO and Relief models. Moreover, Twitter data was employed for evaluating the proposed model. The experimental results indicated that the suggested technique was performing well than conventional techniques.

2.1 INFERENCES FROM LITREATURE SURVEY

- Inferences from the literature Survey is that the publisher have developed a model with the Multinominal Naive Bayes Algorithm where we can’t assure the accurate, predicted results as we excepted.
- So since instead of using Multinominal Naive Bayes single model we are going to implement the different Machine learning Models to get more accurate predictions.
- Considering the isolated test model to test in the different popular algorithm like Logistic Regression, Support Vector Machine, Decision Tree, Random forest, comparing all these test result we are going to suggest the bases production with 99.9% accuracy as excepted.

2.2 OPEN PROBLEMS IN EXISTING METHOD

Existing work of the project is that the creator of the Model with all the process like he used Data Exploration, Correlation, Sentimental Analysis and for the implements the machine learning models he used only the one Algorithm model i.e. Multinominal Naive Bayes.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDIES/RISK ANALYSIS OF THE PROJECT

The project plan and objective of the project is clear and this project is implemented till application level without any problem. Also, the risk involved in this project is very minimal which is negligible. This project can further be carried out in next phase as well it has wide scope. As currently four classification algorithms are used in project. The accuracy and performance of each algorithm is used further to predict with the client-side data and it can also be improved in future. The applications is built with minimal risk. So, from this we can analyze that the project and feasible and it is submitted on time. The theme of the feasibility study is to find out the application is technical, economical, operational and market feasible so that applications sustain for longer time and gives return on investment with good results. It can be used to inform future decisions.

Technical Feasibility:

This involves assessing whether the required machine learning algorithms and technologies are available and whether they can be developed within the constraints of the project.

Economic Feasibility:

This involves analyzing the cost and benefits of the machine learning project, including the cost of data acquisition, model training, and deployment. It also involves assessing the potential return on investment.

Operation Feasibility:

This involves assessing whether the machine learning model can be effectively integrated into existing operations and whether the necessary resources and expertise are available to operate and maintain the model.

Market Feasibility:

This involves analyzing the market for the machine learning model, including the 10 demands for the product, competition, and potential customers.

3.2 SOFTWARE REQUIREMENT SPECIFICATION DOCUMENT

It is crucial to comprehend and record all forms of requirements to guarantee that the software system or application is created in a way that meets the expectations of the users and stakeholders and operates as anticipated within its designated environment. A project typically encompasses three types of requirements, namely functional requirements, non-functional requirements, and environmental requirements.

3.2.1 Functional Requirements

It describes what the software system or application should do, including its features, capabilities, and functionality. They typically specify the tasks that the system needs to perform, the inputs and outputs, and the behavior of the system under different conditions. Examples of functional requirements include login and authentication, data input and validation, search and filtering, reporting and analytics, and workflow management.

3.2.2 Non-Functional Requirements

It describes how the software system or application should perform, including its performance, reliability, security, and usability. They typically specify the quality attributes of the system, such as availability, response time, scalability, maintainability, and portability. Examples of non-functional requirements include performance requirements such as response time, reliability requirements such as fault tolerance and disaster recovery, security requirements such as authentication and authorization, and usability requirements such as accessibility and user experience.

3.2.3 Environmental Requirements

It describes the environment in which the software system or application will operate, including the hardware, software, and network infrastructure. They typically specify the resources required to deploy and run the system, such as hardware specifications, operating systems, and software dependencies. Examples of environmental requirements include the operating system, 11 database management system, web server, network protocols, and required software libraries. An operating system is windows7+ or macOS 10.6+, anaconda and vs-code are the software requirement of this project to run it uninterruptable. As far as hardware is concerned a good SSD is recommended to run it smoothly but HDD is also fine to run the program. Minimum of 4GB

RAM and 128GB storage is required.

3.3 LIBRARIES AND SOFTWARE USED

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python libraries are pre-written modules of code that can be imported into a Python script to provide additional functionality or to simplify complex tasks. Python is known for its extensive library of modules and packages, which cover a wide range of tasks, from scientific computing and data analysis to web development and machine learning. Libraries used in project are discussed below.

NumPy: It is an open-source Python library that's used in almost every field of science and engineering. It's the universal standard for working with numerical data in Python, and it's at the core of the scientific Python and PyData ecosystems.

Pandas: It is a popular open-source library for data manipulation and analysis in Python. It provides tools for working with structured data, such as tabular data and time series data.

Matplotlib: It is a popular open-source library for creating static, animated, and interactive visualizations in Python. It provides a wide range of tools for creating various types of plots, such as scatter plots, bar charts, histograms, and more.

Seaborn: It is a popular open-source library for statistical data visualization in Python. It is built on top of the matplotlib library and provides a higher-level interface for creating more attractive and informative visualizations.

Scikit-learn: It is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling 12 including classification, regression, clustering and dimensionality reduction in Python.

Anaconda Navigator: Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda® Distribution that allows you to launch applications and manage conda packages, environments, and channels without using command line interface

(CLI) commands. It is available for Windows, macOS, and Linux.

Jupyter Notebook: It is a web-based interactive computing environment that allows users to create and share documents that contain live code, equations, visualizations, and narrative text.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

Existing work of the project is that the creator of the Model with all the process like he used Data Exploration, Correlation, Sentimental Analysis and for the implements the machine learning models he used only the one Algorithm model i.e. Multinominal Naive Bayes. The Proposed methodology for the model is Know the reviews of Product using the comments they gave. The purpose of this model to Analysis the bulk of comments and categorize the products according to their unique id and also we Data Exploration for the model in that we did the process for the Training dataset.

4.1 SELECTED METHODOLOGY OR PROCESS MODEL:

There are different methodologies and process models that can be used for a project involving sentiment analysis of Amazon reviews. However, one commonly used approach is the following:

- Problem definition: Define the problem statement and objectives of the project, such as identifying the sentiment of Amazon reviews to gain insights into customer opinions and preferences.
- Data collection: Gather relevant data, such as Amazon reviews, from various sources and store them in a suitable format.
- Data preprocessing: Clean and preprocess the data to remove noise, such as irrelevant words, punctuation, and stop words, and convert it into a suitable format, such as a bag of words or a TF-IDF matrix.
- Model selection: Select an appropriate machine learning model or algorithm for sentiment analysis, such as Naive Bayes, Support Vector Machines (SVM), or Recurrent Neural Networks (RNN).
- Model training: Train the selected model on the preprocessed data and fine-tune its hyperparameters to achieve optimal performance.
- Model evaluation: Evaluate the performance of the trained model using appropriate metrics, such as accuracy, precision, recall, and F1 score.
- Model deployment: Deploy the trained model into a production environment to analyze new Amazon reviews and generate sentiment scores.
- Feedback and improvement: Continuously monitor and evaluate the

performance of the deployed model and collect feedback from users to identify areas for improvement and refine the model accordingly.

Overall, this process model follows a typical machine learning pipeline for sentiment analysis and can be customized based on the specific requirements and constraints of the project.

4.2 Architecture / Overall Design of Proposed System

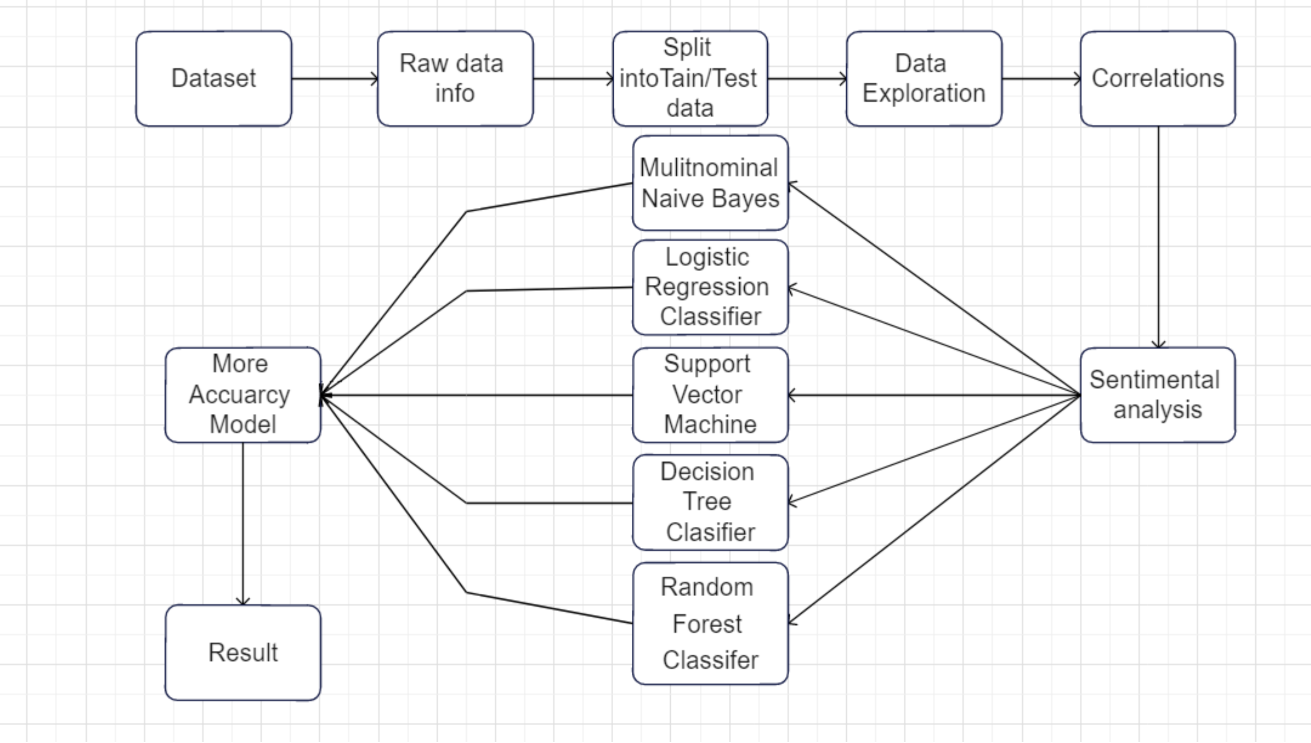


Fig 4.1. System Architecture of Model

4.3 Description of Software for Implementation and Testing plan of the Proposed Model/System

4.3.1 PYTHON

Python was created by Guido van Rossum, and first released on February 20, 1991. While you may know the python as a large snake, the name of the Python programming language comes from an old BBC television comedy sketch series called Monty Python's Flying Circus.

Python has syntax that allows developers to write programs with fewer lines than some other programming languages. Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

Python is a dynamic, interpreted (bytecode-compiled) language. There are no type declarations of variables, parameters, functions, or methods in source code. This makes the code short and flexible, and you lose the compile-time type checking of the source code.

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics developed by Guido van Rossum. It was originally released in 1991. Designed to be easy as well as fun, the name "Python" is a nod to the British comedy group Monty Python.

Python is commonly used for developing websites and software, task automation, data analysis, and data visualization. Since it's relatively easy to learn, Python has been adopted by many non-programmers such as accountants and scientists, for a variety of everyday tasks, like organizing finances.

A function is a block of code which only runs when it is called. You can pass data, known as parameters, into a function. A function can return data as a result.

Python is used by Intel, IBM, NASA, Pixar, Netflix, Facebook, JP Morgan Chase, Spotify, and a number of other massive companies. It's one of the four main languages at Google, while Google's YouTube is largely written in Python. Same with Reddit, Pinterest, and Instagram.

There are four basic data types supported in Python. Variables are named identifiers for any of these four types of values stored.

Python is a popular language for web and software development because you can create complex, multi-protocol applications while maintaining concise, readable syntax. In fact, some of the most popular applications were built with Python.

4.3.2 MACHINE LEARNING

Machine learning (ML) is **a branch of artificial intelligence (AI) that enables computers to “self-learn” from training data and improve over time, without being explicitly programmed**. Machine learning algorithms are able to detect patterns in data and learn from them, in order to make their own predictions.

There are four basic approaches:supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. The type of algorithm data scientists choose to use depends on what type of data they want to predict.

Machine learning is an application of AI that enables systems to learn and improve from experience without being explicitly programmed. Machine learning focuses on developing computer programs that can access data and use it to learn for themselves.

There is one crucial reason why data scientists need machine learning, and that is: 'High-value predictions that can guide better decisions and smart actions in real-time without human intervention.

The purpose of machine learning is to discover patterns in your data and then make predictions based on often complex findings to answer business questions, detect and analyses trends and help solve problems.

It is the subset of Artificial Intelligence, and we all are using this either knowingly or unknowingly. For example, we use Google Assistant that employs ML concepts, we take help from online customer support, which is also an example of machine learning, and many more.

Machine learning is the form of Artificial Intelligence that deals with system programming and automates data analysis to enable computers to learn and act through experiences without being explicitly programmed.

Machine Learning is easily one of humanity's best allies by enabling businesses to make more informed decisions, helping developers look at problems in innovative ways, and offering insights round the clock with inhuman speeds and accuracy.

Training is the most important part of Machine Learning. Choose your features and hyper parameters carefully. Machines don't take decisions, people do. Data cleaning is the most important part of Machine Learning.

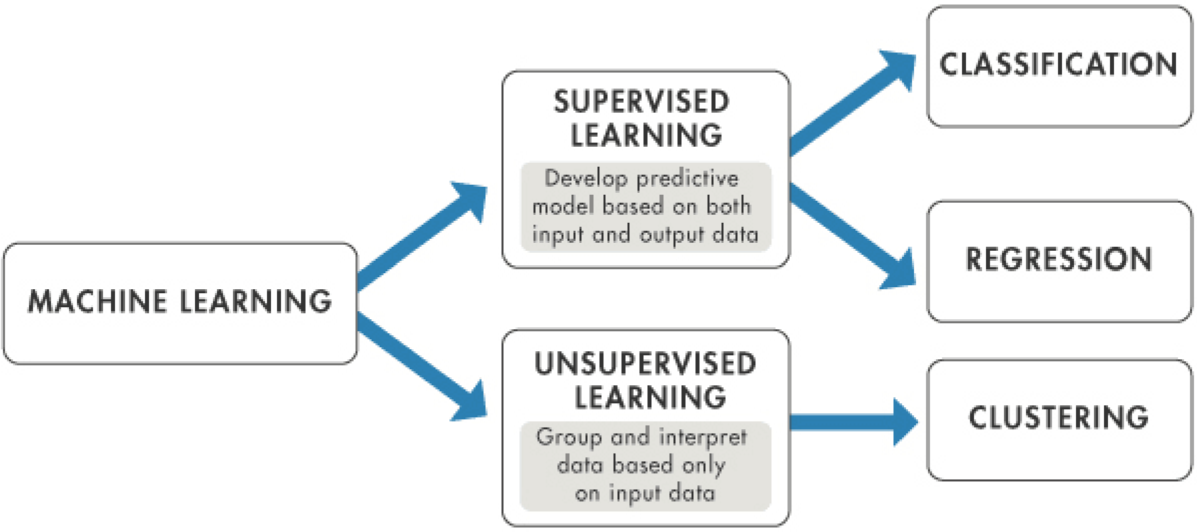

MACHINE LEARNING
SUPERVISED LEARNING
Develop predictive model based on both input and output data
UNSUPERVISED LEARNING
Group and interpret data based only on input data
CLASSIFICATION
REGRESSION
CLUSTERING

Fig 4.2. Machine learning

4.3.2.1 SUPERVISED MACHINE LEARNING

Supervised learning, also known as supervised machine learning, is **a subcategory of machine learning and artificial intelligence**. It is defined by its use of labeled datasets to train algorithms that to classify data or predict outcomes accurately.

Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.

It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process. The goal of supervised learning is to build an artificial system that can learn the mapping between the input and the output, and can predict the output of the system given new inputs.

An advantage of supervised learning is its ability to collect data or produce a data output from the previous experience. A disadvantage of the model is that decision boundary might be overstrained if your training set doesn't have examples that you want to have in a class.

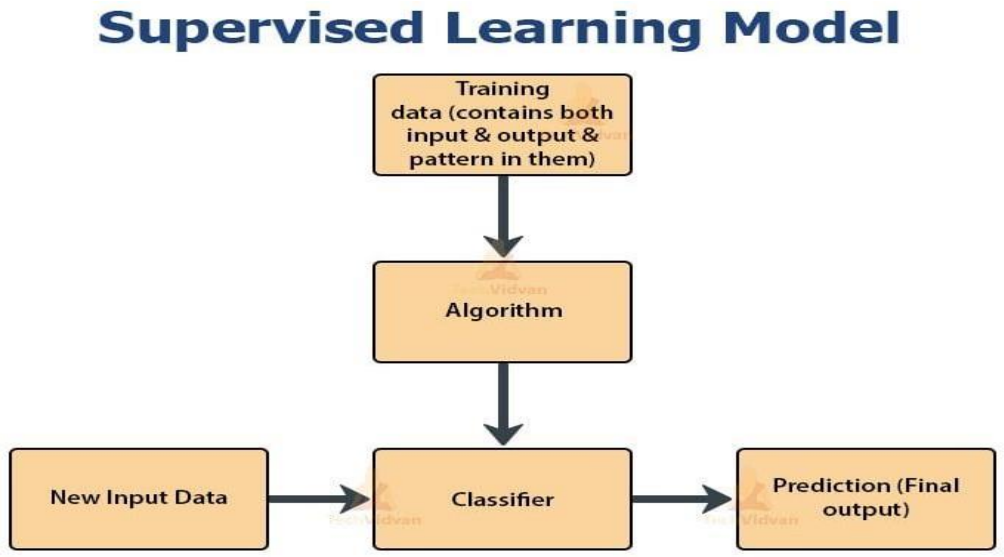

Supervised Learning Model
Training data (contains both input & output & pattern in them)
Algorithm
New Input Data
Classifier
Prediction (Final output)

Fig 4.3. Supervised Learning

4.3.2.2 UNSUPERVISED MACHINE LEARNING

Unsupervised learning, also known as unsupervised machine learning, **uses machine learning algorithms to analyze and cluster unlabeled datasets**. These algorithms discover hidden patterns or data groupings without the need for human intervention.

Unsupervised learning is when it can provide a set of unlabelled data, which it is required to analyze and find patterns inside. The examples are dimension reduction and clustering. Unsupervised Learning draws inferences from datasets without labels. It is best used if you want to find patterns but don't know exactly what you're looking for. This makes it useful in cybersecurity where the attacker is always changing methods.

We can think of unsupervised learning problems as being divided into two categories: clustering and association rules. Clustering is an unsupervised learning technique, which groups unlabeled data points based on their similarity and differences.

An example of unsupervised machine learning would be a case where a supermarket wants to increase its revenue. It decides to implement a machine learning algorithm on its sold products' data. It was observed that the customers who bought cereals more often tend to buy milk or those who buy eggs tend to buy bacon.

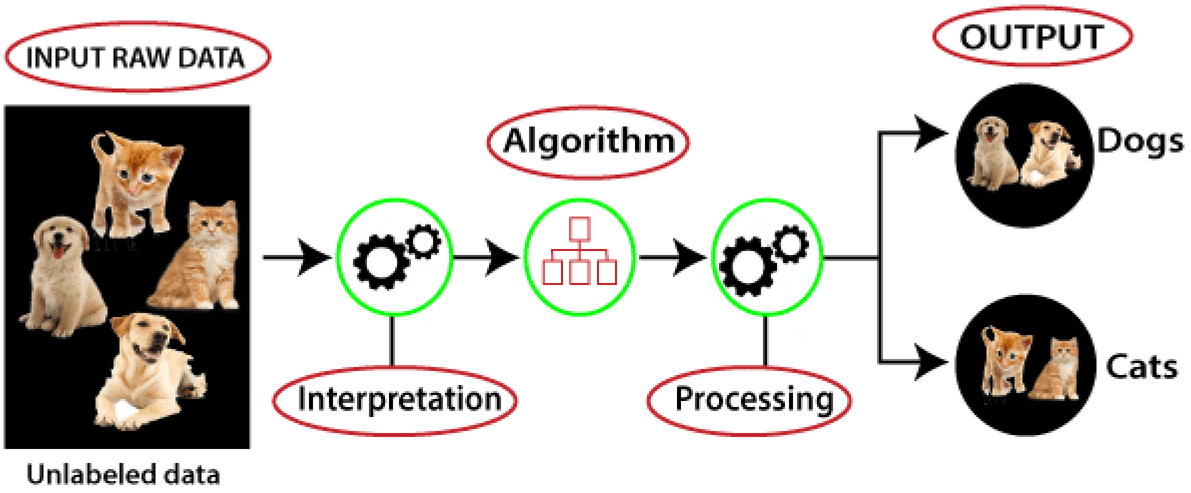


Fig 4.4. Unsupervised Machine Learning Model

4.3.3 SENTIMENATL ANALYSIS

Sentiment analysis, also referred to as opinion mining, is **an approach to natural language processing (NLP) that identifies the emotional tone behind a body of text**. This is a popular way for organizations to determine and categorize opinions about a product, service or idea.

Sentiment analysis is the process of using natural language processing, text analysis, and statistics to analyze customer sentiment. The best businesses understand the sentiment of their customers—what people are saying, how they're saying it, and what they mean.

Some other implementations use more classes or grades between Positive, Negative and Neutral (0–5 stars, 0–10 grade). Historically, it is considered that sentiment analysis started in early 2000's with the articles published by Bo Pang and Lillian Lee and by Peter Turney.

Sentiment analysis is a machine learning tool that analyzes texts for polarity, from positive to negative. By training machine learning tools with examples of emotions in text, machines automatically learn how to detect sentiment without human input.

Sentiment analysis (or opinion mining) is a natural language processing (NLP) technique used to determine whether data is positive, negative or neutral. Sentiment analysis is often performed on textual data to help businesses monitor brand and product sentiment in customer feedback, and understand customer needs.

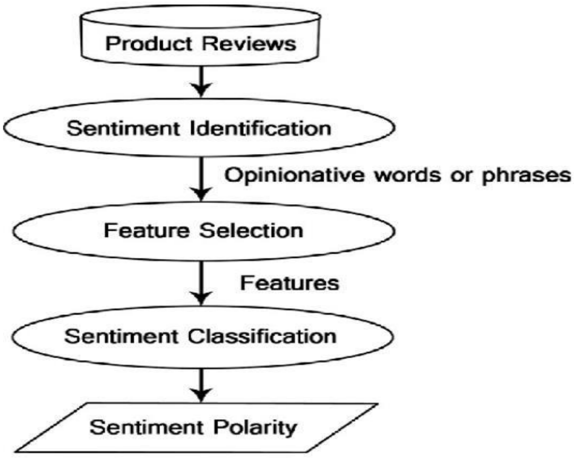


Fig 4.5. Sentimental Analysis

4.3.4 NUMPY

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

NumPy (Numerical Python) is an open source Python library that's used in almost every field of science and engineering. It's the universal standard for working with numerical data in Python, and it's at the core of the scientific Python and PyData ecosystems.

In Python we have lists that serve the purpose of arrays, but they are slow to process.NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.Arrays are very frequently used in data science, where speed and resources are very important.

Numeric, the ancestor of NumPy, was developed by Jim Hugunin. Another package Numarray was also developed, having some additional functionalities. In 2005, Travis Oliphant created NumPy package by incorporating the features of Numarray into Numeric package. There are many contributors to this open source project.

Operations using NumPy

Using NumPy, a developer can perform the following operations −

- Mathematical and logical operations on arrays.
- Fourier transforms and routines for shape manipulation.
- Operations related to linear algebra. NumPy has in-built functions for linear algebra and random number generation.

NumPy is often used along with packages like SciPy (Scientific Python) and Mat−plotlib (plotting library). This combination is widely used as a replacement for MatLab, a popular platform for technical computing. However, Python alternative to MatLab is now seen as a more modern and complete programming language.

The best way to enable NumPy is to use an installable binary package specific to your operating system. These binaries contain full SciPy stack (inclusive of NumPy, SciPy, matplotlib, IPython, SymPy and nose packages along with core Python).

4.3.5 PANDAS

Pandas is **a Python library used for working with data sets**. It has functions for analyzing, cleaning, exploring, and manipulating data. The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.

Pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labeled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real-world data analysis in Python.

Pandas were initially developed by Wes McKinney in 2008 while he was working at AQR Capital Management. He convinced the AQR to allow him to open source the Pandas. Another AQR employee, Chang She, joined as the second major contributor to the library in 2012. Over time many versions of pandas have been released. The latest version of the pandas is 1.5.3, released on Jan 18, 2023.

The first step of working in pandas is to ensure whether it is installed in the Python folder or not. If not then we need to install it in our system using pip command. Type cmd command in the search box and locate the folder using cd command where python-pip file has been installed. After locating it, type the command:

Pandas Series is a one-dimensional labeled array capable of holding data of any type (integer, string, float, python objects, etc.). The axis labels are collectively called indexes. Pandas Series is nothing but a column in an excel sheet. Labels need not be unique but must be a hashable type. The object supports both integer and label-based indexing and provides a host of methods for performing operations involving the index.

Pandas DataFrame is a two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns. Pandas DataFrame consists of three principal components, the data, rows, and columns.

4.3.6 Matplotlib

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

Matplotlib comes with a wide variety of plots. Plots helps to understand trends, patterns, and to make correlations. They’re typically instruments for reasoning about quantitative information. Some of the sample plots are covered here.

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible. Create publication quality plots. Make interactive figures that can zoom, pan, update.

Matplotlib is a multi-platform data visualization library built on NumPy arrays, and designed to work with the broader SciPy stack. It was conceived by John Hunter in 2002, originally as a patch to IPython for enabling interactive MATLAB-style plotting via gnuplot from the IPython command line.

Matplotlib is extremely powerful because it allows users to create numerous and diverse plot types. It can be used in variety of user interfaces such as IPhython shells, Python scripts, Jupyter notebooks, as well as web applications and GUI toolkits. It has support for LaTeX-formatted labels and texts.

It allows the decision-makers to make decisions very efficiently and also allows them in identifying new trends and patterns very easily. It is also used in high-level data analysis for Machine Learning and Exploratory Data Analysis (EDA).

Matplotlib is useful for transforming statistical analyses and operations into visually interesting findings. Similar to other open-source data science tools, the Matplotlib library also has an active community of Python developers and users that regularly make contributions to the library.

matplotlib is a Python-based plotting library with full support for 2D and limited support for 3D graphics, widely used in the Python scientific computing community. The library targets a broad range of use cases.

The matplotlib function imshow() creates an image from a 2-dimensional numpy array. The image will have one square for each element of the array. The color of each square is determined by the value of the corresponding array element and the color map used by imshow() .

