

GROUP ID: GD 8

A PROJECT REPORT ON

**HYBRID MACHINE LEARNING APPROACH FOR SENTIMENT
ANALYSIS OF AMAZON PRODUCTS**

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN
THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE OF

**BACHELOR OF ENGINEERING
(COMPUTER ENGINEERING)**

SUBMITTED BY

**OM SARULKAR
SHIVAM TIKHE
SUMIT GIRI
ROHAN MORE**

**Exam No: B190334455
Exam No: B190334501
Exam No: B190334297
Exam No: B190334376**



**DEPARTMENT OF COMPUTER ENGINEERING
PCET's PIMPRI CHINCHWAD COLLEGE OF ENGINEERING**

SECTOR NO. 26, PRADHIKARAN, NIGDI, PUNE 411044

**SAVITRIBAI PHULE PUNE UNIVERSITY
2022 -2023**



CERTIFICATE

This is to certify that the project report entitled

“HYBRID MACHINE LEARNING APPROACH FOR SENTIMENT ANALYSIS OF AMAZON PRODUCTS: A SURVEY”

Submitted by

OM SARULKAR

Roll No: B190334455

SHIVAM KISHOR TIKHE

Roll No: B190334501

SUMIT GIRI

Roll No: B190334297

ROHAN MORE

Roll No: B190334376

is a bonafied student of this institute and the work has been carried out by him/her under the supervision of **Prof. Rahul Pitale** and it is approved for the partial fulfilment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering** (Computer Engineering).

(Prof. Rahul Pitale)

(Dr. K. Rajeswari)

Guide

Head,

Department of Computer Engineering

Department of Computer Engineering

(Prof. Dr. G.N. Kulkarni)

Director,

Pimpri Chinchwad College of Engineering Pune – 44

Place: Pune

ACKNOWLEDGEMENT

We express our sincere thanks to our **Guide Prof. Rahul Pitale** for his/her constant encouragement and support throughout our project, especially for the useful suggestions given during the course of project and having laid down the foundation for the success of this work.

We would also like to thank our **Project Coordinator, Prof. Sushma R. Vispute** for her assistance, genuine support and guidance from early stages of the project. We would like to thank **Prof. Dr. K. Rajeswari, Head of Computer Department** for her unwavering support during the entire course of this project work. We are very grateful to our **Director, Prof. Dr. G.N. Kulkarni** for providing us with an environment to complete our project successfully. We also thank all the staff members of our college and technicians for their help in making this project a success.

We also thank all the web committees for enriching us with their immense knowledge. Finally, we take this opportunity to extend our deep appreciation to our family and friends, for all that they meant to us during the crucial times of the completion of our project.

NAME OF THE STUDENTS

Om Sarulkar	B190334455
Shivam Tikhe	B190334501
Sumit Giri	B190334297
Rohan More	B190334376

ABSTRACT

This project focuses on developing a machine learning and natural language processing-based solution for sentiment analysis of Amazon product reviews. The project encompasses various modules, including data collection, pre-processing, model training, and the creation of a local web application for review classification.

Initially, a dataset of labelled positive and negative Amazon product reviews is downloaded from the Amazon website. The collected data is then pre-processed using libraries such as Pandas and NumPy to clean the text data, handle missing values, and perform feature engineering. Several machine learning algorithms, including Support Vector Machine (SVM), Random Forests, K Nearest Neighbours, and Decision Tree Classifiers, are trained using Scikit-learn. The trained models are evaluated, and the SVM model, with the highest accuracy, is selected for sentiment classification. This model is serialized using the Pickle library for future use. Subsequently, a local web application is developed using Flask, HTML, and CSS. The web application accepts an Amazon product URL as input, scrapes online reviews using BeautifulSoup, and utilizes the SVM model to classify the reviews as positive or negative. Through this project, users can conveniently analyse the sentiment of Amazon product reviews by simply inputting the product URL into the web application. The application provides accurate classification results, enhancing user decision-making when it comes to purchasing products based on customer feedback. Overall, this project combines machine learning, natural language processing, and web development to create an effective solution for sentiment analysis of online reviews.

Keywords – Sentiment analysis, Sentiment classification, opinion mining, machine learning, polarity classification, supervised algorithms, Amazon classification, Ensemble Learning, Hybrid Machine learning

TABLE OF CONTENTS

Sr. No.	Title of Chapter	Page No.
01	Introduction	1
1.1	Overview	1
1.2	Motivation	2
1.3	Problem statement and Objectives	3
1.4	Project Scope & Limitations	3
02	Literature Survey	4
03	Software Requirements Specification	11
3.1	Assumptions and Dependencies	11
3.2	Functional Requirements	12
3.3	External Interface Requirements (If Any)	14
3.3.1	User Interfaces	14
3.3.2	Hardware Interfaces	15
3.3.3	Communication Interfaces	16
3.4	Nonfunctional Requirements	17
3.4.1	Performance Requirements	17
3.4.2	Safety Requirements	17
3.4.3	Software Quality Attributes	18
3.5	System Requirements	19
3.5.1	Software Requirements (Platform Choice)	19
3.5.2	Hardware Requirements	20
3.6	Analysis Models: SDLC Model to be applied	22
04	System Design	24
4.1	System Architecture	24
4.2	Mathematical Model	25
4.5	UML Diagrams	26
05	Project Plan	29
5.1	Project Estimate	29
5.1.1	Reconciled Estimates	29
5.1.2	Project Resources	29
5.2	Risk Management	30
5.2.1	Risk Identification	30
5.2.2	Risk Analysis	30
5.2.3	Overview of Risk Mitigation, Monitoring, Management	30
5.3	Project Schedule	31
5.3.1	Project Task Set	31
5.3.2	Timeline Chart	31
5.4	Team Organization	33
5.4.1	Team structure	33
5.4.2	Management reporting and communication	33
06	Project Implementation	35
6.1	Overview of Project Modules	35

	6.2	Tools and Technologies Used	36
	6.3	Algorithm Details	38
	6.3.1	Support Vector Machine	38
	6.3.2	Random Forest	38
	6.3.3	Logistic Regression	39
	6.3.4	Decision Tree Classifier	40
07		Software Testing	41
	7.1	Type of Testing	41
	7.2	Test cases & Test Results	42
08		Results	44
	8.1	Outcomes	44
	8.2	Screen Shots	45
09		Conclusions	48
	9.1	Conclusions	48
	9.2	Future Work	48
	9.3	Applications	49
		Appendix A: Problem statement feasibility assessment using, satisfiability analysis and NP Hard, NP-Complete or P type using modern algebra and relevant mathematical models.	51
		Appendix B: Details of paper publication: name of the conference/journal, comments of reviewers, certificate, paper. (Atleast . 2 papers)	52
		Appendix C: Plagiarism Report of project report.	53

LIST OF ABBREVIATIONS

Abbreviation	Illustration
SNA	Social Network Analysis
ML	Machine Learning
SVM	Support Vector Machine
RF	Random Forests
NLP	Natural Language Processing
HTML	Hyper Text Transfer Protocol
CSS	Cascading Style Sheets

LIST OF FIGURES

Figure No.	Illustration	Page No.
1.1	Example of reviews of amazon product	19
4.2.1	Activity Diagram	26
4.2.2	Model diagram	27
4.2.3	Class Diagram	27
4.2.4	State Chart Diagram	28
4.1	System Architecture	24
6.1	SVM Architecture	38
6.2	Random Forest architecture	39
6.3	Logistics Regression Architecture	39
6.4	Decision Tree Architecture	40
8.1	Dataset For Training	45
8.2	Logistic Regression Evaluation Metrics	45
8.3	SVM Evaluation Metrics	45
8.4	KNN Evaluation Metrics	46
8.5	Decision Tree Evaluation Metrics	46
8.6	User Interface before pasting link	47
8.7	User Interface after pasting link	47

LIST OF TABLES

TABLE	ILLUSTRATION	PAGE NO.
5.1	Timeline Chart	31

CHAPTER 1

INTRODUCTION

In the modern world, media platforms, online retail and ecommerce play a significant part in forming an online community and allowing them to voice their views and ideas on any topic. For instance, amazon inc. subsidiary, amazon retail is a well-known online store these days. It has an option given to users to post and converse about their opinions about any item available on the platform, due to which a huge amount of data is generated which is classified as semi-structured data.

In order to uncover crucial information about the items that have reviews posted about them, understand people's sentiment, sentiment analysis is utilised to explore and assess this data. Sentiment analysis (SA), often known as text classification or sentiment analysis, is an integral branch in natural language processing (NLP). The branch of machine learning to understand human language is called natural language processing.

This study evaluates recent supervised classification algorithms and their combination that have been used to identify sentiment analysis in Amazon product evaluations in order to locate the best one that can deliver trustworthy and accurate findings. This method may then be used as a starting point for Amazon reviews, categorization jobs, recommendation systems, and so on

1.1 OVERVIEW

Amazon is one of the biggest internet merchants in the world. It had expanded since its inception as an online platform in 1994. It now offers over 12 million goods and has 200 million active users accessing the store from their PC or their phone, making it a microcosm for great user-supplied evaluations. Amazon offers a variety of things such as books, phone applications, movies, apparel, gadgets, toys, and so on, and uses a star-based

rating system ranging from 1 to 5 stars (1=least, 5=most) and provides an option to write a review. This score system comes with no instructions on how to use it, and the product evaluations are subjective and personal.

As a result, a user might give an excellent product a "1" but have a bad user experience, such as no satisfaction with the quality or delivery compromise, and vice versa. The lack of rules makes identifying the user's feelings regarding various product elements and components of a purchasing experience challenging. Moreover, a "5" product review does not always correspond to the product review of an item. To gain more information about the product review, sentiment analysis is done.

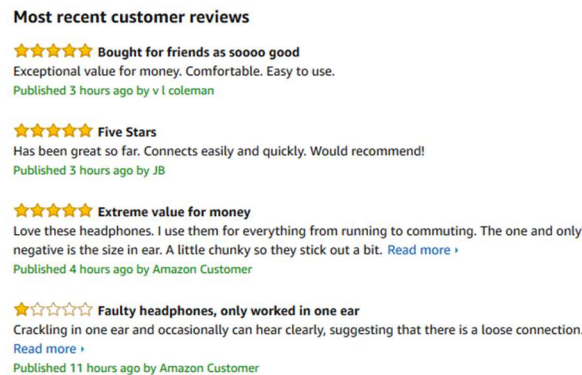


Figure 1.1: Sample Amazon Product Review

1.2 MOTIVATION

Sentiment analysis is critical process so to provide more easy and effective way to categorize the text reviews posted on amazon e-commerce website. There exist few techniques for sentiment analysis. To improve existing technique and to create a starting point in future works in natural language processing. To gain insight into ensemble and hybrid learning methods used in Natural language Processing.

1.3 PROBLEM DEFINITION AND OBJECTIVE

Amazon has functionality to post review of the product. Buyers can post review about the product. We are going to analyse reviews of amazon product. To analyse the reviews, machine learning and data analysis techniques will be used. In machine learning we are going to perform natural language processing on the text. The virtualization techniques is required to find the patterns in the reviews. In this project we will be learning all the techniques.

After the analysis the output will be used to provide to producer. Sentiment analysis can be used to improve the quality of the product and create awareness in online shopping for the buyers.

