

**Savitribai Phule Pune University**

**A PROJECT PHASE I REPORT ON**

## “SENTIMENT ANALYSIS OF PRODUCT REVIEW USING MACHINE LEARNING”

SUBMITTED TOWARDS THE

PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

## BACHELOR OF ENGINEERING IN

**INFORMATION TECHNOLOGY BY**

|  |  |
| --- | --- |
| Ms. Archana Ambhure  Ms. Swamini Chavan  Mr. Mauli Karche  Mr. Tushar Kolhe | Exam Seat No: B1907308501  Exam Seat No: B1907308506  Exam Seat No: B1907308529  Exam Seat No: B1907308560 |

Under The Guidance of

**Prof. Vinita Kute**



### SEM-I

**DEPARTMENT OF INFORMATION TECHNOLOGY JSPM’s BHIVARABAI SAWANT INSTITUTE OF TECHNOLOGY AND RESEARCH, PUNE - 412207**

**Academic Year 2024-25**



**JSPM’s**

**BHIVARABAI SAWANT INSTITUTE OF TECHNOLOGY AND RESEARCH, PUNE - 412207**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

# CERTIFICATE

This is to certify that the Project Entitled

## “AUTOMATION BASED WASTE MANAGEMENT SYSTEM USING IOT TO SUPPORT SWACHH BHARAT ABHIYAN”

Submitted by

|  |  |
| --- | --- |
| Ms. Archana Ambhure  Ms. Swamini Chavan  Mr. Mauli Karche  Mr. Tushar Kolhe | Exam Seat No: B1907308501  Exam Seat No: B1907308506  Exam Seat No: B1907308529  Exam Seat No: B1907308560 |

Is a bonafide work carried out by the students under the supervision of **Prof. Vinita Kute** and it is submitted towards the partial fulfillment of the requirement of the Bachelor of Engineering in Information Technology.

|  |  |
| --- | --- |
| Prof. Vinita Kute | Dr. Vinod Wadne |
| Project Guide | H.O.D |
| Prof. Rekha Kotwal | Dr. T. K. Nagaraj |
| Project Co-coordinator | Principal |
| External Examiner | |

Place : Date :

## ACKNOWLEDGEMENT

*It gives us great pleasure in presenting the preliminary project report on*

***‘Sentiment Analysis on Product Review using Machine Learning’****.*

*We would like to take this opportunity to thank our internal guide* ***Prof. Vinita Kute*** *for giving me all the help and guidance we needed. We are really greatful to her for their kind support. Her valuable suggestions were very helpful.*

*We are also greatful to our project coordinator* ***Prof. Vinita Kute*** *for her valuable time, support, comment, suggestions and persuasion. We are really greatful to* ***Dr. Vinod Wadne****, Head of Information Technology Department, Bhivarabai Sawant Institute of Tech. and Research, Pune for his unconditional support and guidance.*

*We would like to thanks our Principal* ***Dr. T. K. Nagaraj****, for allowing us to pursue our project in this institute.*

*We would also like to thank the institute for providing the required facilities, internet access and important books.*

Ms. Archana Ambhure Ms. Swamini Chavan Mr. Mauli Karche

Mr. Tushar Kolhe

## ABSTRACT

Sentiment analysis of product reviews plays a crucial role in understanding consumer feedback, improving customer experience and making informed business decisions. This paper explores the application of machine learning and deep learning algorithms to effectively classify and analyse the sentiment of product reviews. Traditional machine learning techniques, such as Naïve Bayes, Support Vector Machines (SVM) and Random Forests are employed for sentiment classification based on manually engineered features. Simultaneously, deep learning approaches like Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks are leveraged to automatically learn complex representations from raw text data. The study compares the performance of these methods in terms of accuracy, precision, recall and F1-score. Additionally, pre-trained language models such as BERT are incorporated to enhance contextual understanding. Experimental results demonstrate that deep learning models particularly LSTM and BERT, outperform traditional machine learning techniques in capturing sentiments. This analysis provides valuable insights into the effectiveness of different algorithms in sentiment analysis tasks, paving the way for more advanced applications in natural language processing and customer sentiment evaluation.

***Keywords: -Sentiment Analysis, Machine Learning, Deep Learning, Product Reviews, Natural Language Processing, Text Pre-processing, Feature Extraction, RNN, LSTM, Classification.***