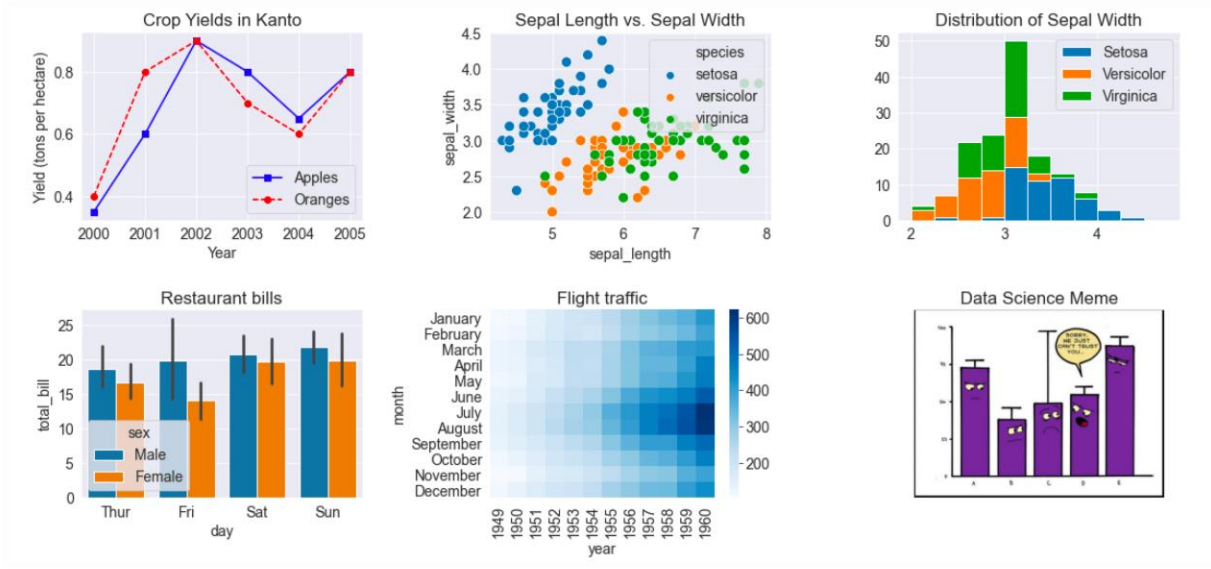


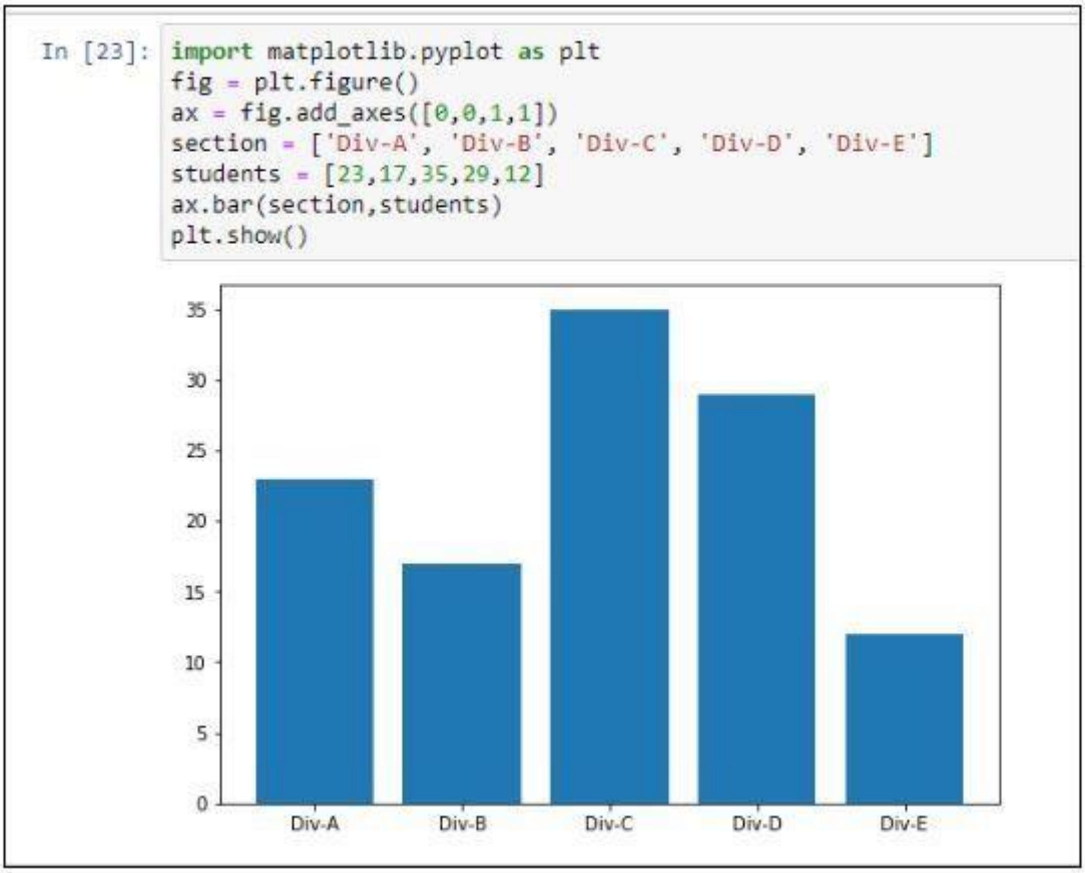
Fig 4.6.Matplotlib

4.3.6.1 Bar Graph

The purpose of a bar graph is to convey relational information quickly in a visual manner. The bars display the value for a particular category of data. The vertical axis on the left or right side of the bar graph is called the y-axis. The horizontal axis at the bottom of a bar graph is called the x-axis.

A bar graph presents data with heights and lengths proportional to the values they present.

Syntax: ax.bar(x, height, width, bottom, align)



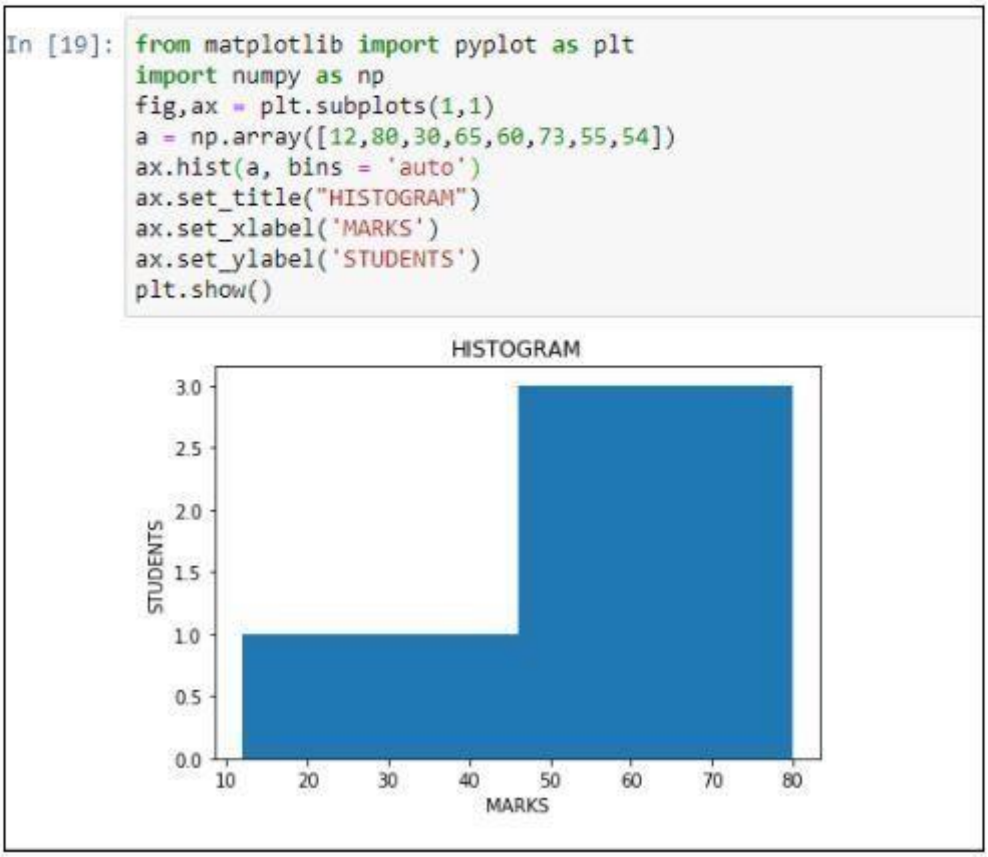
```
In [23]: import matplotlib.pyplot as plt
         fig = plt.figure()
         ax = fig.add_axes([0,0,1,1])
         section = ['Div-A', 'Div-B', 'Div-C', 'Div-D', 'Div-E']
         students = [23,17,35,29,12]
         ax.bar(section,students)
         plt.show()
```

Fig 4.7. Bar graphs

4.3.6.2 Histograms

A histogram is a graphical representation of data points organized into user-specified ranges. Similar in appearance to a bar graph, the histogram condenses a data series into an easily interpreted visual by taking many data points and grouping them into logical ranges or bins.

A histogram is used to understand the distribution of a continuous numerical variable.



```
In [19]: from matplotlib import pyplot as plt
         import numpy as np
         fig,ax = plt.subplots(1,1)
         a = np.array([12,80,30,65,60,73,55,54])
         ax.hist(a, bins = 'auto')
         ax.set_title("HISTOGRAM")
         ax.set_xlabel('MARKS')
         ax.set_ylabel('STUDENTS')
         plt.show()
```

Fig 4.8. Histogram

4.3.6.3 Scatter Plots

Scatter plots are the graphs that present the relationship between two variables in a data-set. It represents data points on a two-dimensional plane or on a Cartesian system. The independent variable or attribute is plotted on the X-axis, while the dependent variable is plotted on the Y-axis.

Scatter plots are used to represent values for two different numeric variables.

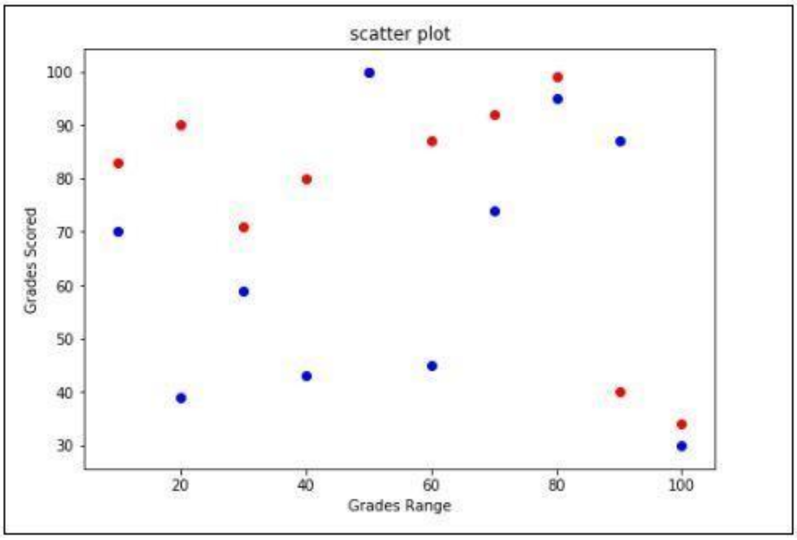
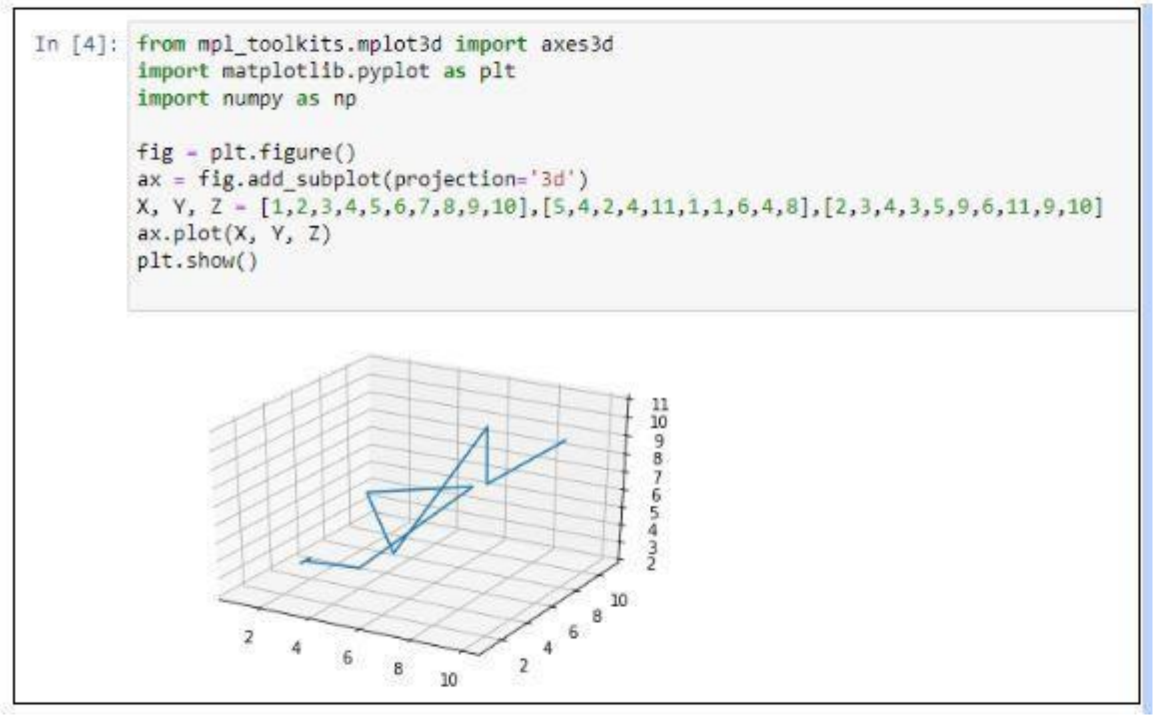


Fig 4.9. Scatter Plot

4.3.6.4 Matplotlib Three-Dimensional Plotting

Three-dimensional plots are enabled by importing the mplot3d toolkit, included with the Matplotlib package. A three-dimensional axes can be created by passing the keyword projection='3d' to any of the normal axes creation routines. We can now plot a variety of three-dimensional plot types.

Importing the mplot3d toolkit enables users to create three-dimensional plots.



```
In [4]: from mpl_toolkits.mplot3d import axes3d
        import matplotlib.pyplot as plt
        import numpy as np

        fig = plt.figure()
        ax = fig.add_subplot(projection='3d')
        X, Y, Z = [1,2,3,4,5,6,7,8,9,10],[5,4,2,4,11,1,1,6,4,8],[2,3,4,3,5,9,6,11,9,10]
        ax.plot(X, Y, Z)
        plt.show()
```

Fig 4.10.Matplotlib Three-Dimensional Plotting

4.3.7 Seaborn

Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures.Seaborn helps you explore and understand your data. Its plotting functions operate on dataframes and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets you focus on what the different elements of your plots mean, rather than on the details of how to draw them.

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. For a brief introduction to the ideas behind the library, you can read the introductory notes or the paper.

This uses the matplotlib rcParam system and will affect how all matplotlib plots look, even if you don’t make them with seaborn. Beyond the default theme, there are several other options, and you can independently control the style and scaling of the plot to quickly translate your work between presentation contexts (e.g., making a version of your figure that will have readable fonts when projected during a talk). If you like the matplotlib defaults or prefer a different theme, you can skip this step and still use the seaborn plotting functions.

Most code in the docs will use the load_dataset() function to get quick access to an example dataset. There’s nothing special about these datasets: they are just pandas data frames, and we could have loaded them with pandas.read_csv() or built them by hand. Most of the examples in the documentation will specify data using pandas data frames, but seaborn is very flexible about the data structures that it accepts.

This plot shows the relationship between five variables in the tips dataset using a single call to the seaborn function relplot(). Notice how we provided only the names of the variables and their roles in the plot. Unlike when using matplotlib directly, it wasn’t necessary to specify attributes of the plot elements in terms of the color values or marker codes. Behind the scenes, seaborn handled the translation from values in the dataframe to arguments that matplotlib understands. This declarative approach lets you stay focused on the questions that you want to answer, rather than on the details of how to control matplotlib.

There is no universally best way to visualize data. Different questions are best answered by different plots. Seaborn makes it easy to switch between different visual representations by using a consistent dataset-oriented API.