1.4 PROJECT SCOPE AND LIMITATION

The scope of the project is to develop a method that employs ML techniques to classify sentiment of review on product. It focuses on improving the feedback process of product and develop fast, better sentiment analysis model. The sentiment analysis can be further used in product recommendation system.

Public Datasets available have limited the scope to few products, The project is highly feasible because python has a big environment of well documented libraries. This makes the development process easy. In the future with greater access to quality text datasets the scope for this can be further increased to include movie and song reviews

Limitations can be stated as the only requirement for the application is images of crops and farms. Also, there are very few datasets available most of which are restricted to specific plants and weeds which limits the scope of the application.

CHAPTER 2

LITERATURE SURVEY

2.1 SENTIMENT ANALYSIS

Opinion mining, also known as sentiment analysis, is one of the studies under NLP research. To investigate people's opinions, it leverages textual data that is readily available on e-commerce sites like Amazon. It focuses on the theme area of the text—a word or a sentence—those points in a positive or negative direction. By offering businesses a thorough understanding of how customers feel about their products, SA plays a vital role in the commercial sphere. As a result, businesses may modify their strategies to meet customer expectations and requests and avoid loss. On the other hand, choosing the items you want to purchase might be helpful for potential buyers.

2.1.1. Sentiment Analysis: Degree

Sentiment analysis is often researched at three different degrees, depending on the text groups: a document which is a collection of sentences, a unique sentence, and lastly, a feature level. In a document, the goal is to determine if the overall tone of the language conveys a favourable or unfavourable emotion toward a certain entity. The Sentence level of analysis, in contrast, is concerned with determining if each sentence in the text carries a positive, negative, or neutral attitude. Item and aspect level analysis may be conducted, however the other levels cannot since they are focused only on identifying whether or not consumers like certain qualities. It is also known as feature level analysis and phrase-level sentiment analysis. It is used while doing sentiment analysis on evaluations of electrical devices and movies.

2.1.2. Approach

In practice, two primary traditional methodologies are applied in tackling sentiment analysis difficulties; Machine Learning & Lexicon based. Fig. 2. demonstrates the methodologies used in a collection of simple sentences based on customer reviews or remarks, to discern whether negative and positive comments are mentioned in that material. To improve the results, a hybrid approach of machine learning methods is used which combines more than two machine learning techniques.

2.2. MACHINE LEARNING

These methods deal with the problem of how text analysis may teach a computer programme to recognise intricate patterns and draw wise conclusions from data. Techniques for supervised and unsupervised learning make up the majority of it. While supervised techniques use ML classification algorithms, unsupervised methods make advantage of clusters that offer Lexicon approaches.

2.2.1. Supervised machine learning method

We focus largely on data classification and categorization in supervised learning. An algorithm typically requires a large labelled training dataset in order to be trained on the relationship between each word (or sequence) in a text and the overall conclusion of the sentence in a supervised way. Among other common supervised methods are Classification Tree DT, Naïve Bayesian NB, Maximum Entropy ME, and Support Vector Machine SVM. This method calls for manually labelling the data, which is usually time-consuming and not always practical.

2.2.2. Unsupervised machine learning method

In contrast, in the unsupervised approach, we concentrate on classifying unordered data based on commonalities or variations without providing the computer with any data training. It makes it possible to analyse the data without the requirement for human involvement using traditional unsupervised clustering types including Hierarchical, K-means, K Nearest Neighbours (KNN), Principal Component Analysis (PCA), and others. When there is a paucity of tagged data, this strategy is helpful. When hybrid learning or semi-supervised learning are used, these methods need some supervision of the output.

2.3. LEXICON BASED METHOD

This approach looks for the vocabulary that expresses the viewpoint and then evaluates it, for instance by using a dictionary of words and phrases that express the opinion as well as their synonyms and antonyms, as well as the associated emotion scales. Additionally, it is separated into dictionary-based and corpus-based approaches.

2.3.1. Dictionary based

WordNet, SentiWordNet, and online dictionaries are just a few examples of opinion dictionaries that often feature both positive and negative opinions. This approach looks for words with ambiguous meaning in the text, compares them to terms from the dictionary, and then calculates the appropriate scores. This approach cannot find views that are domain- or context-specific.

2.3.2. Corpus Based

In order to find domain- or context-specific views that dictionary-based techniques are unable to find, it finds opinionated keywords in the corpus and assigns polarity to all of these words. It calls for an English dictionary or a dictionary with a sizable word definition database. The algorithm must be able to access and retrieve it.

2.4. HYBRID MACHINE LEARNING

Hybrid Machine learning is a method where two or more machine learning algorithms are used together to obtain better results. Results of one model are used to augment the input to another model. This kind of ensemble learning improves the quality of data when it is fed to the classification model.

2.5. PREVIOUS WORK

At first glance, we begin by looking at related work which uses traditional supervised learning algorithms to calculate the performance of machine learning models. The algorithms that are in focus are Support Vector Machines SVM, Naive Bayes NB, and Decision Trees. The Ensemble Classifier beat the aforementioned machine learning algorithms when it was compared in [1] to others including logistic regression, SVM, Naive Bayes, Decision Tree, and Multinomial. In [2] In this Paper, authors used a combination of bigram mode with SVM so the hybrid algorithm gives the highest accuracy of 85%. In [3] the authors compare between two machine learning approaches which are SVM and NB for analysing the sentiment of the customers reviews on Amazon products. SVM offers a much greater accuracy and precision recall. The authors of [4] analyse the dataset of Amazon reviews and investigate sentiment categorization using several machine learning techniques. The reviews were first converted into word vectors using a variety of methods, including glove, Tf-Idf,

and bag-of-words. Then, they trained many machine learning algorithms, including bert, naive bias, bidirectional long-short memory and long - term, random forest, and logistic regression. The models were then assessed using cross-entropy gradient descent, precision, f1-score, accuracy, and recall. In [5], the authors examine pre-processing procedures on the dataset, such as stemming, tokenization, casing, stop-word removal, and eventually offer a rating for its categorization in negativity or positivity. In [6], we see a rise in accuracy of scores while using unstructured data. The model achieves an accuracy of 98% of Naive Bayes Algorithm and accuracy of 93% of SVM. In [7] the authors had done the context-based analysis for amazon products. The data was collected from amazon product site and pre-processed accordingly for analysis. They had used the naive Bayes and Support vector Machine models to classify the reviews and then perform the context-based analysis. Measures of performance i.e., precision, recall and F1 scores were calculated and on the basis of that models were compared. The area of work was to improve the sales based on the sentiments delineated, every product was considered whether it has positive or negative inclined reviews. In [8] the authors had done the sentiment analysis of products using machine learning. They had gathered the data from amazon product site for the following products: Camera's, Laptops, Tablets and Television's. The data is treated with Pre-processing technique. The pre-processing technique used is Bags of Words (BOW). The data then is used to train Naive Bayes and support vector machine classifiers to mould the models. Naive bayes classifier came up with 90% and above accuracies for each product whereas the support vector machine classifier performed dim with accuracies less than 90%. Thus, the Naive bayes was superior to SVM in sentiment analysis. The authors of [9] conducted a sentiment analysis of user reviews for Amazon items. They had gathered the information from the Amazon product page, performed some rudimentary pre-processing on it, and then utilised it right away for model training. Decision Tree, Naive Bayes, and Support Vector Machine were the algorithms used for the study. The writers of [10] had collected the information from the Amazon goods page. Following that, the data was analysed using review-level and sentence-

level classification. The categorizing method used was called "Phrase of Speech." The training of the model was then supplied with this data. The classification algorithms Naive Bayes and Support vector machine was taught. [11] describes a categorization method the authors developed for a dataset of music CDs and Microsoft goods that were scanned using a Python crawler. They looked at five different categories (Most Negative, Negative, Neutral, Positive, and Most Positive). The paper used three different types of adverbs as features, namely Adverbs RB, Comparative adverbs RBR, Superlative adverbs RRS, as well as a mixture of them, to achieve review level classification. Other classifiers included RF, DT, NB, SVM, GB, and LSTM classifiers. The analyses show that a single RBR feature is adequate for most classifiers, with the exception of LSTM and NB, and that a combination of RBR-RBS features is more effective for all classifiers. [12] They made use of the Amazon polarity dataset for their study. They have used deep learning models LSTM and CNN, SVM, and logistic regression. A sizable dataset had been used to test each model. The optimal combination approach was found to operate stemming over lemmatization and exclude spelling checking. They investigated and analysed several pre-processing strategies that increase accuracy. They used a variety of feature techniques, including their TF-IDF, bag-of-words, and n-grams. Moving on towards Hybrid Machine learning approaches where techniques such as ensemble learning is used to change NLP rules or augment input data. In [13] the authors performed ensemble learning compared to Naive Bayes and SVM. The ensemble method gave much better results while the other two suffered. In [14] technologies used are data cleaning and pre-processing. This paper dataset is used as relevant graphs. This dataset has the highest accuracy, almost 95.7%. In [15] the authors tried a hybrid rule-based approach to observe results of algorithms such as SVM, RF and NB. The hybrid rule-based approach got better results. [16] The authors used RF to form an ensemble of decision trees. The tree data structure was used with SVM to form a classifier model. The Hybrid model showed a 2% rise in accuracy. [17] The authors have revisited the RF ensemble method paired with SVM. They achieved a greater accuracy than [16] With the same dataset.

Bootstrap method was used as an extension of Random Forest. [18] The authors employed an ensemble learning method in data pre-processing where unigram, bigram and trigram with and without stop word removal was used. RF with unigram with stop word removal showed the best results. In [19] the researchers had used natural language processing on the Arabic language reviews on products. They had built the Recurrent Neural network of the sentiment analysis of those reviews. They had built the dataset of the Arabic language reviews. The model performs at the considerably efficiency of 85% on the given dataset which consists of 7480 test items. The model will behave more precisely when trained with the large data. Table 1,2 shows the comparison between different research approaches based on the literature review

CHAPTER 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 ASSUMPTIONS AND DEPENDENCIES

When considering a hybrid machine learning approach for sentiment analysis of Amazon products, there are several assumptions and dependencies to keep in mind. These assumptions and dependencies are important for developing and evaluating the model effectively. Here are some key points:

1. **Data Availability:** The assumption is that a labelled dataset of Amazon product reviews with sentiment annotations is available for training and evaluation purposes. The quality and size of the dataset can significantly impact the performance of the model.
2. **Feature Extraction:** The hybrid approach relies on extracting relevant features from the product reviews. The assumption is that effective feature extraction techniques, such as bag-of-words, word embeddings, or other domain-specific methods, can be applied to capture the sentiment-related information from the textual data.
3. **Supervised Learning:** The hybrid approach typically involves a combination of supervised learning algorithms. It assumes that a training phase can be conducted, where the model learns from the labeled data to generalize patterns and make predictions on unseen data. The availability of training resources and computational power to train the models efficiently is a dependency.
4. **Traditional ML Algorithms:** The hybrid approach may incorporate traditional machine learning algorithms, such as Naive Bayes, Support Vector Machines (SVM), or Random Forests, as a baseline or feature extraction technique. The assumption is that these algorithms are suitable for the task and have been effectively implemented.
5. **Deep Learning Algorithms:** The hybrid approach may also involve deep learning algorithms, such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), or Transformer models like BERT. These models assume the availability of appropriate architectures, pre-trained models, and computational resources for training and fine-tuning.
6. **Integration of ML Models:** The hybrid approach assumes that the outputs of multiple models can be combined effectively to obtain improved sentiment predictions. Techniques such as model stacking, ensemble learning, or weighted voting can be utilized. The assumption is that the integration process enhances the overall performance.
7. **Evaluation Metrics:** The evaluation of the hybrid model's performance requires assumptions about appropriate evaluation metrics. Common metrics for sentiment analysis include

accuracy, precision, recall, F1 score, and area under the ROC curve (AUC). The choice of metrics depends on the specific requirements and objectives of the sentiment analysis task.