The function relplot() is named that way because it is designed to visualize many different statistical relationships. While scatter plots are often effective, relationships where one variable represents a measure of time are better represented by a line. The relplot() function has a convenient kind parameter that lets you easily switch to this alternate representation.

Seaborn has five built-in themes to style its plots: darkgrid , whitegrid , dark , white , and ticks . Seaborn defaults to using the darkgrid theme for its plots, but you can change this styling to better suit your presentation needs. To use any of the preset themes pass the name of it to sns.

Seaborn (styled as “seaborn”) is an open-source Python library used for visualizing the explorative statistical plots of data. It is a basic library used to extract various information that can provide an understanding of a dataset.

4.3.8 CORRELATION

The correlation between two variables numerically describes whether larger and smaller than average values of one variable are related to larger or smaller than average values of the other variable. It is measuring the strength and direction of a linear relationship between two variables.

Correlation is a statistical measure (expressed as a number) that describes the size and direction of a relationship between two or more variables. A correlation between variables, however, does not automatically mean that the change in one variable is the cause of the change in the values of the other variable.

In summary, correlation coefficients are used to assess the strength and direction of the linear relationships between pairs of variables. When both variables are normally distributed use Pearson's correlation coefficient, otherwise use Spearman's correlation coefficient.

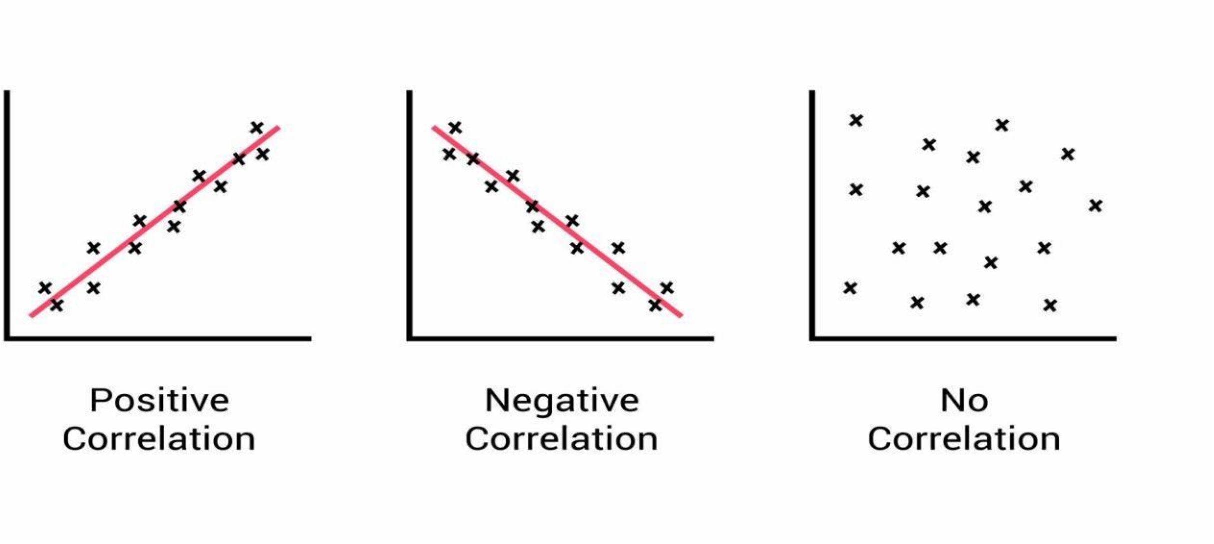


Fig 4.11. Correlation

4.4 Project Management Plan

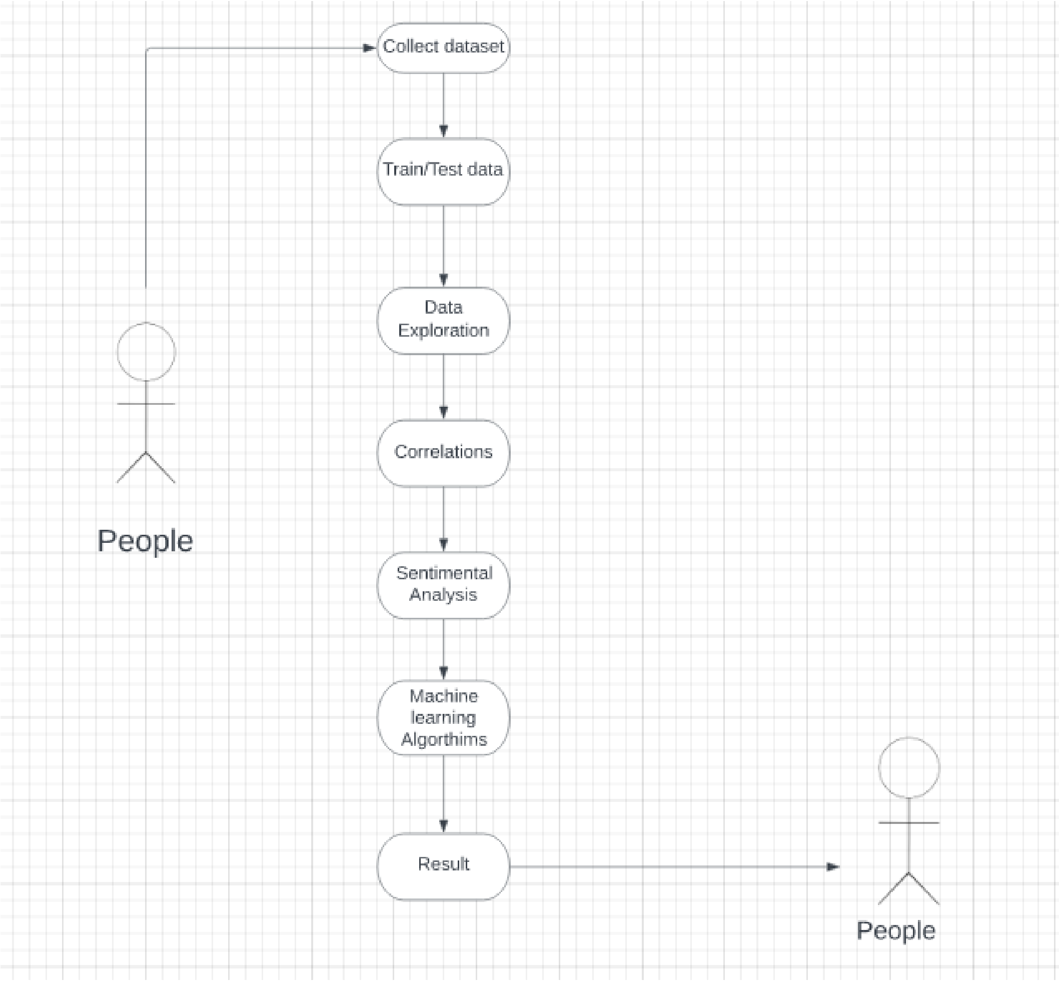


Fig 4.12. Project management plan model

4.5 FINANCIAL REPORT ON ESTIMATING COST

The process of estimating costs for a project involves identifying all the expenses required to complete the project. In the case of the project of Amazon reviews using sentiment analysis, some of the costs that may need to be considered include:

- Technology costs - this includes the cost of the hardware and software required to develop and run the sentiment analysis model.
- Labor costs - this includes the cost of the personnel required to develop and maintain the sentiment analysis model. This may include data scientists, software engineers, and project managers.

- Data costs - this includes the cost of acquiring and storing the data required to train the sentiment analysis model.
- Infrastructure costs - this includes the cost of the servers and other infrastructure required to run the sentiment analysis model.
- Miscellaneous costs - this includes other expenses such as travel expenses, marketing expenses, and legal expenses.

Once all the costs have been identified, they need to be estimated as accurately as possible. This may involve getting quotes from vendors, conducting research on the cost of similar projects, and estimating the cost based on previous experience.

Finally, all the estimated costs are added up to get the total project cost. It's important to note that the actual cost may differ from the estimated cost due to unexpected expenses, changes in project scope, and other factors. Therefore, it's important to keep track of the actual expenses throughout the project and make adjustments to the budget as necessary.

4.6 TRANSITION/ SOFTWARE TO OPERATIONS PLAN

A Transition/Software to Operations (S2O) plan for the Amazon Reviews project using sentimental analysis would typically include the following components:

- Deployment plan: This section would detail how the project will be deployed in a production environment. It would include information on the infrastructure requirements, deployment process, and any necessary configurations for the system to operate efficiently.
- Monitoring and Alerting plan: This section would define the monitoring and alerting mechanism for the system. It would include details on the metrics to monitor, how the monitoring will be done, and how alerts will be triggered and managed.
- Maintenance and Support plan: This section would describe how the system will be maintained and supported in a production environment. It would include information on the processes for handling issues, bug fixes, and feature enhancements.
- Security plan: This section would detail the security measures in place to protect the system from unauthorized access or malicious attacks. It would include information on access control, data protection, and disaster recovery.
- Training and Documentation plan: This section would define the training and

documentation requirements for operating and maintaining the system. It would include information on the training materials, support resources, and user documentation.

- Performance and Scalability plan: This section would describe the performance and scalability requirements for the system. It would include details on how the system will handle increases in load and how performance will be monitored and optimized.

Overall, the Transition/S2O plan for the Amazon Reviews project using sentimental analysis would aim to ensure that the system is deployed, maintained, and supported in a production environment with the highest level of reliability, security, and efficiency.

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 Development and Deployment Setup

The Proposed methodology for the model is Know the reviews of Product using the comments they gave. The purpose of this model to Analysis the bulk of comments and categorize the products according to their unique id and also we Data Exploration for the model in that we did the process for the Training dataset.

And also Explore the column of dataset like asins, name, reviews.rating, reviews.doRecommend, reviews.numHelpful, reviews.Text. Here we explore all the columns details to check with unique values. Further we will go for Correlation technique to Analysis the columns and we are Check sentimental analysis to know the nature of the comments like positive or negative. To deal the model for classification we use many Machine learning models like naïve bayes, Random forest, Logistic regression, Support vector machine.

Take the Kaggle dataset and analyse it. Add all required library functions to it.Open the Notebook and read the csv dataset. Using commands, obtain full dataset information. Brief the dataset after visualising it.Divide the dataset into test and training data.Start Examining the training data and the columns,Following exploration, compare every column to look for any duplicate names. Consider creating graphs for the dataset's columns with names, ASINs, reviews.rating,

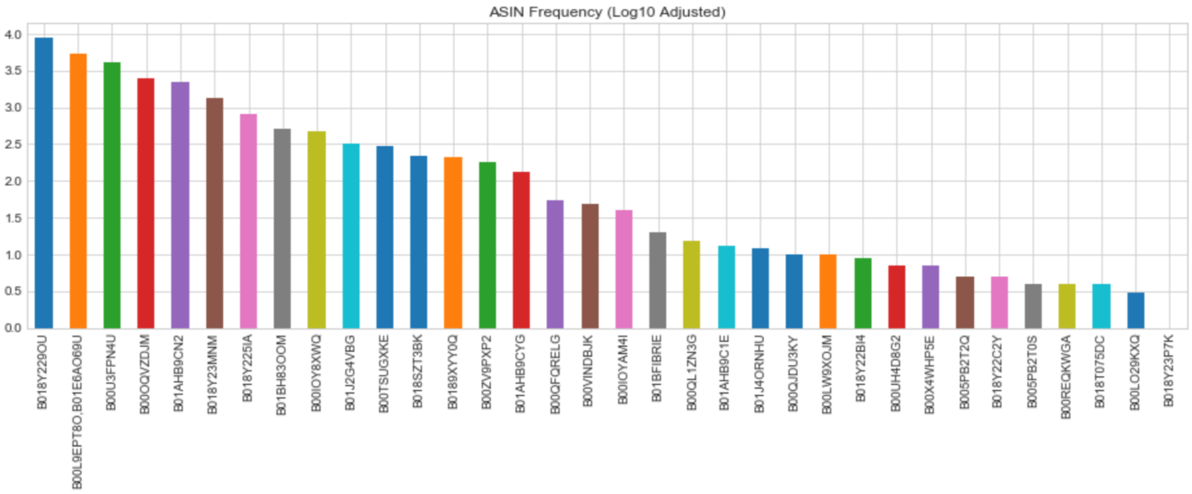


Fig 5.1. asins frequency

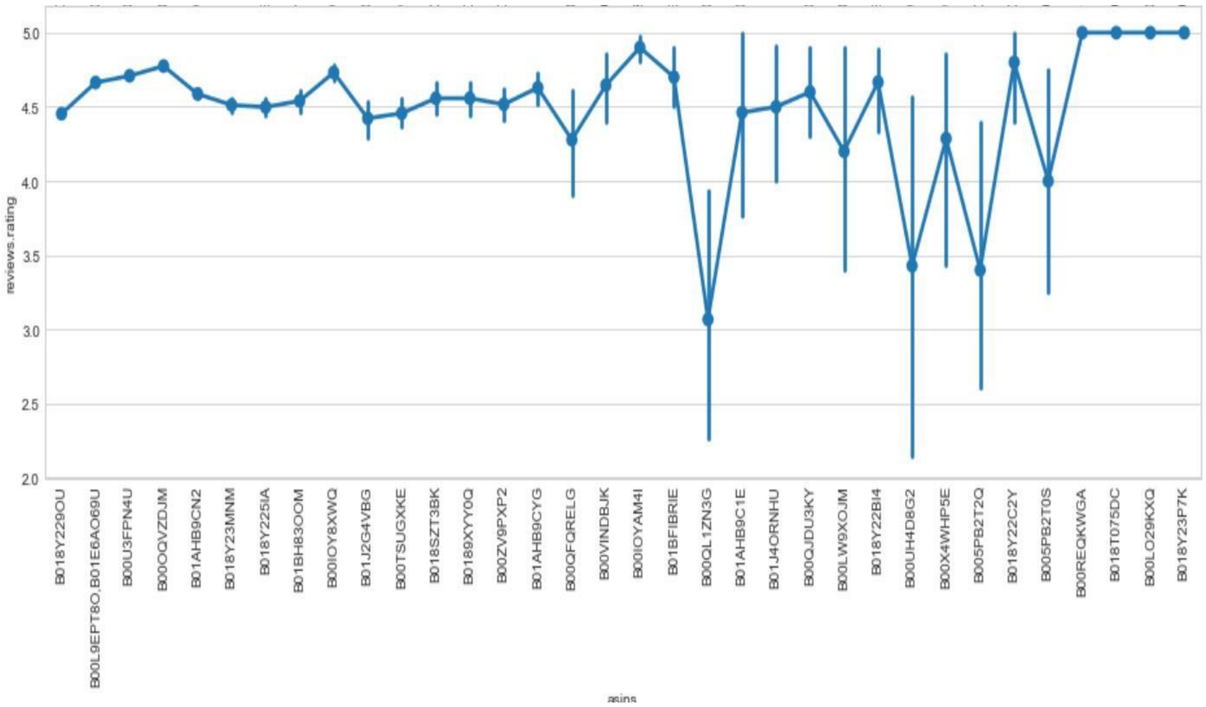


Fig 5.2. point-plot of reviews.rating

ASINs for reviews that do recommend. Continue with graphs after the representation of the graph. As a next step, we will analyse the review using the correlation technique. Using the asins column to rate. Look at review.info and do a head count. Find the mean of the column's ratings. Plot the correlation Scatter plot graph. Start your sentimental analysis now. First Set the target variable for sentiments. Check the Nature of the data and preapre data for a sentimental analysis.