8. **Domain Adaptation:** The sentiment analysis of Amazon products assumes that the model can effectively generalize to new and unseen product reviews. However, the model's performance might be influenced by domain-specific language, product categories, or evolving trends. Adapting the model to the specific domain or continuously updating it with new data may be necessary.
9. **Annotation Consistency:** The performance of the hybrid model heavily depends on the quality and consistency of the sentiment annotations in the training dataset. It assumes that the sentiment labels assigned to the reviews are accurate and reliable. Annotator biases or subjective interpretations of sentiment can introduce noise in the dataset.
10. **Evolution of Amazon:** The hybrid sentiment analysis model assumes that the characteristics of Amazon's product reviews and user sentiments remain relatively stable over time. Changes in Amazon's policies, user behaviour, or review patterns may impact the model's performance, requiring periodic retraining or adaptation.

3.2 FUNCTIONAL REQUIREMENTS

1. **Data Collection:** The system should be capable of gathering Amazon product reviews and associated metadata from the Amazon platform or other relevant sources. This may involve web scraping, API integration, or access to pre-existing datasets.
2. **Data Pre-processing:** The system should pre-process the raw data to ensure its suitability for analysis. This may involve tasks such as text cleaning, tokenization, stop-word removal, stemming or lemmatization, and handling of special characters or symbols.
3. **Feature Extraction:** The system should employ appropriate techniques to extract meaningful features from the pre-processed text. This may include methods like bag-of-words, word embeddings (e.g., Word2Vec or GloVe), or more advanced techniques like BERT embeddings.
4. **Model Training:** The system should provide functionality for training machine learning models using the labelled data. This involves selecting and configuring the appropriate algorithms (e.g., Naive Bayes, SVM, CNN, RNN, BERT) and optimizing their hyperparameters.
5. **Hybrid Model Integration:** The system should support the integration of multiple machine learning models to create a hybrid approach. This may involve combining the outputs of

individual models using techniques like model stacking, ensemble learning, or weighted voting.

6. **Model Evaluation:** The system should have mechanisms to evaluate the performance of the sentiment analysis models. This includes measuring metrics such as accuracy, precision, recall, F1 score, AUC, or any other relevant evaluation metrics specific to sentiment analysis.
7. **Real-time Analysis:** The system should allow for real-time sentiment analysis of Amazon product reviews. This means it should be capable of handling new incoming reviews and providing sentiment predictions in near real-time.
8. **Sentiment Visualization:** The system should provide visualizations or summaries of sentiment analysis results, allowing users to easily interpret and understand the sentiment patterns of Amazon products. This may include sentiment distribution charts, word clouds, or sentiment trends over time.
9. **Customization and Adaptability:** The system should allow users to customize and adapt the sentiment analysis models based on specific requirements. This may involve fine-tuning pre-trained models, incorporating domain-specific knowledge, or adjusting thresholds for sentiment classification.
10. **Integration and Deployment:** The system should be deployable as a service or software component, allowing it to be easily integrated into existing applications or workflows. It should have well-defined APIs or interfaces for seamless integration.
11. **Scalability:** The system should be scalable to handle large volumes of data and user requests. It should be able to efficiently process and analyse a significant number of Amazon product reviews without compromising performance.
12. **Model Maintenance and Updates:** The system should support periodic retraining or updating of the sentiment analysis models to adapt to changing review patterns, user sentiments, or new product categories on Amazon.
13. **Security and Privacy:** The system should adhere to security and privacy best practices, ensuring the protection of sensitive data, compliance with relevant regulations (e.g., GDPR), and appropriate access control measures.

3.3 EXTERNAL INTERFACE REQUIREMENTS

3.3.1 User Interface

1. Front-end and Back-end technologies:

Front-end Technologies:

1. HTML: Hypertext Markup Language is the standard markup language for creating the structure and content of web pages.
2. CSS: Cascading Style Sheets is used to style and format the appearance of HTML elements on a webpage.

Back-end Technologies:

1. Python: Python is a popular programming language that provides a wide range of libraries and frameworks for web development and data processing.
2. Flask: Flask is a lightweight web framework in Python that allows you to build web applications quickly and easily.
3. BeautifulSoup: BeautifulSoup is a Python library for parsing HTML and XML documents, which can be useful for extracting information from web pages.
4. Pickle: Pickle is a Python module that allows you to serialize and deserialize Python objects, making it useful for storing and loading trained machine learning models.
5. Requests: The Requests library is used to send HTTP requests in Python, making it useful for fetching data from web APIs or scraping web pages.
6. Scikit-learn: Scikit-learn is a popular machine learning library in Python that provides various algorithms and tools for machine learning tasks, including sentiment analysis.
7. Pandas: Pandas is a powerful data manipulation library in Python that provides data structures and functions for efficiently working with structured data.
8. Matplotlib: Matplotlib is a plotting library in Python that allows you to create various types of visualizations, such as line plots, bar charts, histograms, etc.

2. Hardware:

1. **Server/Hosting:** You will need a server or hosting infrastructure to deploy the web application. The hardware specifications of the server will depend on factors such as the number of concurrent users, expected traffic, and computational requirements. It is recommended to have a server with sufficient CPU and memory resources to handle the expected workload.
2. **CPU:** The sentiment analysis process, especially when using machine learning algorithms, can be computationally intensive. Having a powerful CPU or a server with multiple cores can help in processing the data efficiently and reducing the response time.
3. **Memory:** Sufficient memory (RAM) is important to store and process the data efficiently. The amount of memory required will depend on the size of the dataset, the complexity of the machine learning models, and the number of concurrent requests the system needs to handle. It is advisable to have enough memory to avoid performance bottlenecks.
4. **Storage:** Depending on the scale of the application, you might need storage to store the dataset, trained models, and other relevant data. The storage requirements will depend on the size of the dataset and the number of product reviews you plan to process.
5. **Network:** A stable and reliable network connection is crucial for handling incoming requests and providing responses in a timely manner. Ensure that the network infrastructure can handle the expected traffic and has sufficient bandwidth.
6. **Scalability:** If you anticipate significant growth in users or data volume over time, it is important to consider a scalable infrastructure. This may involve using cloud services or technologies that allow for horizontal scaling, such as load balancers or distributed computing frameworks.
7. **Backup and Redundancy:** Implementing backup and redundancy measures is essential to ensure data integrity and system availability. This may involve regular backups of datasets and trained models, as well as redundancy in server infrastructure to minimize downtime.

3.3.2 Communication Interfaces

1. **Web Interface:** A web interface allows users to interact with the sentiment analysis system through a web browser. It provides the user interface (UI) for inputting text data, displaying results, and visualizations. HTML, CSS, and JavaScript are commonly used to design and develop the web interface.
2. **API (Application Programming Interface):** APIs provide a standardized way for different software components to communicate with each other. The sentiment analysis system can expose APIs that allow external systems or applications to interact programmatically. This enables integration with other services, such as retrieving data from external sources or providing sentiment analysis results to other applications. APIs can be built using technologies like REST (Representational State Transfer) or GraphQL.
3. **Data Sources:** Communication interfaces are required to fetch data from relevant sources such as Amazon product reviews or external databases. These interfaces can involve web scraping techniques using libraries like BeautifulSoup or accessing data via APIs provided by the data source.
4. **File I/O:** Communication interfaces for file input/output (I/O) enable reading and writing data to files. This can be useful for loading datasets, saving trained models, or exporting analysis results to files in various formats. Python provides built-in modules such as **csv**, **pickle**, or **json** for handling file I/O operations.
5. **External Services Integration:** If the sentiment analysis system needs to integrate with external services, such as third-party APIs for additional data sources or services, communication interfaces are necessary to establish connections and exchange data. This can involve using libraries or SDKs provided by the external services to interact with their APIs.
6. **Real-time Communication:** If real-time analysis or feedback is required, communication interfaces such as websockets can be utilized to establish a bidirectional communication channel between the server and the web browser. This allows for real-time updates and notifications.

3.4 NON-FUNCTIONAL REQUIREMENTS:

3.4.1 Performance Requirements:

1. **Response Time:** The system should be capable of providing sentiment analysis results within an acceptable response time. The response time can vary depending on the complexity of the analysis and the expected workload. It is important to define specific response time targets to meet user expectations.
2. **Scalability:** The system should be designed to scale horizontally or vertically to accommodate increasing workloads. Horizontal scaling involves adding more servers or instances to distribute the load, while vertical scaling involves upgrading the hardware resources of existing servers. The system should be able to handle increased traffic or data volume without significant degradation in performance.
3. **CPU Utilization:** The system should make efficient use of CPU resources to process the sentiment analysis tasks in a timely manner. Optimized algorithms and code should be implemented to minimize CPU usage and maximize processing efficiency.
4. **Load Testing:** Performance testing and load testing should be conducted to assess the system's performance under realistic workloads. This involves simulating concurrent users or high volumes of requests to identify potential performance bottlenecks and fine-tune the system for optimal performance.
5. **Error Handling:** The system should be designed to handle errors and exceptions gracefully. Proper error handling mechanisms should be implemented to prevent system failures or performance degradation in the event of unexpected scenarios or exceptions.

3.4.2 Safety Requirements:

1. **Data Security:** The system should have measures in place to protect sensitive data, including customer reviews, user information, and any other confidential or personally identifiable information. This may involve encryption of data in transit and at rest, access control mechanisms, and secure storage practices.
2. **User Privacy:** The system should respect user privacy and adhere to relevant privacy regulations, such as GDPR (General Data Protection Regulation). It should clearly communicate the data handling practices, obtain necessary user consent, and provide options for users to control their data.

3. **System Reliability:** The system should be designed to ensure high reliability and minimize the risk of system failures. Redundancy, fault tolerance, and backup mechanisms should be implemented to mitigate the impact of hardware or software failures.
4. **Compliance:** The system should comply with relevant industry standards, regulations, and legal requirements. This may include data protection regulations, intellectual property rights, or specific industry-specific regulations, such as those in the healthcare or financial sectors.
5. **User Safety:** The system should prioritize user safety by ensuring that the sentiment analysis results and recommendations provided are accurate and reliable. Any potential biases or limitations of the analysis should be communicated transparently to users.