The data extraction feature is now ready. We use Count Vectorizer from SciKit-Learn for the Extract feature procedure. We process the Text Preprocessing, occurrence counting, Feature vector in scikit-Count vectorizer Then Pipeline is develop for extracted Features.We use Multinomial Naive Bayes for pipeline.

After building the pipeline, now test the model with Multinomial Naïve Bayes algorithm and Check the model's accuracy. The remaining algorithms, such as the Logistic Regression Classifier, Support Vector Machine Classifier, Decision Tree Classifier, and Random Forest Classifier, are also checked for greater accuracy.

The accuracy of each method must finally be compared, and the algorithm with the highest accuracy must be used for the remaining steps.We will use the train dataset for each algorithm model and fit the data to that model. After identifying the algorithm with the highest accuracy relative to the others, a detailed analysis of that method must be done. Find the algorithm's confusion matrix. Observe the outcome.

5.2 Algorithms

5.2.1 Naive Bayes

A Naive Bayes classifier is a probabilistic machine learning model that is used for classification task. The crux of the classifier is based on the Bayes theorem. Bayes Theorem is a simple mathematical formula used for calculating conditional probabilities. In this project we have used Multinomial Naive Bayes and Bernoulli Naive Bayes.

Naive Bayes is a simple, yet effective and commonly-used, machine learning classifier. It is a probabilistic classifier that makes classifications using the Maximum A Posteriori decision rule in a Bayesian setting. It can also be represented using a very simple Bayesian network.

Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions. It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.

Naive Bayes is called naive because it assumes that each input variable is independent. This is a strong assumption and unrealistic for real data; however, the technique is very effective on a large range of complex problems.

It is a classification technique based on Bayes' Theorem with an independence assumption among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

The naive Bayes classifier is an algorithm used to classify new data instances using a set of known training data. It is a good algorithm for classification; however, the number of features must be equal to the number of attributes in the data.

It is easy and fast to predict the class of the test data set. It also performs well in multi-class prediction. When assumption of independence holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need less training data.

Naive Bayes algorithms are mostly used in **sentiment analysis, spam filtering, recommendation systems** etc. They are fast and easy to implement but their biggest disadvantage is that the requirement of predictors to be independent.

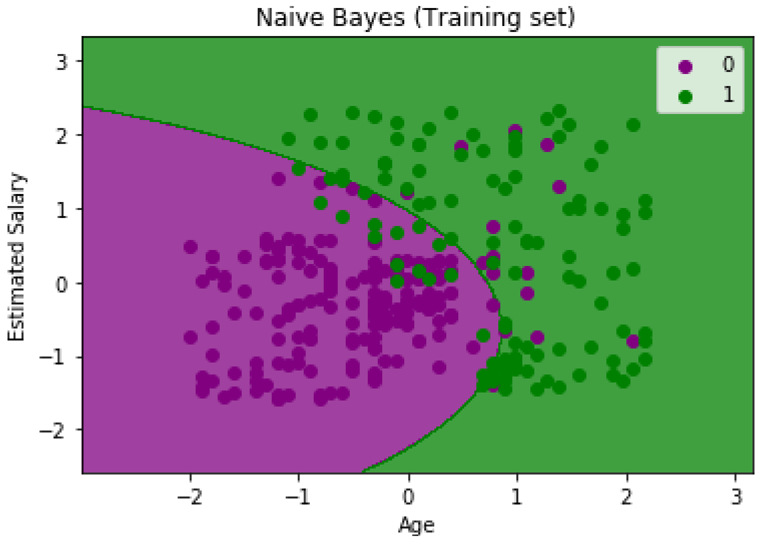


Fig 5.3. NAÏVE BAYES

5.2.2.1 Multinomial Naive Bayes

This is mostly used for document classification problem, i.e., whether a document belongs to the category of sports, politics, technology etc. The features/predictors used by the classifier are the frequency of the words present in the document. A feature vector x=(x1,….,xn) is then a histogram, with xi counting the number of times event I was observed in a 20 particular instance. This is the event model typically used for document classification, with events representing the occurrence of a word in a single document.

The multinomial naïve Bayes is widely used for assigning documents to classes based on the statistical analysis of their contents. It provides an alternative to the "heavy" AI-based semantic analysis and drastically simplifies textual data classification.

The term Multinomial Naive Bayes simply lets us know that each p(fi|c) is a multinomial distribution, rather than some other distribution. This works well for data which can easily be turned into counts, such as word counts in text.

It is a classification technique based on Bayes' Theorem with an independence assumption among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

$$P\left(\frac{B}{A}\right) = \left(\frac{P(A \cap B)}{P(A)}\right)$$

5.2.2 Logistic Regression Classifier

Logistic Regression is **a classification technique used in machine learning**. It uses logistic function to model the dependent variable. The dependent variable is dichotomous in nature, i.e. there could only be two possible classes (eg.: either the cancer is malignant or not).

It is only a classification algorithm in combination with a decision rule that makes dichotomous the predicted probabilities of the outcome. Logistic regression is a regression model because it estimates the probability of class membership as a (transformation of a) multilinear function of the features.

It is named 'Logistic Regression' because its underlying technique is quite the same as Linear Regression. The term “Logistic” is taken from the Logit function that is used in this method of classification.

Logistic regression becomes a classification technique only when a decision threshold is brought into the picture. The setting of the threshold value is a very important aspect of Logistic regression and is dependent on the classification problem itself.

Logistic Regression is a classification technique used in machine learning. It uses a logistic function to model the dependent variable. The dependent variable is dichotomous in nature, i.e. there could only be two possible classes (eg.: either the cancer is malignant or not).

Logistic Regression is one of the basic and popular algorithms to solve a classification problem. It is named 'Logistic Regression' because its underlying technique is quite the same as Linear Regression. The term “Logistic” is taken from the Logit function that is used in this method of classification.

Logistic regression is a data analysis technique that uses mathematics to find the relationships between two data factors. It then uses this relationship to predict the value of one of those factors based on the other. The prediction usually has a finite number of outcomes, like yes or no.

The Differences between Linear Regression and Logistic Regression. Linear Regression is used to handle regression problems whereas Logistic regression is used to handle the classification problems. Linear regression provides a continuous output but Logistic regression provides discreet output.

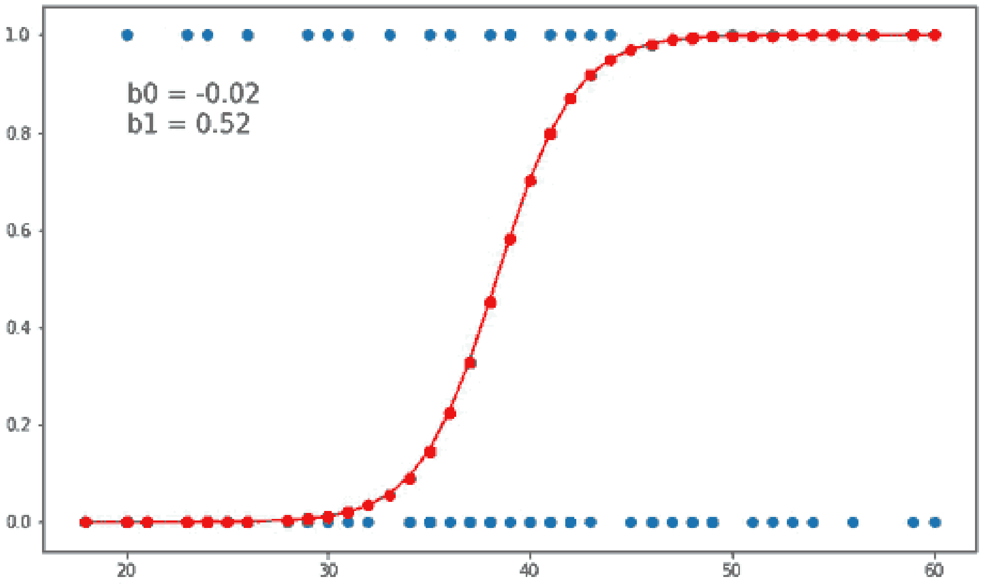


Fig 5.4. Logistic Regression Classifier

5.2.3 Support Vector Machine Classifier

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane.

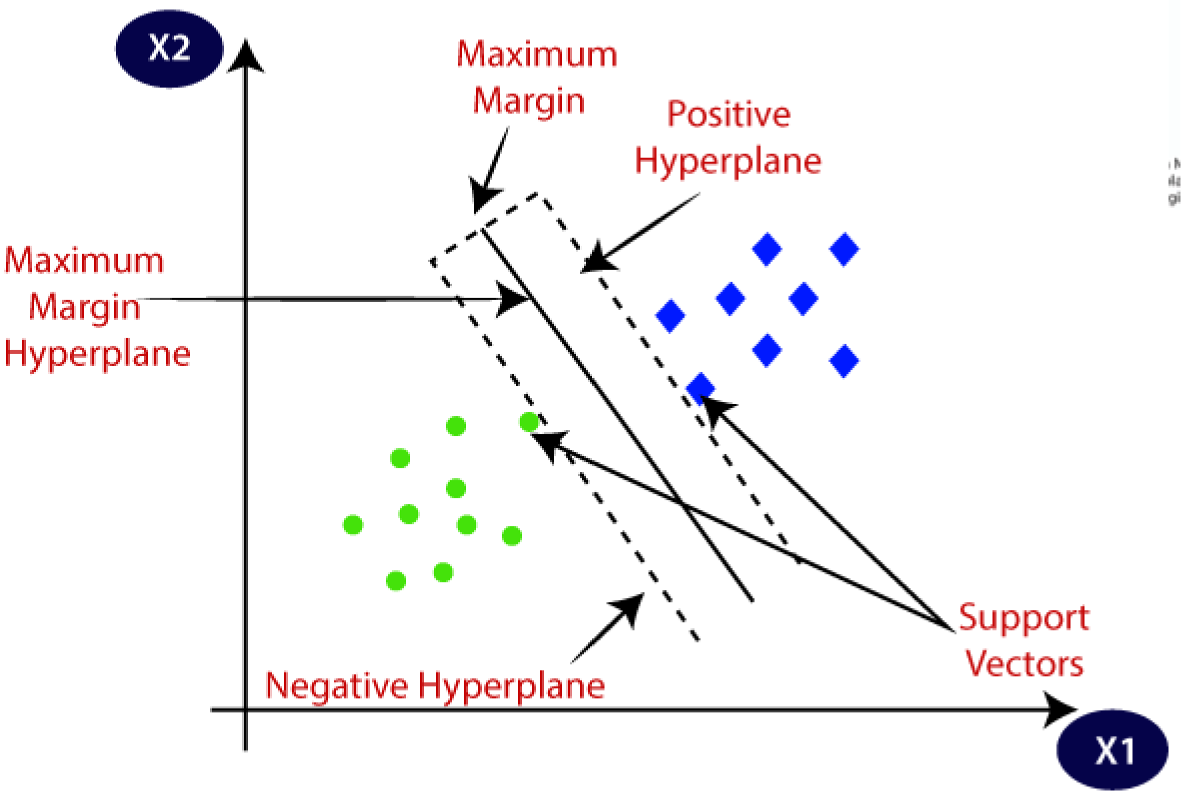


Fig 5.5. Support Vector Machine

TYPES OF SVM:

- **Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
- **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

5.2.4 Random Forest Classifier

The random forest classifier was chosen due to its superior performance over a single decision tree with respect to accuracy. It is essentially an ensemble method based on bagging. The classifier works as follows: Given D, the classifier firstly creates k bootstrap samples of D, with each of the samples denoting as Di.

A Di has the same number of tuples as D that are sampled with replacement from D. By sampling with replacement, it means that some of the original tuples of D may not be included in Di, whereas others may occur more than once. The classifier then constructs a decision tree based on each Di.

As a result, a “forest" that consists of k decision trees is formed. To classify an unknown tuple, X, each tree returns its class prediction counting as one vote. The final decision of X’s class is assigned to the one that has the most votes. The decision tree algorithm implemented in scikit-learn is CART (Classification and Regression Trees).

Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

Random Forests is a Machine Learning algorithm that tackles one of the biggest problems with Decision Trees: variance. Even though Decision Trees is simple and flexible, it is greedy algorithm. It focuses on optimizing for the node split at hand, rather than taking into account how that split impacts the entire tree.

Random forest improves on bagging because it decorrelates the trees with the introduction of splitting on a random subset of features. This means that at each split of the tree, the model considers only a small subset of features rather than all of the features of the model.

It can perform both regression and classification tasks. A random forest produces good predictions that can be understood easily. It can handle large datasets efficiently. The random forest algorithm provides a higher level of accuracy in predicting outcomes over the decision tree algorithm.

Random Forest is suitable for situations when we have a large dataset, and interpretability is not a major concern. Decision trees are much easier to interpret and understand. We take multiple decision trees in a random forest and then aggregate the result.

A decision tree combines some decisions, whereas a random forest combines several decision trees. Thus, it is a long process, yet slow. Whereas, a decision tree is fast and operates easily on large data sets, especially the linear one. The random forest model needs rigorous training.

Gradient boosting trees can be more accurate than random forests. Because we train them to correct each other's errors, they're capable of capturing complex patterns in the data. However, if the data are noisy, the boosted trees may overfit and start modeling the noise.