3.4.3 Software Quality Attributes:

1. **Reliability:** Reliability refers to the ability of the software system to perform its intended functions consistently and accurately. A reliable system should operate without failures, errors, or unexpected behaviors. It should be available for use when needed and should be able to recover from faults or errors gracefully.
2. **Usability:** Usability focuses on the ease of use and user-friendliness of the software system. A usable system should be intuitive, well-designed, and provide a positive user experience. It should have clear and understandable user interfaces, provide helpful guidance and feedback, and be accessible to a wide range of users.
3. **Performance:** Performance refers to the system's ability to meet specific performance requirements and provide timely responses. This includes aspects such as response time, throughput, scalability, and resource utilization. A high-performance system should be able to handle expected workloads efficiently and deliver optimal performance under varying conditions.
4. **Maintainability:** Maintainability measures the ease with which the software system can be maintained, modified, or extended over time. A maintainable system is characterized by well-structured and modular code, clear documentation, and appropriate use of coding standards and best practices. It should facilitate efficient debugging, troubleshooting, and future enhancements without introducing unintended side effects.

3.5 SYSTEM REQUIREMENTS

3.5.1 Software Requirements

1. Web Browser:

- The system should be compatible with popular web browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.
- The system's user interface should be designed and tested to ensure proper rendering and functionality across different web browsers.
- It may be necessary to consider specific versions or browser settings to ensure compatibility and consistent user experience.

2. Python:

- The system should be developed using Python as the primary programming language.
- The system should be compatible with the specific version(s) of Python intended for use. For example, Python 3.x is commonly used for new projects, with Python 3.7 or later versions being popular choices.
- Any external libraries or dependencies required for the sentiment analysis system should be compatible with the chosen version of Python.
- The system may rely on web scraping libraries, such as BeautifulSoup, to retrieve product reviews from the web. Ensure that these libraries are compatible with the chosen Python version.
- Flask, a popular web framework, can be used for building the back-end of the system. The Flask library should be compatible with the chosen Python version.
- Other libraries such as pandas, scikit-learn, matplotlib, and pickle may be used for data processing, machine learning, visualization, and model persistence. Ensure that these libraries are compatible with the chosen Python version.

3. Operating System:

- The sentiment analysis system should be compatible with the operating system(s) commonly used by the target users, such as Windows, macOS, or Linux distributions like Ubuntu.
- The required software components, including the web browser and Python, should be available and supported on the chosen operating system(s).

4. Development Tools:

- Depending on the specific development environment and preferences, an integrated development environment (IDE) such as PyCharm, Visual Studio Code, or Jupyter Notebook can be used for coding, debugging, and testing.
- Version control systems like Git can be utilized for efficient code management and collaboration.

3.5.2 Hardware Requirements

1. **Processor:** The system should have a processor with sufficient processing power to handle the computational requirements of the sentiment analysis tasks. A multi-core processor with a clock speed of at least 2.0 GHz or higher is recommended to ensure efficient processing of large amounts of text data.
2. **Memory (RAM):** Sufficient memory is essential for handling the data processing and model training tasks. The amount of RAM required depends on the size of the dataset and the complexity of the machine learning models used. As a general guideline, a minimum of 8 GB of RAM is recommended, but more may be necessary for larger datasets and more complex models.
3. **Storage:** Adequate storage space is needed to store the sentiment analysis system, the dataset, and any associated libraries or dependencies. The required storage capacity will depend on the size of the dataset and the amount of data to be processed. Additionally, having solid-state drives (SSDs) can significantly improve the system's performance, as they provide faster data access and retrieval compared to traditional hard disk drives (HDDs).
4. **Network Connectivity:** The system should have a stable and reliable internet connection to fetch product reviews from Amazon or access any external APIs. A high-speed internet connection is recommended to ensure efficient data retrieval and processing.
5. **Graphics Processing Unit (GPU):** While not strictly necessary, having a dedicated GPU can greatly accelerate the training and inference processes of machine learning models. GPUs are particularly useful for deep learning algorithms that heavily rely on matrix computations. If using deep learning models, a GPU with CUDA support, such as NVIDIA GeForce or Tesla GPUs, can provide significant performance improvements.
6. **Operating System:** The sentiment analysis system can be developed and deployed on various operating systems, including Windows, macOS, or Linux distributions like Ubuntu. Choose an operating system that is compatible with the required software components and libraries.

7. Other Peripherals: Depending on the specific implementation and user requirements, additional peripherals such as monitors, keyboards, and mice may be required to interact with the system effectively.

3.6 ANALYSIS MODEL:

Agile model

Agile Methodology is used to adapt to changes fast and efficiently. Its main goal is to facilitate quick project completion. In the Agile model the requirements are decomposed into small parts that are developed incrementally. These are the following phases:

1. Concept

First is the concept phase. Here we determine the scope of the project. We discussed key requirements and prepared documentation to outline them, including what features will be supported and the proposed end results. We kept the requirements to a minimum as they can be added to in later stages. This detailed analysis helped us to decide whether a project is feasible.

2. Inception

Once the concept is outlined, we started with software development planning. We started the design process. We planned and drew some sample mockup user interface and built the project architecture. The inception stage helped us determine the product functionality.

3. Iteration

Next up is the iteration phase. It is the longest phase as the bulk of the work is carried out here. We will work on UX to combine all product requirements and turn the design into code. The goal is to build the bare functionality of the product by the end of the first iteration or sprint. Additional features and tweaks can be added in later iterations.

4. Release

The product is almost ready for release. But for quality assurance needs to perform some tests to ensure the software is fully functional. The team members will test the system to ensure the code is clean if potential bugs or defects are detected, the developers will address them swiftly.

5. Maintenance

The software will now be fully deployed and made available to customers. This action moves it into the maintenance phase. During this phase, the software development team will provide ongoing support to keep the system running smoothly and resolve any new bugs.



Figure No. 3.1 Agile Model

CHAPTER 4

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

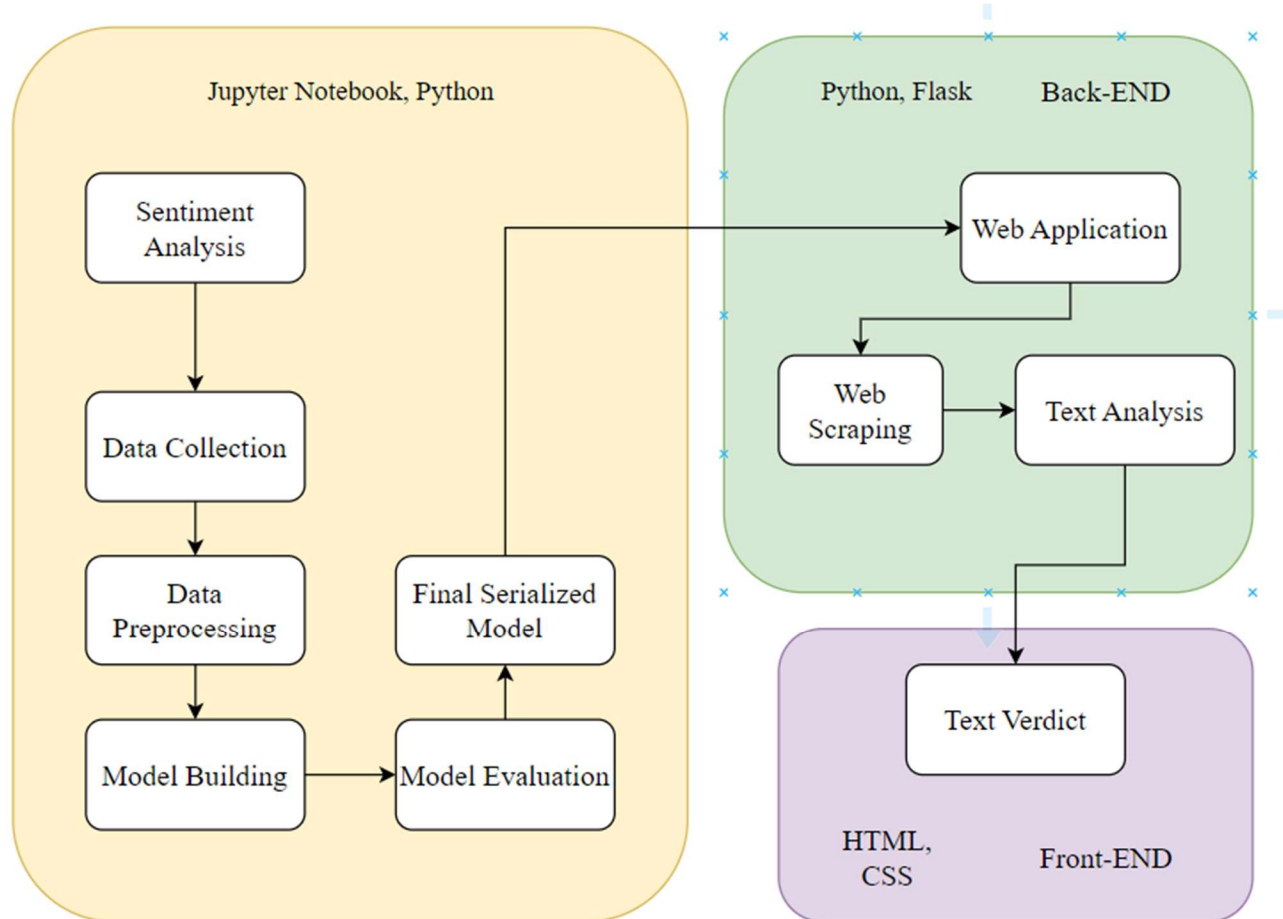


Figure No. 4.1 System Architecture diagram

4.2 MATHEMATICAL MODEL

Sentiment Analysis and classification

- Input: Product review in text format.
- Output: Positive or Negative response
- Algorithms: Natural Language Processing and Hybrid Machine Learning Model
- Mathematical Formulation:
- System = {Train, Test, classification}
- Train = {pre-process, feature extract, classification}
- Test = {pre-process, feature extract, classification}
- Object detection = {response to text review in Positive or negative category}
- Success condition: If we train the model successfully on an unbiased dataset without any issues then we get accurate output. Thus, model will classify the review in positive or negative category successfully.
- Failure condition: If the dataset is biased and bad training can result in reduced accuracy

4.4 UML DIAGRAMS

4.4.1 Activity Diagram:

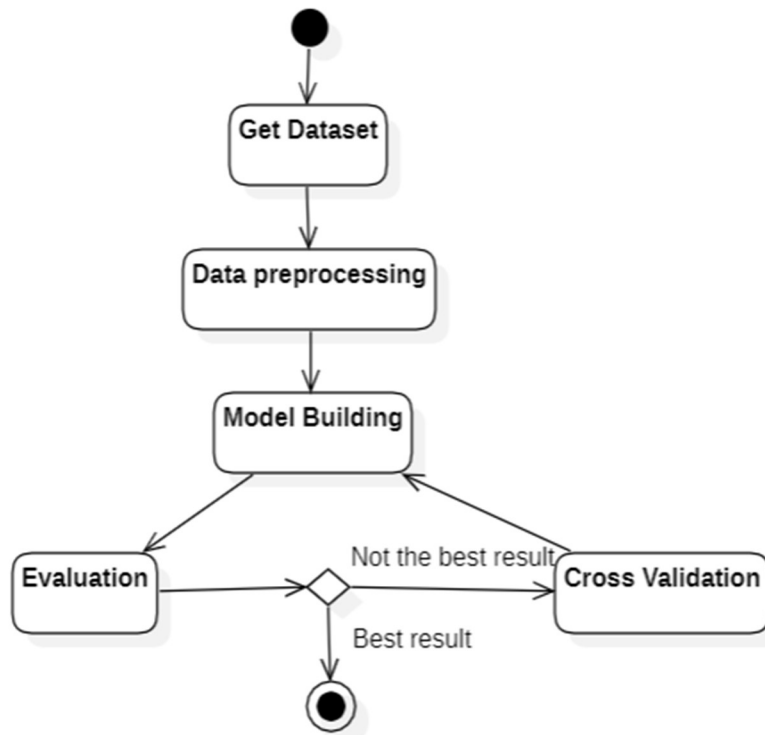


Figure 4.4.1 Activity Diagram

4.4.2 Model Building

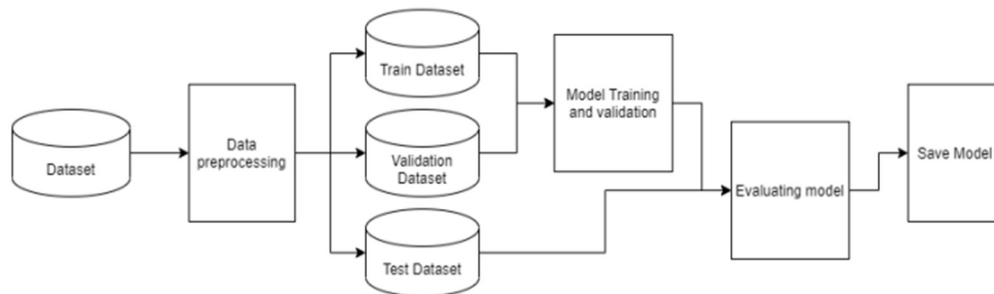


Figure 4.4.2 Model Building

4.4.3 Class Diagram:

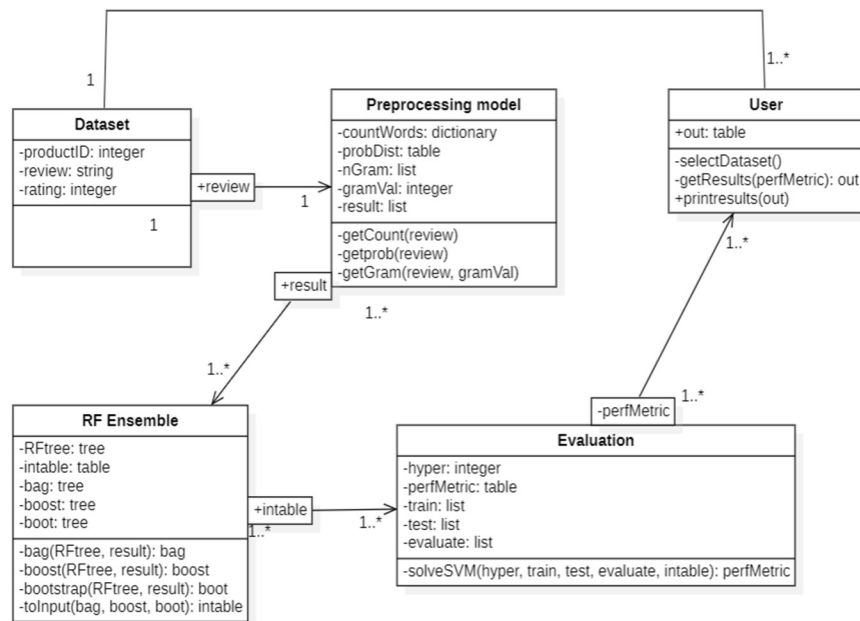


Figure 4.4.3 Class diagram

4.4.4 State Diagram:

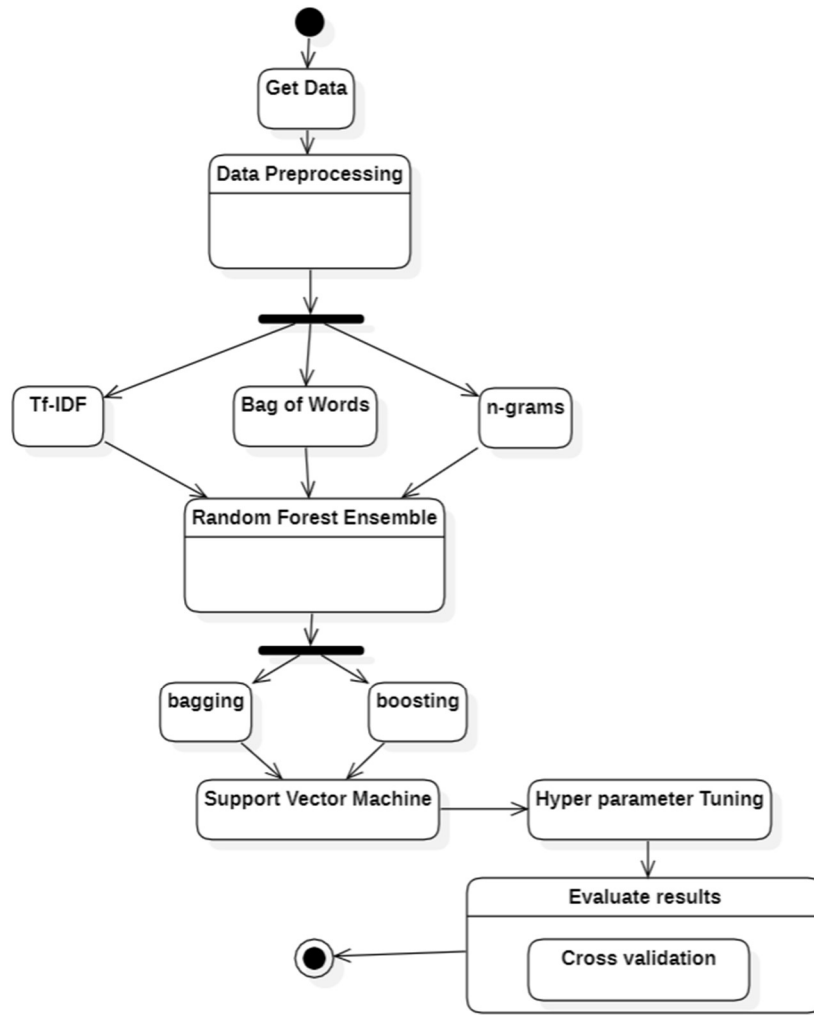


Figure 4.4.4 State Diagram

CHAPTER 5

PROJECT ESTIMATE

5.1. PROJECT ESTIMATE

It is the process of analyzing available data to predict the time, cost, and resources needed to complete a project.

5.1.1 Reconciled Estimates

- For reconciling estimates, we mainly focused on four steps.
- We first estimated the size of the development product.
- Also, for better planning and scheduling, we estimated the schedule over the calendar.
- We also estimated the project cost for its development.

5.1.2 Project Resources

- Project resources simply mean resources that are required for the successful development and completion of a project. These resources can be capital, people, material, tools, or supplies that are helpful to carry out certain tasks in a project.
- Resources can be categorized into three parts represented as a resource pyramid. These comprise people, reusable software components, and hardware and software tools.
- Hardware and software tools were planned initially before the development of the product.
- The reusable resources also known as cost resources are very helpful as they help in reducing the overall cost of development

5.2. RISK MANAGEMENT

Risk Management is an important part of project planning activities. It involves identifying and estimating the probability of risks with their order of impact on the project.

5.2.1 Risk Identification

- Risk identification involves brainstorming activities. It also involves the preparation of a risk list.
- Basically, for this system we identified the risks associated with it.
- The risk list was prepared by analyzing from the stakeholders' points of view.

5.2.2 Risk Analysis

- For risk analysis, we analyzed the risk list.
- Also, we analyzed the occurrence of a particular risk with its probability.
- The table was prepared to consist of values and order risk based on exposure factor.

5.2.3 Overview of Risk Mitigation, Monitoring, and Management

- The purpose of this technique is to altogether eliminate the occurrence of risks. so the method to avoid risks is to reduce the scope of projects by removing non-essential requirements.
- In risk monitoring, the risk is monitored continuously by reevaluating the risks, the impact of risk, and the probability of occurrence of the risk.

5.3 PROJECT SCHEDULE

5.3.1 Project Task Set

- Requirements gathering
 - The first and foremost task was requirements gathering.
 - We understood the need for the system.
 - Gathered knowledge of tools and technologies that will be used to implement the system.
- Deciding on the project modules

For project implementation 5 transfer learning algorithms were decided for comparison

 1. Support Vector Machine
 2. Random Forests
 3. Logistic regression
 4. K Nearest Neighbors
 5. Decision Tree Classifiers
- Dataset gathering

Publicly available datasets for Pneumonia detection were gathered from Kaggle
- Implementing project modules

Project modules were implemented by making ML models sentiment analysis

5.3.2 Timeline Chart

Schedule		Project activity
July	3 rd Week	Formation of the project group
	4 th Week	Project Topic Selection
August	1 st Week	Synopsis Writing

	2 nd Week	Synopsis Submission
	3 rd Week	Literature Survey
September	1 st Week	Survey Paper Writing
	3 rd Week	Feasibility Analysis
October	1 st Week	Mid Sem Presentation
	2 nd Week	Project Module Discussion
	3 rd Week	Project Module Finalization
November	2 nd Week	Report preparation and submission
	3 rd Week	Project stage-I exam
January	1 st Week	Discussion about further strategy for project module implementation
	3 rd Week	Algorithm implementation
February	2 nd Week	Algorithm implementation
	4 th Week	Result analysis
March	1 st Week	Project implementation improvements
	3 rd Week	Research paper finalization
April	2 nd Week	Report Writing
May	1 st Week	Report finalization and submission

Table No. 5.3: Timeline Chart

5.4 TEAM ORGANIZATION

5.4.1 Team Structure

1. Om Sarulkar
2. Shivam Tikhe
3. Sumit Giri
4. Rohan More

5.4.2 Management Reporting and Communication

- Weekly Team Meetings: Set up regular weekly team meetings to go over the TTS paper's development's progress, challenges, and next actions. Team members can discuss updates, trade ideas, and ask the community for advice or direction at these meetings.
- Task Assignment and Tracking: Assign duties and responsibilities to each team member in detail. Maintain a task tracking system to keep track of task status, due dates, and any dependencies, such as a project management application or shared document. During team meetings, periodically examine the work tracking system to make sure everyone is on track.
- Progress Reports: Encourage team members to produce regular progress reports, such as bi-weekly or monthly, depending on the project timetable. These reports should summarize the work done, the problems encountered, and any notable results or insights. This allows management to stay up to date on progress and make educated decisions.
- Document Sharing and Collaboration: Utilize a shared document or version control system to facilitate collaboration on the paper development process. This allows team members to work

simultaneously, provide feedback, and track changes effectively. Regularly update and share the document with the team and management for review and input.