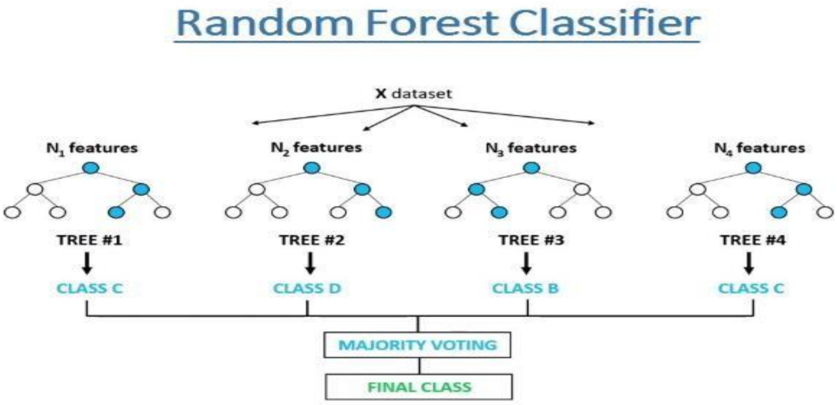


Fig 5.6. Random Forest Classifier

5.2.5 Decision Tree Classifier

Decision Tree is a **Supervised learning technique** that can be used for both classificationand Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where **internal nodes represent the features of a dataset, branches represent the decision rules** and **each leaf node represents the outcome.**

In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.

The decisions or the test are performed on the basis of features of the given dataset,It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm.A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

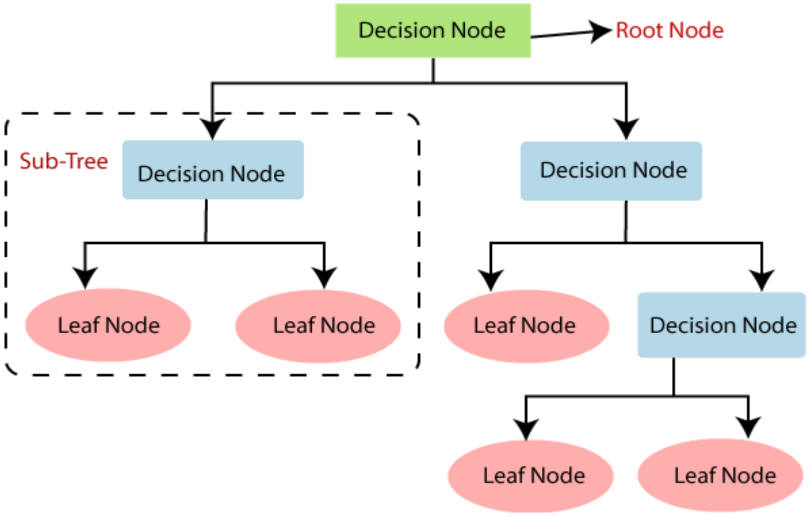


Fig 5.7. Decision Node

5.2.6 TERM FREQUENCY

Term frequency is the measurement of how frequently a term occurs within a document. The easiest calculation is simply counting the number of times a word appears. However, there are ways to modify that value based on the document length or the frequency of the most frequently used word in the document.

5.2.7 TERM FREQUENCIES TIMES INVERSE DOCUMENT FREQUENCY

TF-IDF stands for term frequency-inverse document frequency and it is a measure, used in the fields of information retrieval (IR) and machine learning, that can quantify the importance or relevance of string representations (words, phrases, lemmas, etc) in a document amongst a collection of documents.

5.3 Testing

Testing of a project for Amazon reviews using sentiment analysis would involve evaluating the accuracy and effectiveness of the sentiment analysis model used to analyze customer reviews of products on Amazon. The process of testing would involve the following steps:

- **Data Collection:** Collecting a dataset of Amazon product reviews to be used for testing the sentiment analysis model.

- **Data Preparation:** Preprocessing the dataset by removing any irrelevant information such as HTML tags, punctuation, and stop words. The data would also be transformed into a format that can be used by the sentiment analysis model.
- **Training and Testing:** Training the sentiment analysis model using a portion of the dataset and then testing the model's performance on the remaining data.
- **Performance Evaluation:** Evaluating the performance of the model by calculating metrics such as accuracy, precision, recall, and F1-score.
- **Improvement:** Improving the model's performance by adjusting the model parameters, changing the training data, or using a different algorithm.
- **Deployment:** Once the model has been tested and its performance is satisfactory, it can be deployed to analyze Amazon product reviews and provide insights to Amazon customers.

Overall, the testing of a project for Amazon reviews using sentiment analysis is a crucial step in ensuring the accuracy and effectiveness of the model used for analyzing customer reviews.

CHAPTER 6

RESULTS AND DISCUSSION

In order to obtain optimal result, each algorithm is analyzed based on five performance metrics they are accuracy, precision, recall, f1-score,confusion matrix.

Confusion Matrix:

A confusion matrix is a table that summarizes the performance of a classification model by comparing the predicted labels with the true labels. It consists of four entries: true positives, false positives, true negatives, and false negatives. The diagonal of the matrix shows the number of correct predictions, while the off-diagonal entries show the misclassifications made by the model.

Accuracy:

Accuracy measures the proportion of correct predictions made by the model.
Accuracy = (TP + TN) / (TP + TN + FP + FN)

Precision:

Precision measures the proportion of true positive predictions among all positive predictions made by the model.
Precision = TP / (TP + FP)

Recall:

Recall measures the proportion of true positive predictions among all actual positive samples in the dataset.
Recall = TP / (TP + FN)

F1 score:

F1 score is the harmonic mean of precision and recall. It is a useful metric when both precision and recall are equally important. Mathematically, F1 score is defined as:
F1 score = 2 * (precision * recall) / (precision + recall)

We will ignore the first row and column because we already replaced all NAN with " ". This is the identical circumstance as was described in the classification report above.

By looking at row 2-4 and column 2-4, which are labelled as negative, neutral, as well as positive, with scores of 246 and 104, respectively, we see that positive sentiment can occasionally be confused with neutral and negative sentiment. Even so, confusion scores of 246 and 104 for neutral as well as negative ratings, respectively, are considered negligible when compared to the total amount of significant positive sentiment at 6445.

This is also the result of a positively skewed dataset, which is consistent with our data exploration and sentiment analysis. As a result, we conclude that the products in this dataset are generally well-liked and should remain on the product roster.

Precision: determines how many objects selected were correct.

Table *6.1 Classification report of Algorithm*

	Precision	recall	f1-score	support
	0.00	0.00	0.00	5
Negative	0.67	0.25	0.36	156
Neutral	0.47	0.11	0.18	292
Positive	0.95	1.00	0.97	6473
AVG/TOTAL	0.92	0.94	0.92	6926

Confusion Matrix:

```
array([[0,0 ,0 ,4],
        [0,32,7 ,119],
        [0,14,21,266],
        [0,4 ,21,6438]], dtype=int64).
```

CHAPTER 7
CONCLUSION

7.1 CONCLUSION

In conclusion, while more data is required to balance out the lower rated products in order to consider their significance, we were able to successfully associate with positive, neutral, and negative sentiments for every product in Catalog.

7.2 Future Work

The future work for a project on analyzing Amazon reviews using sentiment analysis would depend on the specific goals and objectives of the project, but here are some potential ideas:

- Improve the accuracy of the sentiment analysis: Sentiment analysis is not perfect and can sometimes misclassify sentiment. Therefore, improving the accuracy of the analysis would be a key area of focus for future work.
- Expand the analysis to other marketplaces: Amazon is just one marketplace for consumer goods. Expanding the analysis to other marketplaces like Walmart, Best Buy, and Target would provide a more comprehensive view of consumer sentiment.
- Incorporate user demographics: Understanding the demographics of the users leaving reviews could provide valuable insights into how different groups of people perceive products. Incorporating demographic data could also help identify trends and patterns that may be missed with sentiment analysis alone.
- Compare sentiment analysis to sales data: Sentiment analysis can provide insights into how people feel about a product, but it doesn't necessarily correlate with sales. Comparing sentiment analysis to sales data could provide a more complete picture of a product's success.
- Use sentiment analysis for product development: Sentiment analysis could be used to inform product development by identifying the features and attributes that customers like or dislike. This could help companies make data-driven decisions about product design and development.
- Overall, there are many different directions that a project on Amazon reviews using sentiment analysis could take in the future, depending on the specific goals of the project and the needs of the business or organization conducting

the analysis.

7.3 Research issues

There are several research issues that could be explored in a project involving Amazon reviews and sentiment analysis. Here are a few potential topics to consider:

- **Accuracy of sentiment analysis:** One issue with sentiment analysis is its accuracy. Researchers could investigate the accuracy of sentiment analysis tools when applied to Amazon reviews, particularly given the complexity of natural language and the potential for sarcasm, irony, and other forms of language that may not be easily classified as positive or negative.
- **Effectiveness of review analysis:** Another issue to consider is whether sentiment analysis of Amazon reviews is an effective way to gauge customer sentiment about a product. Researchers could explore whether other factors such as the number of reviews or the timing of reviews may also impact the overall perception of a product.
- **Generalizability of findings:** It's important to consider whether the findings of any sentiment analysis of Amazon reviews are generalizable to other products or industries. For example, do the sentiments expressed in Amazon reviews of electronics products hold true for other consumer goods or services?
- **Impact of reviews on customer behavior:** Researchers could investigate the impact of Amazon reviews on customer behavior, such as whether positive reviews lead to more purchases, or whether negative reviews lead to fewer purchases. This could provide insight into the importance of reviews as a marketing tool.
- **Ethical considerations:** There are also ethical considerations to be explored when it comes to analyzing Amazon reviews. For example, is it ethical to analyze reviews without the consent of the reviewers? What about the privacy implications of collecting and analyzing large amounts of customer data? Researchers could explore these issues and others related to data privacy and ethics in their analysis.

7.4 Implementation issues

There are several implementation issues to consider when working on a project involving Amazon reviews and sentiment analysis. Here are a few potential issues to keep in mind:

- **Data collection:** One key issue is collecting the data needed for sentiment analysis. This may involve scraping Amazon reviews, which can be technically challenging and may violate Amazon's terms of service. Alternatively, researchers could work with a third-party data provider that specializes in collecting and providing Amazon review data.
- **Data preprocessing:** After collecting the data, it will need to be preprocessed to prepare it for sentiment analysis. This may involve cleaning the data, removing duplicate reviews, and identifying the most relevant reviews for analysis.
- **Choice of sentiment analysis tool:** There are a variety of sentiment analysis tools available, each with its own strengths and weaknesses. Researchers will need to choose a tool that is appropriate for their specific research question and dataset.
- **Handling ambiguity:** Sentiment analysis is not always straightforward, as natural language can be ambiguous and difficult to classify as positive or negative. Researchers will need to develop strategies for handling this ambiguity and ensuring that their results are as accurate as possible.
- **Integration with other data sources:** Researchers may want to integrate Amazon review data with other data sources, such as sales data or demographic data, to gain a more complete picture of customer sentiment. This will require technical expertise and an understanding of how to effectively integrate different data sources.
- **Scalability:** Depending on the size of the dataset, sentiment analysis can be computationally expensive and time-consuming. Researchers will need to ensure that their analysis is scalable and can be applied to larger datasets if needed.
- **Interpretation of results:** Finally, it's important to consider how the results of the sentiment analysis will be interpreted and presented. Researchers will need to develop clear and meaningful visualizations and explanations of their findings in order to effectively communicate their results to stakeholders.

REFERENCES:

[1] Dataset:https://www.kaggle.com/datafiniti/consumer-reviews-of-amazon-products

[2] Anh, K.Q., Nagai, Y. and Nguyen, L.M., 2019. Extracting customer reviews from online shopping and its perspective on product design. Vietnam Journal of Computer Science, 6(01), pp.43-56.

[3] Chong, A.Y.L., Li, B., Ngai, E.W.,Chang, E. and Lee, F., 2016. Predicting online product sales via online reviews,sentiments, and promotion strategies: A big data architecture and neural network approach. International Journal of Operations & Production Management.

[4] Chauhan, U.A., Afzal, M.T., Shahid, A., Abdar, M., Basiri, M.E., and Zhou, X., 2020.

[5] R. Collobert, J. Weston, L. Bottou, M. Karlen,K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. Journal of Machine Learning Research, 12(Aug):2493–2537, 2011.

[6] K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In Proceedings of the 12th international conference on World Wide Web, pages 519–528. ACM, 2003.

[7] M. S. Elli and Y.-F. Wang. Amazon reviews, business analytics with sentiment analysis.

[8] S. Hota and S. Pathak. Knn classifier based approach for

multi-class sentiment analysis of twitter data. In International Journal of Engineering Technology, pages 1372–1375. SPC, 2018.

[9] B. Liu and L. Zhang. A Survey of Opinion Mining andSentiment Analysis, pages 415–463. Springer US, Boston, MA, 2012.

[10] C. Rain. Sentiment analysis in amazon reviews using probabilistic machine learning. Swarthmore College, 2013.

[11] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning,A. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedingsof the 2013 conference on empirical methods in natural language processing, pages 1631–1642, 2013.

[12] Y. Xu, X. Wu, and Q. Wang. Sentiment analysis of yelps ratings based on text reviews, 2015.

[13] Du, J, Rong, J, Michalska, S, Wang, H & Zhang, Y. 2019. Feature selection for helpfulness prediction of online product reviews: An empirical study', PloS one, 14(12), p. e0226902. Dey, S., Wasif, S., Tonmoy, D.S., Sultana, S., Sarkar, J., and Dey, M., 2020.

[14] Blitzer, J., M. Dredze, and F. Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In Proceed-ings of Annual Meeting of the Association forComputational Linguistics (ACL-2007), 2007.

[15] Bollegala, D., D. Weir, and J. Carroll. Using multiple sources to construct a sentiment sensitive thesaurus for cross-domain sentiment classification. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL-2011), 2011.

[16] Abdul-Mageed, M., M.T. Diab, and M. Korayem. Subjectivity and sentiment analysis of modern standard Arabic. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:shortpapers, 2011.

[17] Andreevskaia, A. and S. Bergler. Mining WordNet for fuzzy sentiment: Sentiment tag extraction from WordNet glosses. In Proceedings of Conference of the European Chapter of the Association for Computational Linguistics (EACL-06), 2006.