- Communication Channels: Maintain efficient group communication methods, such as a dedicated messaging platform or email thread. Encourage open communication so that team members may ask questions, seek clarification, and provide timely updates. Check and respond to messages on a regular basis to establish a productive and collaborative work atmosphere.
- Management Updates: Update the management team on the status of the TTS paper development on a regular basis. This can take the form of executive summaries, presentations, or meetings where the team presents key findings, accomplishments, and future scope. These updates make ensuring that the management is knowledgeable and prepared to offer direction or help as required.
- Feedback and Review Process: Establish a feedback and review process within the group to ensure the quality and rigor of the paper. Encourage team members to review each other's work, provide constructive feedback, and engage in peer discussions to enhance the overall quality of the paper. Regularly seek feedback from management on the direction and content of the paper to align with their expectations and requirements.
- By implementing effective management reporting and communication practices, the paper development group can ensure transparency, accountability, and collaboration, leading to successful outcomes and a high-quality research paper.

CHAPTER 6

PROJECT IMPLEMENTATION

6.1 OVERVIEW OF PROJECT MODULES

1. Data Collection Module:

- This module focuses on downloading Amazon product reviews from the Amazon website.
- It involves web scraping using libraries like BeautifulSoup to extract reviews from web pages.
- The collected data is stored in a suitable format (e.g., CSV or JSON) for further processing.

2. Data Pre-processing Module:

- This module handles the pre-processing of the collected data.
- It involves tasks such as cleaning the text data, removing noise, handling missing values, and performing feature engineering.
- Libraries like Pandas and NumPy are used for efficient data manipulation and pre-processing.

3. Model Training Module:

- This module focuses on training machine learning models using the pre-processed data.
- It involves splitting the data into training and testing sets, selecting appropriate features, and training different models.
- Scikit-learn is used to train models such as Support Vector Machine, Random Forests, K Nearest Neighbours, and Decision Tree Classifiers.

4. Model Evaluation Module:

- This module is responsible for evaluating the trained models and selecting the best-performing one.

- It involves metrics like accuracy, precision, recall, and F1 score to assess the performance of each model.
- The models are tested on the testing data to compare their performance and choose the model with the highest accuracy.

5. Model Persistence Module:

- This module handles the saving of the selected model for future use.
- The chosen SVM model is serialized using the Pickle library and saved as a pickle file.
- This allows the model to be loaded and used later without retraining.

6. Web Application Module:

- This module focuses on building a local web application that utilizes the trained SVM model for sentiment classification of online reviews.
- Flask, along with HTML and CSS, is used to develop the web application.
- The web app takes an Amazon product URL as input, scrapes online reviews using BeautifulSoup, and classifies them as positive or negative using the loaded SVM model.

6.2. TOOLS AND TECHNOLOGIES USED

1. Python: A high-level programming language known for its simplicity and readability. Python was used as the primary programming language for implementing the project.
2. Pandas: A Python library used for data manipulation and analysis. It provides data structures like Data Frames that make it easier to work with structured data.
3. NumPy: A Python library for scientific computing. NumPy provides support for large, multi-dimensional arrays and mathematical functions to operate on these arrays efficiently.
4. Matplotlib: A Python library for creating visualizations such as line plots, bar charts, histograms, and more. It is often used in conjunction with Pandas for data visualization.

5. Scikit-learn: A popular Python library for machine learning. Scikit-learn provides a wide range of machine learning algorithms, tools for data preprocessing, model evaluation, and utilities for model selection and hyperparameter tuning.
6. Jupyter Notebook: An interactive coding environment that allows you to create and share documents containing code, visualizations, and explanatory text. It is commonly used for data analysis and prototyping machine learning models.
7. Flask: A Python web framework used for building web applications. Flask provides tools and libraries to handle HTTP requests, route URLs, and render dynamic HTML templates.
8. HTML: Hypertext Markup Language is the standard markup language for creating web pages. It provides the structure and content of a webpage.
9. CSS: Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in HTML. CSS defines how HTML elements should be displayed on the webpage.
10. BeautifulSoup: A Python library used for web scraping. BeautifulSoup helps extract data from HTML and XML documents, making it useful for extracting online reviews from web pages.
11. Requests: A Python library used for making HTTP requests. It simplifies the process of sending HTTP requests to web servers and handling the responses in Python.
12. Pickle: A Python module used for object serialization. Pickle allows you to save and load Python objects to and from disk, making it convenient for storing and retrieving trained machine learning models.

6.3 ALGORITHM DETAILS

6.3.1 Support Vector Machine

SVM is a popular supervised learning algorithm used for both classification and regression tasks. It works by finding an optimal hyperplane that separates the data points of different classes in the feature space. SVMs can handle high-dimensional data and are effective in cases where the data is not linearly separable by transforming the data into a higher-dimensional space using a kernel trick

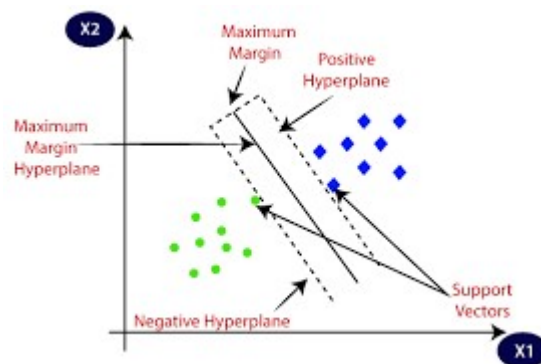


Figure No. 6.1 SVM Architecture

6.3.2 Random Forest

Random Forests is an ensemble learning method that combines multiple decision trees to make predictions. It creates a set of decision trees on randomly selected subsets of the data and aggregates their predictions to produce the final result. Random Forests can handle high-dimensional data, are resistant to overfitting, and can provide feature importance rankings.

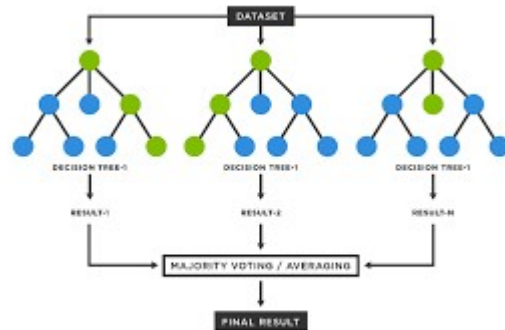


Figure No. 6.2 Random Forest Architecture

6.3.3 Logistic Regression

KNN is a simple yet effective supervised learning algorithm used for classification and regression tasks. Given a new data point, it looks for the k nearest data points in the training set based on a distance metric (e.g., Euclidean distance) and assigns the majority class label or average value of the k neighbours as the prediction. KNN does not build an explicit model but relies on the local neighbourhood of points for classification.

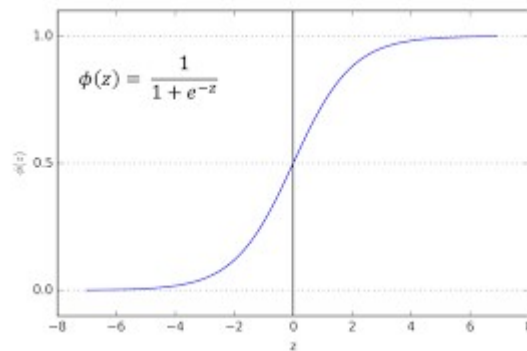


Figure No. 6.3 LR Architecture

6.3.4 Decision Tree Classifier

Decision trees are a type of supervised learning algorithm that partitions the feature space based on a sequence of if-else questions and creates a tree-like model of decisions. Each internal node represents a question on a feature, and each leaf node represents a class label or a regression value. Decision trees can handle both categorical and numerical data and are interpretable. They can also be used as a basis for ensemble methods like Random Forests.

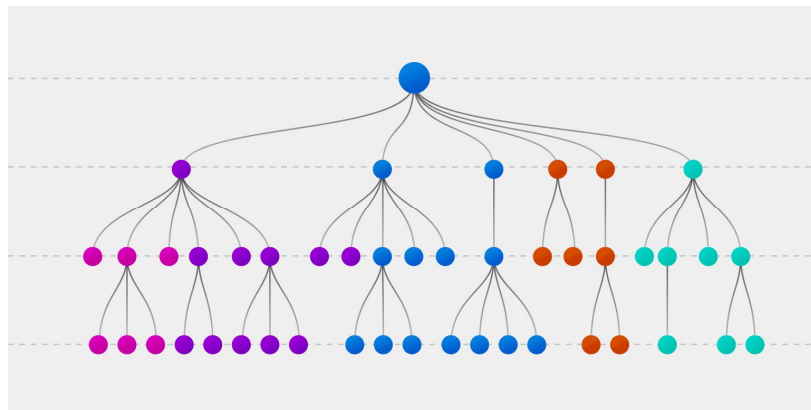


Figure No. 6.4 Decision tree Architecture

CHAPTER 7

SOFTWARE TESTING

7.1 TYPES OF TESTING

Functional Testing: Functional testing is a type of software testing that verifies whether the system meets the specified functional requirements. In the context of this project, functional testing can be applied to ensure that the web application and its associated functionalities work as expected. Here's a brief description of how functional testing can be applied:

1. **Input Validation:** Test the web application's input validation by providing various inputs, including valid and invalid URLs, to verify that it correctly handles different scenarios and displays appropriate error messages.
2. **Review Classification:** Validate the sentiment classification functionality by inputting a set of reviews and ensuring that the web application accurately classifies them as positive or negative.
3. **Web Scraping:** Test the web scraping functionality by providing different Amazon product URLs and verifying that the web application successfully extracts relevant reviews from the web pages.
4. **User Interface:** Verify the user interface elements, such as buttons, forms, and result displays, to ensure they are visually appealing, intuitive, and responsive across different devices and browsers.
5. **Navigation and Flow:** Test the flow of the web application by navigating through different pages, submitting forms, and ensuring that the expected actions and transitions occur without errors.

User Acceptance Testing (UAT): User Acceptance Testing is performed to determine whether a system meets the user's requirements and expectations. It involves validating the system's functionality, usability, and compatibility with real-life scenarios. Here's a brief description of how user acceptance testing can be applied in this project:

1. **Test with Real Users:** Involve actual users who represent the target audience to interact with the web application. They should perform common tasks, such as inputting URLs, reviewing results, and providing feedback on their experience.
2. **Usability Testing:** Evaluate the web application's usability by observing users as they navigate through the application, complete tasks, and provide feedback on its ease of use, clarity of instructions, and overall user-friendliness.

3. **Compatibility Testing:** Ensure that the web application works well across different browsers, devices, and screen sizes commonly used by the target audience. Test the application on multiple platforms to ensure consistent performance and compatibility.
4. **Performance Testing:** Validate the web application's performance by simulating realistic user loads and monitoring its response time, resource utilization, and scalability to ensure it can handle expected user traffic effectively.
5. **Feedback and Iteration:** Gather feedback from users and stakeholders throughout the testing process. Incorporate their suggestions and iterate on the application to address any identified issues or concerns, improving the overall user acceptance.

7.2 Test Cases

1. **Web Application User Interface Test:**
 - **Test Case:** Verify that the web application has a user-friendly interface with proper input fields, buttons, and styling.
 - **Test Result:** The web application displays an intuitive user interface with clear input fields and buttons, styled using CSS.
2. **Web Application Input Validation Test:**
 - **Test Case:** Validate the input provided by the user, specifically the Amazon product URL.
 - **Test Result:** The web application checks if a valid Amazon product URL is entered and displays an error message for invalid inputs.
3. **Web Scraping Test:**
 - **Test Case:** Scrape online reviews using the provided product URL.
 - **Test Result:** The web application successfully extracts relevant reviews from the web page using the BeautifulSoup library.
4. **Review Classification Test:**
 - **Test Case:** Classify the scraped reviews as positive or negative using the loaded SVM model.
 - **Test Result:** The web application applies the SVM model to the scraped reviews and accurately classifies them as positive or negative.
5. **Web Application Output Test:**
 - **Test Case:** Verify that the web application displays the classified reviews in an organized and readable format.

- Test Result: The web application presents the classified reviews clearly, with positive and negative reviews clearly distinguished.
6. Robustness Test:
- Test Case: Check how the web application handles unexpected inputs or errors.
 - Test Result: The web application gracefully handles errors or unexpected inputs, displaying informative error messages or providing fallback options.
7. Performance Test:
- Test Case: Evaluate the response time of the web application when processing and classifying a large number of reviews.
 - Test Result: The web application responds quickly and efficiently, even when dealing with a substantial amount of review data.
8. Cross-Browser Compatibility Test:
- Test Case: Verify that the web application functions correctly across different web browsers (e.g., Chrome, Firefox, Safari).
 - Test Result: The web application is tested on various browsers, and it displays and functions consistently across all supported browsers.
9. Responsiveness Test:
- Test Case: Test the web application's responsiveness and adaptability to different screen sizes (e.g., desktop, tablet, mobile).
 - Test Result: The web application adjusts its layout and design appropriately to provide an optimal user experience on different devices.
10. Integration Test:
- Test Case: Execute end-to-end testing of the web application, including input validation, scraping, classification, and output display.
 - Test Result: The complete web application workflow operates smoothly, producing accurate and well-presented results for the user

CHAPTER 8

RESULTS

8.1 OUTCOMES

In this project, a dataset consisting of 1000 labelled Amazon product reviews was used to train four machine learning models: Support Vector Machine (SVM), Random Forests, K Nearest Neighbours, and Decision Tree Classifiers. Out of the 1000 data points, 490 were labelled as positive and the rest as negative.

After training and evaluating the models, the SVM model emerged as the top performer with a remarkable accuracy of 90%. This high accuracy was achieved through hyperparameter tuning, coupled with cross-validation techniques, to optimize the model's performance.

To showcase the project's functionality, a local web application was developed using Flask, HTML, and CSS. The web application allows users to input an Amazon product URL, scrape online reviews using BeautifulSoup, and classify them as positive or negative using the trained SVM model. The output of the web application can be viewed at the following link: [Link to UI Output](#).

These results demonstrate the efficacy of the SVM model in accurately classifying sentiment in Amazon product reviews. The web application provides a user-friendly interface for users to analyse the sentiment of reviews and make informed purchasing decisions based on customer feedback.

	label	review
0	pos	Stuning even for the non-gamer: This sound tra...
1	pos	The best soundtrack ever to anything.: I'm rea...
2	pos	Amazing!: This soundtrack is my favorite music...
3	pos	Excellent Soundtrack: I truly like this soundt...
4	pos	Remember, Pull Your Jaw Off The Floor After He...

Fig 8.1 Dataset used for training model

	precision	recall	f1-score	support
neg	0.840000	0.880000	0.860000	1649.000000
pos	0.870000	0.830000	0.850000	1651.000000
accuracy	0.850000	0.850000	0.850000	0.850000
macro avg	0.850000	0.850000	0.850000	3300.000000
weighted avg	0.850000	0.850000	0.850000	3300.000000

Fig 8.2 Logistic Regression Evaluation metrics

	precision	recall	f1-score	support
neg	0.860000	0.890000	0.870000	1649.000000
pos	0.890000	0.850000	0.870000	1651.000000
accuracy	0.870000	0.870000	0.870000	0.870000
macro avg	0.870000	0.870000	0.870000	3300.000000
weighted avg	0.870000	0.870000	0.870000	3300.000000

Fig 8.3 SVM Evaluation metrics

	precision	recall	f1-score	support
neg	0.610000	0.860000	0.720000	1649.000000
pos	0.770000	0.460000	0.580000	1651.000000
accuracy	0.660000	0.660000	0.660000	0.660000
macro avg	0.690000	0.660000	0.650000	3300.000000
weighted avg	0.690000	0.660000	0.650000	3300.000000

Fig 8.4 KNN Evaluation metrics

	precision	recall	f1-score	support
neg	0.780000	0.530000	0.630000	1649.000000
pos	0.640000	0.850000	0.730000	1651.000000
accuracy	0.690000	0.690000	0.690000	0.690000
macro avg	0.710000	0.690000	0.680000	3300.000000
weighted avg	0.710000	0.690000	0.680000	3300.000000

Fig 8.5 Decision Tree Evaluation Metrics

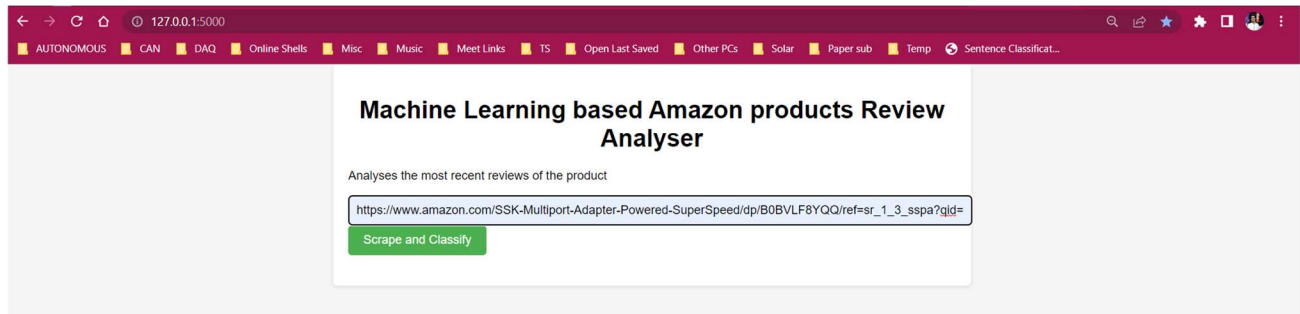


Fig 8.6 User Interface Before pasting the website Link

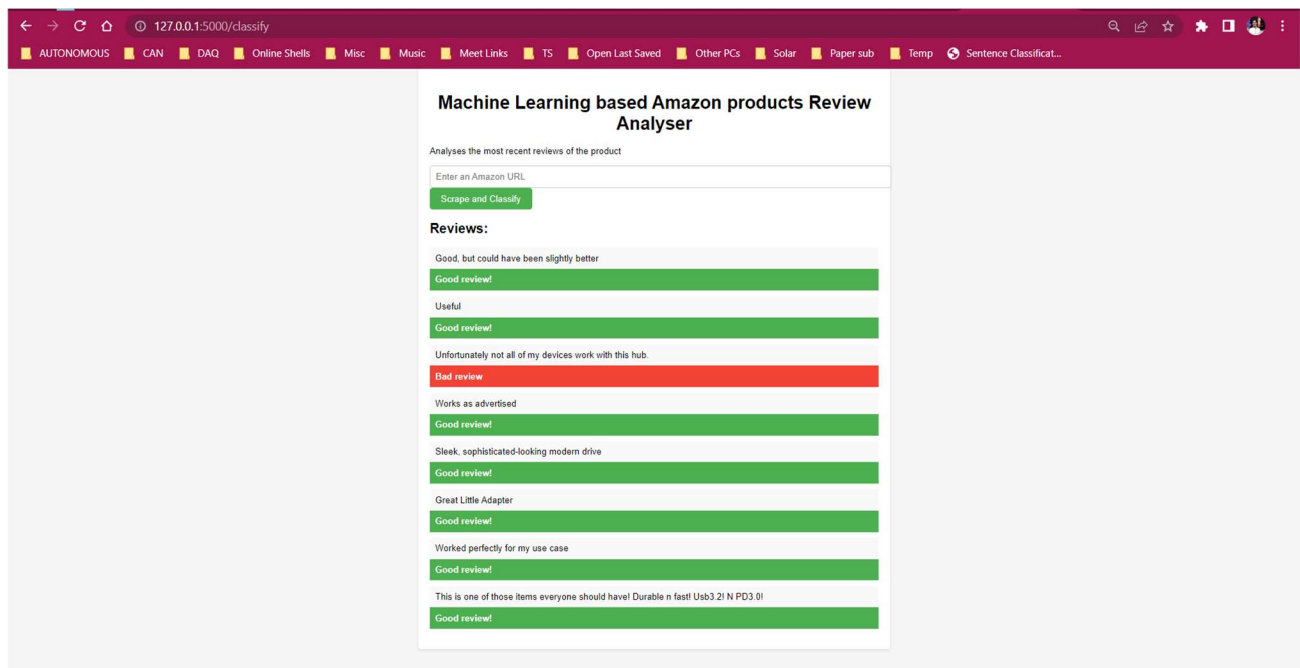


Fig 8.7 User Interface Output after pasting the link

CHAPTER 9

CONCLUSION

9.1 CONCLUSION AND FUTURE SCOPE

In conclusion, this project successfully implemented a machine learning and natural language processing solution for sentiment analysis of Amazon product reviews. By training and evaluating different machine learning models, the Support Vector Machine (SVM) model emerged as the top performer with a high accuracy of 90%. The web application, developed using Flask and other technologies, provided a user-friendly interface for users to input an Amazon product URL, scrape online reviews, and classify them as positive or negative using the trained SVM model.

The project demonstrated the effectiveness of machine learning techniques in analysing sentiment from customer reviews. The accurate classification of reviews can assist users in making informed decisions about purchasing products based on customer feedback. The integration of web scraping and machine learning into a user-friendly web application adds value by simplifying the sentiment analysis process.

Future Scope:

The project opens up several avenues for future enhancements and expansions:

1. **Expansion to Multiple E-commerce Platforms:** Extend the web application to support sentiment analysis on reviews from various e-commerce platforms beyond just Amazon, allowing users to analyse sentiment across multiple sources.
2. **Aspect-Based Sentiment Analysis:** Enhance the sentiment analysis by incorporating aspect-based sentiment analysis, which can provide insights into sentiment regarding specific aspects or features of a product.

3. **Real-Time Analysis:** Implement real-time sentiment analysis, enabling the web application to process and classify reviews as they are posted, providing up-to-date sentiment analysis for products.
4. **Sentiment Visualization:** Develop visualizations to present sentiment analysis results, such as sentiment trends over time or sentiment distribution across different product categories, enhancing the understanding of overall sentiment patterns.
5. **User Feedback and Rating Prediction:** Extend the project to predict user ratings or satisfaction levels based on sentiment analysis, providing a comprehensive view of customer sentiment.

By exploring these future enhancements, the project can be further improved and expanded to provide more advanced and comprehensive sentiment analysis capabilities for various e-commerce platforms.

Applications:

1. **E-commerce Platforms:** The sentiment analysis model developed in this project can be integrated into e-commerce platforms to automatically analyze and classify product reviews. This information can help businesses gain insights into customer satisfaction, identify areas for improvement, and make data-driven decisions for product development and marketing strategies.
2. **Customer Support and Feedback Analysis:** By applying sentiment analysis to customer support interactions and feedback, businesses can quickly identify and address customer concerns, improve customer experience, and enhance their overall reputation.
3. **Brand Monitoring and Reputation Management:** Sentiment analysis can be employed to monitor and analyze online discussions, social media mentions, and reviews about a brand or company. This allows businesses to track public sentiment, manage their online reputation, and proactively address any negative sentiment or customer dissatisfaction.