APPENDIX

A.SOURCE CODE

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
import math
import warnings
warnings.filterwarnings('ignore') # Hides warning
warnings.filterwarnings("ignore", category=DeprecationWarning)
warnings.filterwarnings("ignore",category=UserWarning)
sns.set_style("whitegrid") # Plotting style
%matplotlib inline # Plots show up in notebook
np.random.seed(7) # seeding random number generator
csv = "1429_1.csv"
df = pd.read_csv(csv)
df.head(2)
data = df.copy()
data.describe()
data.info()
data["asins"].unique()
asins_unique = len(data["asins"].unique())
print("Number of Unique ASINs: " + str(asins_unique))
data.hist(bins=50, figsize=(20,15)) # builds histogram and set the number of bins and fig size (width, height)
plt.show()
from sklearn.model_selection import StratifiedShuffleSplit
print("Before {}".format(len(data)))
dataAfter = data.dropna(subset=["reviews.rating"]) # removes all NAN in reviews.rating
print("After {}".format(len(dataAfter)))
```

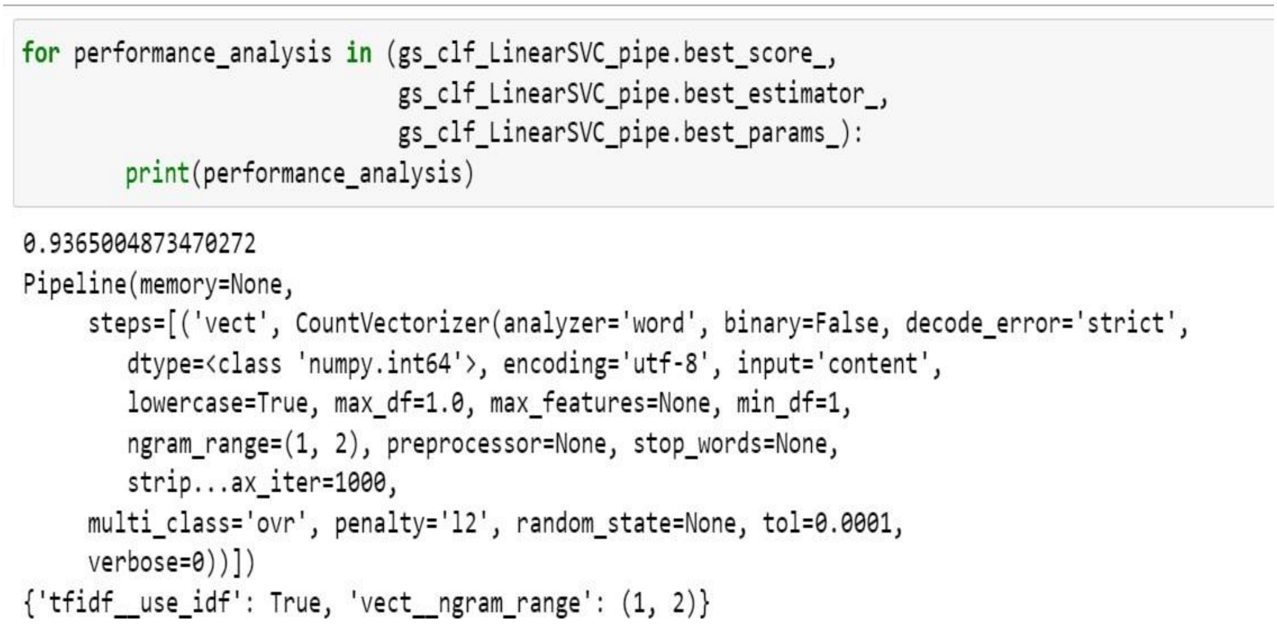
```
dataAfter["reviews.rating"] = dataAfter["reviews.rating"].astype(int)
split = StratifiedShuffleSplit(n_splits=5, test_size=0.2)
fortrain_index,test_indexinsplit.split(dataAfter,dataAfter["reviews.rating":
strat_train = dataAfter.reindex(train_index)
strat_test = dataAfter.reindex(test_index)
len(strat_train)
strat_train["reviews.rating"].value_counts()/len(strat_train) # value_count() counts all the values based on column
len(strat_test)
strat_test["reviews.rating"].value_counts()/len(strat_test)
reviews = strat_train.copy()
reviews.head(2)
len(reviews["name"].unique()), len(reviews["asins"].unique())
reviews.info()
reviews.groupby("asins")["name"].unique()
# Lets see all the different names for this product that have 2 ASINs
different_names = reviews[reviews["asins"] == "B00L9EPT8O,B01E6AO69U"]["name"].unique()
for name in different_names:
   print(name)
reviews[reviews["asins"]=="B00L9EPT8O,B01E6AO69U"]["name"].value_counts()
fig = plt.figure(figsize=(16,10))
ax1 = plt.subplot(211)
ax2 = plt.subplot(212, sharex = ax1)
reviews["asins"].value_counts().plot(kind="bar", ax=ax1, title="ASIN Frequency")
np.log10(reviews["asins"].value_counts()).plot(kind="bar", ax=ax2, title="ASIN Frequency (Log10 Adjusted)")
plt.show()
# Entire training dataset average rating
reviews["reviews.rating"].mean()
asins_count_ix = reviews["asins"].value_counts().index
```

```
plt.subplots(2,1,figsize=(16,12))
plt.subplot(2,1,1)
reviews["asins"].value_counts().plot(kind="bar", title="ASIN Frequency")
plt.subplot(2,1,2)
sns.pointplot(x="asins", y="reviews.rating", order=asins_count_ix, data=reviews)
plt.xticks(rotation=90)
plt.show()
plt.subplots (2,1,figsize=(16,12))
plt.subplot(2,1,1)
reviews["asins"].value_counts().plot(kind="bar", title="ASIN Frequency")
plt.subplot(2,1,2)
sns.pointplot(x="asins", y="reviews.doRecommend", order=asins_count_ix,
data=reviews)
plt.xticks(rotation=90)
plt.show()
corr_matrix = reviews.corr()
corr_matrix
# Here we can analyze reviews.ratings with asins
reviews.info()
counts = reviews["asins"].value_counts().to_frame()
counts.head()
avg_rating = reviews.groupby("asins")["reviews.rating"].mean().to_frame()
avg_rating.head()
table = counts.join(avg_rating)
table.head(30)
plt.scatter("asins", "reviews.rating", data=table)
table.corr()
def sentiments(rating):
    if (rating == 5) or (rating == 4):
        return "Positive"
    elif rating == 3:
        return "Neutral"
    elif (rating == 2) or (rating == 1):
        return "Negative"
# Add sentiments to the data
strat_train["Sentiment"] = strat_train["reviews.rating"].apply(sentiments)
strat_test["Sentiment"] = strat_test["reviews.rating"].apply(sentiments)
strat_train["Sentiment"][:20]
# Prepare data
X_train = strat_train["reviews.text"]
X_train_targetSentiment = strat_train["Sentiment"]
X_test = strat_test["reviews.text"]
X_test_targetSentiment = strat_test["Sentiment"]
print(len(X_train), len(X_test))
# Replace "nan" with space
X_train = X_train.fillna(' ')
X_test = X_test.fillna(' ')
X_train_targetSentiment = X_train_targetSentiment.fillna(' ')
X_test_targetSentiment = X_test_targetSentiment.fillna(' ')
# Text preprocessing and occurance counting
from sklearn.feature_extraction.text import CountVectorizer
count_vect = CountVectorizer()
X_train_counts = count_vect.fit_transform(X_train)
X_train_counts.shape
from sklearn.feature_extraction.text import TfidfTransformer
tfidf_transformer = TfidfTransformer(use_idf=False)
X_train_tfidf = tfidf_transformer.fit_transform(X_train_counts)
X_train_tfidf.shape
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import Pipeline
clf_multiNB_pipe = Pipeline([("vect", CountVectorizer()), ("tfidf", TfidfTransformer()),
("clf_nominalNB", MultinomialNB())])
```

```
clf_multiNB_pipe.fit(X_train, X_train_targetSentiment)
import numpy as np
predictedMultiNB = clf_multiNB_pipe.predict(X_test)
np.mean(predictedMultiNB == X_test_targetSentiment)
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import Pipeline
clf_logReg_pipe = Pipeline([("vect", CountVectorizer()), ("tfidf", TfidfTransformer()),
("clf_logReg", LogisticRegression())])
clf_logReg_pipe.fit(X_train, X_train_targetSentiment)
import numpy as np
predictedLogReg = clf_logReg_pipe.predict(X_test)
np.mean(predictedLogReg == X_test_targetSentiment)
from sklearn.svm import LinearSVC
clf_linearSVC_pipe = Pipeline([("vect", CountVectorizer()), ("tfidf",
TfidfTransformer()), ("clf_linearSVC", LinearSVC())])
clf_linearSVC_pipe.fit(X_train, X_train_targetSentiment)
predictedLinearSVC = clf_linearSVC_pipe.predict(X_test)
np.mean(predictedLinearSVC == X_test_targetSentiment)
from sklearn.tree import DecisionTreeClassifier
clf_decisionTree_pipe = Pipeline([("vect", CountVectorizer()), ("tfidf",
TfidfTransformer()),
("clf_decisionTree", DecisionTreeClassifier())])
clf_decisionTree_pipe.fit(X_train, X_train_targetSentiment)
predictedDecisionTree = clf_decisionTree_pipe.predict(X_test)
np.mean(predictedDecisionTree == X_test_targetSentiment)
from sklearn.ensemble import RandomForestClassifier
clf_randomForest_pipe = Pipeline([("vect", CountVectorizer()),
("tfidf",TfidfTransformer()),("clf_randomForest",RandomForestClassifier())])
clf_randomForest_pipe.fit(X_train, X_train_targetSentiment)
predictedRandomForest = clf_randomForest_pipe.predict(X_test)
np.mean(predictedRandomForest == X_test_targetSentiment)
```

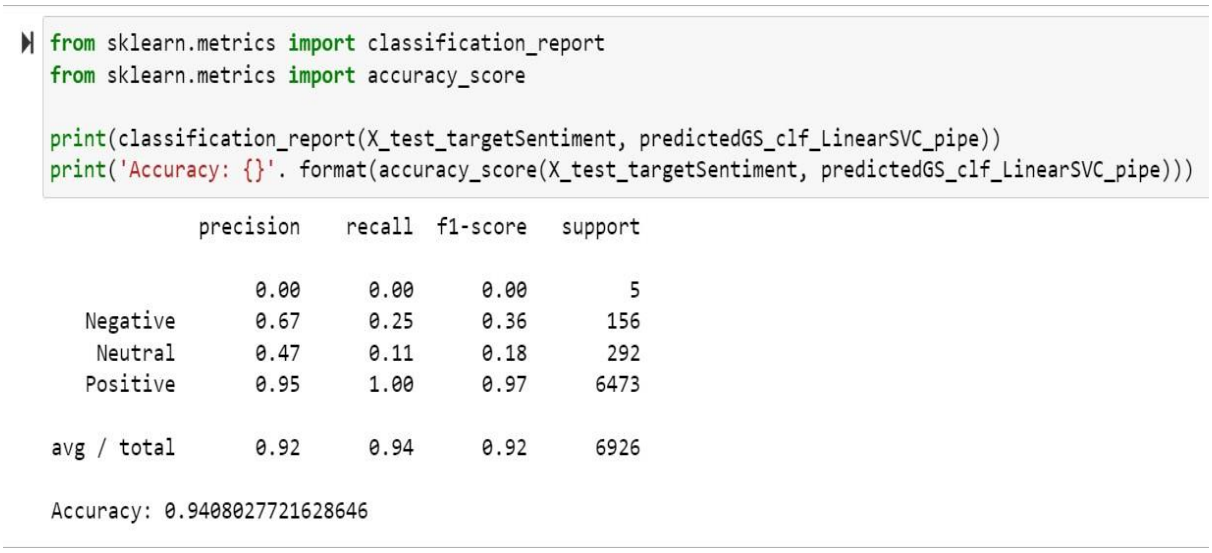
```
predictedGS_clf_LinearSVC_pipe=gs_clf_LinearSVC_pipe.predict(X_test)
np.mean(predictedGS_clf_LinearSVC_pipe==X_test_targetSentiment)
for performance_analysis in (gs_clf_LinearSVC_pipe.best_score_,
                             gs_clf_LinearSVC_pipe.best_estimator_,
                             gs_clf_LinearSVC_pipe.best_params_):
        print(performance_analysis)
from sklearn.metrics import classification_report
from sklearn.metrics import accuracy_score

print(classification_report(X_test_targetSentiment,
predictedGS_clf_LinearSVC_pipe))
print('Accuracy: {}'.format(accuracy_score(X_test_targetSentiment,
predictedGS_clf_LinearSVC_pipe)))
from sklearn import metrics
metrics.confusion_matrix(X_test_targetSentiment,
predictedGS_clf_LinearSVC_pipe)
```

B.SCREENSHOTS



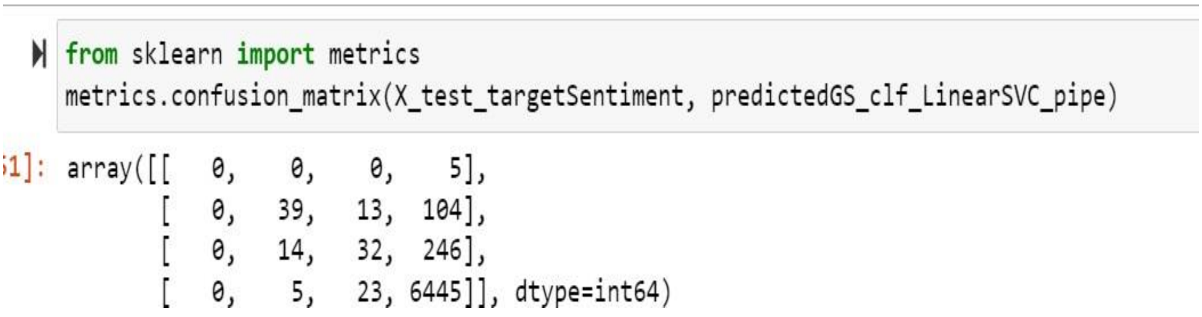
```
for performance_analysis in (gs_clf_LinearSVC_pipe.best_score_,
                             gs_clf_LinearSVC_pipe.best_estimator_,
                             gs_clf_LinearSVC_pipe.best_params_):
        print(performance_analysis)
```