4. **Market Research and Competitor Analysis:** Sentiment analysis can provide valuable insights into consumer opinions and preferences related to specific products or brands. This information can be utilized for market research, competitor analysis, and identifying emerging trends.
5. **Product Recommendation Systems:** By incorporating sentiment analysis, personalized product recommendation systems can be developed that take into account not only the user's preferences but also the sentiment of reviews associated with similar products.
6. **Social Media Analytics:** Sentiment analysis can be utilized in social media monitoring and analytics to gauge public sentiment towards specific topics, events, or campaigns. This information can be beneficial for marketing strategies, public relations, and understanding customer perceptions.
7. **Online Review Aggregators:** Sentiment analysis can be employed in online review aggregators to automatically summarize and categorize large volumes of reviews, enabling users to quickly assess overall sentiment and make informed decisions.

Appendix A

Problem statement feasibility assessment using, satisfiability analysis and NP Hard, NP-Complete or P type using modern algebra and relevant mathematical models.

The problem statement in this project, which involves sentiment analysis of Amazon product reviews, does not fall into the categories of NP-Hard or NP-Complete. The computational complexity of sentiment analysis tasks, such as data pre-processing, model training, and evaluation, typically falls under the P-type. While specific algorithms like Support Vector Machines (SVM) and hyperparameter optimization can introduce additional computational overhead, they can be managed efficiently using modern algebra and optimization techniques. Satisfiability analysis, which deals with logical constraints, is not directly applicable in this context. Overall, the problem statement demonstrates feasible computational requirements and can be effectively addressed using machine learning approaches within practical limits.

Appendix B

1. International Conference On Computational Intelligence - ICCI 2022



3rd International Conference on Computational Intelligence



Organised by Indian Institute of Information Technology Pune

Technically Sponsored by Soft Computing Research Society

December 29-30, 2022


E-CERTIFICATE OF PARTICIPATION

Om Sarulkar

presented the paper titled “Hybrid Machine Learning Approach for Sentiment Analysis of Amazon Products: A Survey” authored by Om Sarulkar, Rahul Pitale, Shivam Tikhe, Sumit Giri, Rohan More in the 3rd International Conference on Computational Intelligence held during December 29-30, 2022.

ICCI2022\PP\28

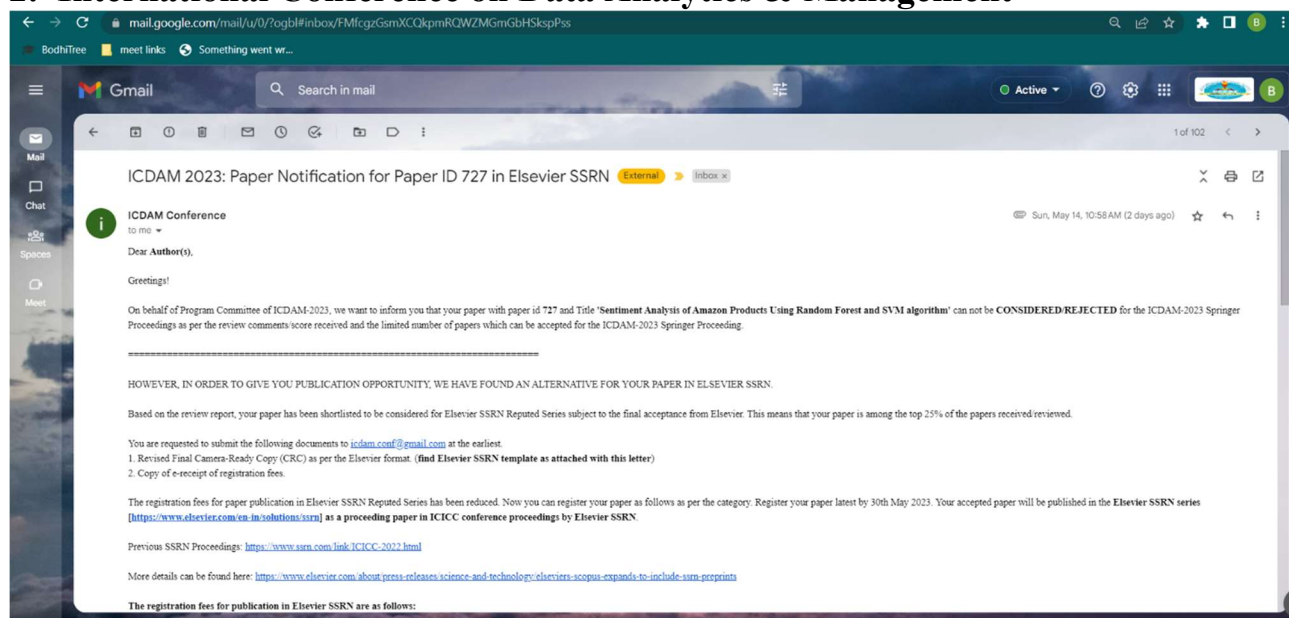

Dr. Mukesh Saraswat
(General Chair)


Dr. Ritu Tiwari
(General Chair)


Dr. Jagdish Chand Bansal
(General Secretary, SCRS)


<https://icci2022.scrs.in>

2. International Conference on Data Analytics & Management



Appendix C

Plagiarism Report of Project report

 **turnitin**

Similarity Report ID: oid:805425034547

● **10% Overall Similarity**

Top sources found in the following databases:

- 3% Internet database
- 8% Publications database
- Crossref database
- Crossref Posted Content database
- 3% Submitted Works database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Monir Yahya Ali Salmony, Arman Rasool Faridi. "Supervised Sentiment ...	6%
	Crossref	
2	scirp.org	<1%
	Internet	
3	mdpi.com	<1%
	Internet	
4	University of Hertfordshire on 2022-09-22	<1%
	Submitted works	
5	University of Kent at Canterbury on 2022-05-24	<1%
	Submitted works	
6	link.springer.com	<1%
	Internet	
7	National College of Ireland on 2019-07-30	<1%
	Submitted works	
8	*Emerging Technologies in Computer Engineering: Cognitive Computin...	<1%
	Crossref	

REFERENCES

- [1] Brownfield, Steven, and Junxiu Zhou. "Sentiment analysis of Amazon product reviews." *Proceedings of the Computational Methods in Systems and Software*. Springer, Cham, 2020.
- [2] S. Maurya and V. Pratap, "Sentiment Analysis on Amazon Product Reviews," 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), 2022, pp. 236-240, doi: 10.1109/COM-IT-CON54601.2022.9850758.
- [3] Parul Sharma S. Dey, S. Wasif, D. S. Tonmoy, S. Sultana, J. Sarkar and M. Dey, "A Comparative Study of Support Vector Machine and Naive Bayes Classifier for Sentiment Analysis on Amazon Product Reviews," 2020 International Conference on Contemporary Computing and Applications (IC3A), 2020, pp. 217-220, doi: 10.1109/IC3A48958.2020.233300.
- [4] AlQahtani, Arwa S. M., Product Sentiment Analysis for Amazon Reviews (2021). *International Journal of Computer Science & Information Technology (IJCSIT)* Vol 13, No 3, June 2021, Available at SSRN: <https://ssrn.com/abstract=3886135>
- [5] Nandal, N., Tanwar, R. & Pruthi, J. Machine learning based aspect level sentiment analysis for Amazon products. *Spat. Inf. Res.* 28, 601–607 (2020). <https://doi.org/10.1007/s41324-020-00320-2>
- [6] Jagdale, R.S., Shirsat, V.S., Deshmukh, S.N. (2019). Sentiment Analysis on Product Reviews Using Machine Learning Techniques. In: Mallick, P., Balas, V., Bhoi, A., Zobaa, A. (eds) *Cognitive Informatics and Soft Computing. Advances in Intelligent Systems and Computing*, vol 768. Springer, Singapore. https://doi.org/10.1007/978-981-13-0617-4_61
- [7] Sindhu, C & Rajkakati, Dewang & Shelukar, Chinmay & Chandra Sekharan, Sindhu. (2020). Context-Based Sentiment Analysis on Amazon Product Customer Feedback Data. 10.1007/978-981-15-5329-5_48.
- [8] Singla, Zeenia, Sukhchandan Randhawa, and Sushma Jain. "Sentiment analysis of customer product reviews using machine learning." 2017 international conference on intelligent computing and control (I2C2). IEEE, 2017
- [9] Jagdale, Rajkumar & Shirsath, Vishal & Deshmukh, Sachin. (2019). Sentiment Analysis on Product Reviews Using Machine Learning Techniques: Proceeding of CISC 2017. 10.1007/978-981-13-0
- [10] Fang, X., Zhan, J. Sentiment analysis using product review data. *Journal of Big Data* 2, 5 (2015). <https://doi.org/10.1186/s40537-015-0015-2>
- [11] S. Kausar, X. Huahu, W. Ahmad, M. Y. Shabir and W. Ahmad, "A Sentiment Polarity Categorization Technique for Online Product Reviews," in *IEEE Access*, vol. 8, pp. 3594-3605, 2020, doi: 10.1109/ACCESS.2019.2963020.
- [12] T. Katić and N. Milićević, "Comparing Sentiment Analysis and Document Representation Methods of Amazon Reviews," 2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY), 2018, pp. 000283-000286, doi: 10.1109/SISY.2018.8524814.

- [13] Sadhasivam, Jayakumar & Babu, Ramesh. (2019). Sentiment Analysis of Amazon Products Using Ensemble Machine Learning Algorithm. *International Journal of Mathematical, Engineering and Management Sciences*. 4. 508-520. 10.33889/IJMEMS.2019.4.2-041.
- [14] F. Iqbal et al., "A Hybrid Framework for Sentiment Analysis Using Genetic Algorithm Based Feature Reduction," in *IEEE Access*, vol. 7, pp. 14637-14652, 2019, doi: 10.1109/ACCESS.2019.2892852.
- [15] Dadhich, A. and Thankachan, B., 2022. Sentiment analysis of amazon product reviews using hybrid rule-based approach. In *Smart Systems: Innovations in Computing* (pp. 173-193). Springer, Singapore.
- [16] Al Amrani, Y., Lazaar, M. and El Kadiri, K.E., 2018. Random forest and support vector machine-based hybrid approach to sentiment analysis. *Procedia Computer Science*, 127, pp.511-520.
- [17] Al Amrani, Yassine, Mohamed Lazaar, and Kamal Eddine El Kadiri. "A Novel Hybrid Classification Approach for Sentiment Analysis of Text Document." *International Journal of Electrical & Computer Engineering* (2088-8708) 8.6 (2018).
- [18] A. Alrehili and K. Albalawi, "Sentiment Analysis of Customer Reviews Using Ensemble Method," 2019 International Conference on Computer and Information Sciences (ICCIS), 2019, pp. 1-6, doi: 10.1109/ICCISci.2019.8716454.
- [19] Alroobaea, Roobaea. "Sentiment Analysis on Amazon Product Reviews using the Recurrent Neural Network (RNN)." *International Journal of Advanced Computer Science and Applications* 13.4 (2022).