```
0.9365004873470272
Pipeline(memory=None,
     steps=[('vect', CountVectorizer(analyzer='word', binary=False, decode_error='strict',
        dtype=<class 'numpy.int64'>, encoding='utf-8', input='content',
        lowercase=True, max_df=1.0, max_features=None, min_df=1,
        ngram_range=(1, 2), preprocessor=None, stop_words=None,
        strip...ax_iter=1000,
     multi_class='ovr', penalty='l2', random_state=None, tol=0.0001,
     verbose=0))])
{'tfidf__use_idf': True, 'vect__ngram_range': (1, 2)}
```



```
from sklearn.metrics import classification_report
from sklearn.metrics import accuracy_score

print(classification_report(X_test_targetSentiment, predictedGS_clf_LinearSVC_pipe))
print('Accuracy: {}'. format(accuracy_score(X_test_targetSentiment, predictedGS_clf_LinearSVC_pipe)))
```

```
             precision    recall  f1-score   support

                  0.00      0.00      0.00         5
   Negative       0.67      0.25      0.36       156
    Neutral       0.47      0.11      0.18       292
   Positive       0.95      1.00      0.97      6473

avg / total       0.92      0.94      0.92      6926

Accuracy: 0.9408027721628646
```



```
from sklearn import metrics
metrics.confusion_matrix(X_test_targetSentiment, predictedGS_clf_LinearSVC_pipe)
```

```
i1]: array([[   0,    0,    0,    5],
            [   0,   39,   13,  104],
            [   0,   14,   32,  246],
            [   0,    5,   23, 6445]], dtype=int64)
```

Research paper

Review using Sentimental Analysis

Sharvirala Kethan
Sathyabama Instuite of science and technology
E-mail:kethankumar637@gmail.com

A.Mary Posonia
Sathyabama Instuite of science and technology
E-mail: maryposonia.cse@sathyabama.ac.in

Abstract— **In day to day life the Reviews has became like part of cooperative industry. Because by using the reviews only the cooperative company come to now how there products are reaching to the customer and their reactions to the products. Using these reviews only they come to now how their feedback is given to the products so using the reviews only they can decide to change the products according to their customers.so in this project the user need to know that reviews of products form customer to do changes or to remove from the Product roaster so for that we are going to take the dataset where it has the all the products with their unique id and process start with the data exploration and then we go for the correlation and for the nature of reviews we use sentimental analysis and the find the which algorithm is best for the process using the accuracy and then we classify the algorithm and we get the classification report for the model and then we get to now the reviews of customers. Keywords— Machine learning, Python, Dataset, Data Exploration, Correlations, Sentimental Analysis, Naive Bayes, Logistic Regression Classifier, Support Vector Machine Classifier, Decision Tree Classifier, Random Forest Classifier.**

I. INTRODUCTION

People buy goods from multiple e-commerce websites when the entire world's commercial sites are virtually on the internet. Reviews of products before purchases are frequently a matter of prerogative. Consumers are more likely to purchase a good after reading reviews. Online retailers and vendors solicit feedback from their customers on their goods. Every day, millions of reviews of products, facilities, and locations are posted online. People buy goods from multiple e-commerce websites when the entire world's commercial sites are virtually on the internet. Reviews of products before purchases are frequently a matter of prerogative. Consumers are more likely to purchase a good after reading reviews. Online retailers and vendors solicit feedback from their customers on their goods. Every day, millions of reviews of products, facilities, and locations are posted online. In the age of artificial intelligence, it takes time to polarize a sample of reviews into particular categories in order to assess a company's appeal to consumers around the world. Analyzing data from specific consumer comments is an important area today.

Customers can write product or service reviews in areas of e-commerce websites to provide comments, suggestions, and thoughts on products. Therefore, e-commerce enterprises absolutely require the study of this review. The number of online reviews is enormous today, and the number of forums is outpacing growth. Several web formats for consumer analysis are available, including products of a particular type (like video cameras), articles from publications like Rolling Stone and customer reports, industry articles on businesses like Amazon, and technical and pages for user analysis in a variety of domains. The websites and forums of websites like, and onfocus.com also feature customer reviews of products. pioneers that consumers use daily for online shopping and receives hundreds of reviews from customers about their favorite products. Consumers rate products and services. As a result of individual ratings determining exact values during the analysis, reviews are grouped with incompatible ratings.

II. LITERATURE SURVEY

Sentiment analysis used recently gained a lot of popularity in the fields of text mining and computational linguistics. Classifying the data is a fundamental effort in sentiment analysis. the document, phrase, or feature/aspect level polarity of a given text. Five key steps make up the sentiment analysis method. They are data gathering, text preparation, sentiment categorization, sentiment detection, and output display.

Fang and Zhan (2015)[1] has conducted sentimental analysis on the data from product reviews, using a Naive Bayes classifier to extract subjective content and address the problem of polarity categorisation. Amazon's online product reviews dataset has been used in this study's research.

The disadvantage of this method is a Naive Bayes does not produce satisfying accuracy.

Goyal and Parulaker (2015)[2] analysed text-based movie reviews by counting the occurrences of each term using a random forest classifier. Sentiment analysis can be used to determine the reviewer's emotional state, such as whether they were "happy," "sad," "angry," or soon. Examine the tone of a few critics' evaluations of different movies to ascertain if they believed the movie was good or awful. the relationships among the words in the review to predict its overall polarity."

Rahul Wadbude [3]and his team (Wadbude et al., 2016) used The field of natural language processing has recently paid a lot of attention. The majority of the early work was concentrated on developing effective feature representations of the classification-related text reviews. During fine-grained sentiment categorization, the algorithms typically disregard other common criteria like user identity, product identity, and help fullness rating.

According to Sharma, Chakraborti,[4] and Jha (2019), Over the past ten years, online shopping has grown in popularity across the globe.

Existing work of the project is that the creator of the Model with all the process like he used Data Exploration, Correlation, Sentimental Analysis and for the implements the machine learning models he used only the one Algorithm model i.e. Multinominal Naive Bayes.

III.PROPOSED METHODOLOGY

The Proposed methodology for the model is Know the reviews of Product using the comments they gave. The purpose of this model to Analysis the bulk of comments and categorize the products according to their unique id and also we Data Exploration for the model in that we did the process for the Training dataset.

And also Explore the column of dataset. Here we explore all the columns details to check with unique values. Further we will go for Correlation technique to Analysis the columns and we are Check sentimental analysis to know the nature of the comments like positive or negative. To deal the model for classification we use many Support vector machine, Machine learning models like naïve bayes, Random forest, Logistic regression.

Work-Flow Diagram

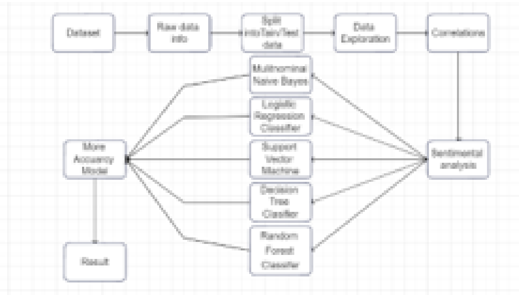
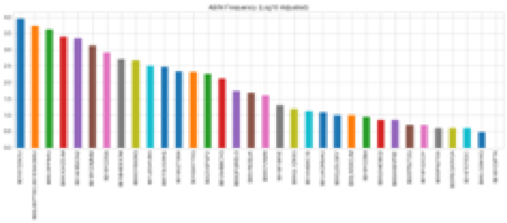


Figure No-1

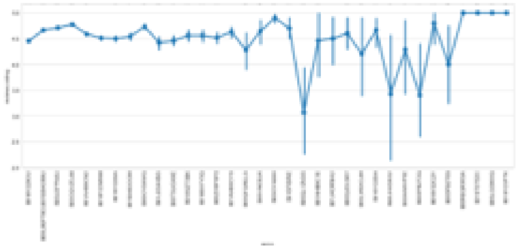
Implementation:

Take the Kaggle dataset and analyse it. Add all required library functions to it.Open the Notebook and read the csv dataset. Using commands, obtain full dataset information. Brief the dataset after visualising it.Divide the dataset into test and training data.Start Examining the training data and the columns,Following exploration, compare every column to look for any duplicate names. Consider creating graphs for the dataset's columns with names, ASINs, reviews.rating,

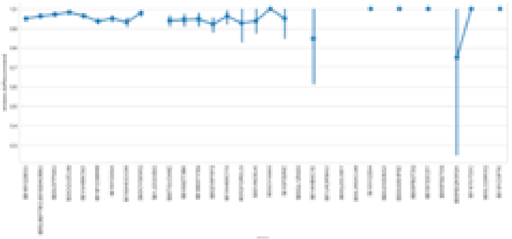
ASINS FREQUENCY



POINT-PLOT OF REVIEWS.RATING



POINT PLOT OF REVIEWS.DORECOMMEND



and ASINs for reviews that do recommend. Continue with graphs after the representation of the graph. As a next step, we will analyse the review using the correlation technique. Using the asins column to rate. Look at review.info and do a head count. Find the mean of the column's ratings. Plot the correlation Scatter plot graph. Start your sentimental analysis now. First Set the target variable for sentiments. Check the Nature of the data and preapre data for a sentimental analysis.

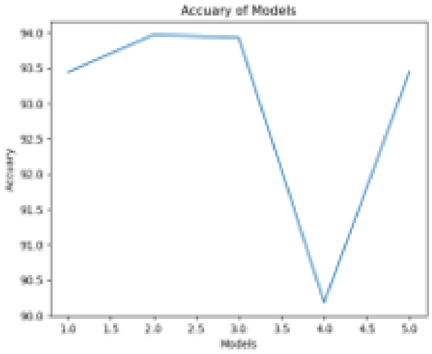
Asins	Review.rating
B005PB2T0S	4.000000
B005PB2T2Q	3.400000
B00IOY8XWQ	4.731183
B00IOYAM4I	4.900000
B00L9EPT8O,B01E6AO69U	4.663462

The data extraction feature is now ready. We use Count Vectorizer from SciKit-Learn for the Extract feature procedure. We process the Text Preprocessing, occurrence counting, Feature vector in scikit-Count vectorizer Then Pipeline is develop for extracted Features.We use Multinomial Naive Bayes for pipeline.

After building the pipeline, now test the model with Multinomial Naïve Bayes algorithm and Check the model's accuracy. The remaining algorithms, such as the Logistic Regression Classifier, Support Vector Machine Classifier, Decision Tree Classifier, and Random Forest Classifier, are also checked for greater accuracy.

Accuracy of Algorithm:

S.NO	ALGORTHIM NAME	ACCUARCY
1.	Multinomial Naïve Bayes	0.93444
2.	Logistic Regression Classifier	0.93704
3.	Support vector Machine Classifier	0.93935
4.	Decision Tree Classifier	0.90181
5.	Random Forest Classifier	0.93459



The accuracy of each method must finally be compared, and the algorithm with the highest accuracy must be used for the remaining steps.

We will use the train dataset for each algorithm model and fit the data to that model. After identifying the algorithm with the highest accuracy relative to the others, a detailed analysis of that method must be done. Find the algorithm's confusion matrix. Observe the outcome.

IV. Result and Discussions

We will ignore the first row and column because we already replaced all NAN with " ". This is the identical circumstance as was described in the classification report above.

By looking at row 2-4 and column 2-4, which are labelled as negative, neutral, as well as positive, with scores of 246 and 104, respectively, we see that positive sentiment can occasionally be confused with neutral and negative sentiment. Even so, confusion scores of 246 and 104 for neutral as well as negative ratings, respectively, are considered negligible when compared to the total amount of significant positive sentiment at 6445.

Here we get to know that this a positive skewed dataset, where it is consistent with the sentimental analysis. As a result we will conclude that dataset is well liked and we should remain the items in roster and we find the how many items were selected.

	Precision	recall	f1-score	support
	0.00	0.00	0.00	5
Negative	0.66	0.24	0.34	155
Neutral	0.46	0.10	0.16	293
Positive	0.94	1.01	0.95	6476
AVG/TOTAL	0.91	0.93	0.91	6925

Confusion Matrix:

```
array([[0,0 ,0 ,4],
       [0,32,7 ,119],
       [0,14,21,266],
       [0,4 ,21,6438]], dtype=int64)
```

V. Conclusion

In conclusion, we get to know that more data is needed to get balance of lower rated items to be significance. Here where successfully finding the data nature with sentimental like positive and negative for every catalog.

VII. References

[1] Dataset:https://www.kaggle.com/datafiniti/consumer-reviews-of-amazon-products

[2] Anh, K.Q., Nagai, Y. and Nguyen, L.M., 2019. Extracting customer reviews from online shopping and its perspective on product design. Vietnam Journal of Computer Science, 6(01), pp.43-56.

[3] Chong, A.Y.L., Li, B., Ngai, E.W.,Chang, E. and Lee, F., 2016. Predicting online product sales via online reviews,sentiments, and promotion strategies: A big data architecture and neural network approach. International Journal of Operations & Production Management.

[4] Chauhan, U.A., Afzal, M.T., Shahid, A., Abdar, M., Basiri, M.E., and Zhou, X., 2020.

[5] R. Collobert, J. Weston, L. Bottou, M. Karlen,K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. Journal of Machine Learning Research, 12(Aug):2493–2537, 2011.

[6] K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In Proceedings of the 12th international conference on World Wide Web, pages 519–528. ACM, 2003.

[7] M. S. Elli and Y.-F. Wang. Amazon reviews, business analytics with sentiment analysis.

[8] S. Hota and S. Pathak. Knn classifier based approach for multi-class sentiment analysis of twitter data. In International Journal of Engineering Technology, pages 1372–1375. SPC, 2018.

[9] B. Liu and L. Zhang. A Survey of Opinion Mining andSentiment Analysis, pages 415–463. Springer US, Boston, MA, 2012.

[10] C. Rain. Sentiment analysis in amazon reviews using probabilistic machine learning. Swarthmore College, 2013.

[11] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning,A. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedingsof the 2013 conference on empirical methods in natural language processing, pages 1631–1642, 2013.

[12] Y. Xu, X. Wu, and Q. Wang. Sentiment analysis of yelps

ratings based on text reviews, 2015.

[13] Du, J, Rong, J, Michalska, S, Wang, H & Zhang, Y. 2019. Feature selection for helpfulness prediction of online product reviews: An empirical study', PloS one, 14(12), p. e0226902. Dey, S., Wasif, S., Tonmoy, D.S., Sultana, S., Sarkar, J., and Dey, M., 2020.