INDEX

|  |  |  |
| --- | --- | --- |
|  | [**ABSTRACT**](#_bookmark0) |  |
| **1** | [**INTRODUCTION**](#_bookmark1) | **1** |
| 1.1 | [PURPOSE](#_bookmark2) | 1 |
| 1.2 | MOTIVATION / BACKGROUND | 2 |
| **2** | [**LITERATURE REVIEW**](#_bookmark3) | **3** |
| **3** | **METHODOLOGY** | **4** |
| **A** | **EXISTING SYSTEM** | 4 |
| 3.1 | ADVANTAGES | 6 |
| 3.2 | DISADVANTAGES | 6 |
| 3.3 | LIMITATIONS | 6 |
| **B** | **PROPOSED SYSTEM** | 7 |
| 3.4 | INTRODUCTION | 7 |
| 3.5 | PROBLEM STATEMENT | 7 |
| 3.6 | OBJECTIVES | 8 |
| 3.7 | ARCHITECTURE | 8 |
| 3.8 | [ADVANTAGES](#_bookmark7) | 10 |
| 3.9 | DISADVANTAGES | 11 |
| 3.10 | APPLICATIONS | 11 |
| **4** | **SYSTEM DESIGN** | **12** |
| 4.1 | DATA FLOW DIAGRAM | 12 |
| 4.2 | USE CASE DIAGRAM | 14 |
| 4.3 | CLASS DIAGRAM | 15 |
| 4.4 | SEQUENCE DIAGRAM | 17 |
| **5** | **PLANNING AND SCHEDULING** | **20** |
| 5.1 | ACTION PLAN | 20 |
| 5.2 | MONTH WISE PLAN | 20 |
| **6** | **REQUIREMENT ANALYSIS** | **21** |
| 6.1 | SOFTWARE REQUIREMENT | 21 |
| 6.2 | HARDWARE REQUIREMENT | 21 |
| 6.3 | FUNCTIONAL REQUIREMENTS | 21 |
| 6.4 | NON-FUNCTIONAL REQUIREMENTS | 22 |
| **7** | **RESULTS AND DISCUSSION** | **24** |
| **8** | **CONCLUSION AND FUTURE SCOPE** | **25** |
| **9** | **REFERNCES** | **26** |

**List of Abbreviations**

|  |
| --- |
| NLP Natural Language Processing |
| ML Machine Learning  CNN Convolutional Neural Network  RNN Recurrent Neural Network  BERT Bidirectional Encoder Reprsenations  form Transformer |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Fig. No.** | **Name of Figure** | **Page No.** |
| 3.7 | Architecture of Proposed system | 8 |
| 4.1 | DFD 0 | 11 |
| 4.1 | DFD 1 | 11 |
| 4.2 | [USECASE DIAGRAM](#_bookmark4) | 13 |
| 4.3 | [CLASS DIAGRAM](#_bookmark5) | 14 |
| 4.4 | [SEQUENCE DIAGRAM](#_bookmark6) | 16 |
| 5.1 | Action Plan | 19 |

# Chapter 1

## INTRODUCTION

### [PURPOSE](file://localhost/C:/Users/Shlok/Downloads/Sample%20Project%20Reprot%20sem%201.docx%23_bookmark3)

In today's digital marketplace, consumer reviews play a pivotal role in shaping purchasing decisions. With an overwhelming volume of reviews available across e- commerce platforms, manually assessing customer feedback becomes impractical. Sentiment analysis offers a powerful solution by leveraging natural language processing (NLP) and machine learning (ML) to automatically evaluate the emotional tone behind textual data.

Sentiment analysis, often referred to as opinion mining, involves determining whether a piece of text expresses a positive, negative. This technique enables businesses to gain insights into customer satisfaction, product strengths, and areas for improvement. For instance, a positive sentiment may indicate customer approval, while a negative sentiment may highlight a product flaw.

Machine learning plays a crucial role in automating this process by training models to recognize sentiment patterns in large datasets. Using labelled review data, a machine learning algorithm can learn to predict sentiment based on word usage, sentence structure and contextual information. Popular ML models for sentiment analysis include logistic regression, decision trees, and advanced methods like deep learning using neural networks. By applying sentiment analysis to product reviews, companies can efficiently monitor consumer opinions, optimize marketing strategies and improve overall customer experience. This project aims to develop a sentiment analysis model that classifies product reviews into positive, negative or neutral categories, providing valuable insights for businesses to make data-driven decisions.

Deep learning has transformed sentiment analysis by enabling models to capture the complexity of human language in text data. Techniques like CNNs, RNNs, LSTMs, and BERT each offer unique advantages for handling language's intricacies. Deep learning's impact on sentiment analysis extends beyond just accuracy; it provides the capacity to understand nuanced expressions, sarcasm, and context-dependent meanings in customer reviews.

### MOTIVATION / BACKGROUND

The motivation behind implementing sentiment analysis of product reviews using machine learning stems from the increasing importance of understanding customer opinions in the digital age. With the vast amount of user-generated content, such as reviews and feedback, it becomes impractical for businesses to manually assess and interpret customer sentiments.

Key motivations for this project include:

* + 1. Customer Insights:

Understanding the emotions and opinions of customers helps companies tailor their products and services to meet customer expectations, improving satisfaction and retention.

* + 1. Market Competitiveness:

In today’s highly competitive market, analyzing reviews can give businesses a competitive edge by quickly identifying strengths and weaknesses of products based on real customer feedback.

* + 1. Automation and Efficiency:

Manually reading thousands of reviews is time- consuming and inefficient. Machine learning automates this process, providing real- time insights which are critical for timely business decisions.

* + 1. Improved Product Development:

By understanding the specific aspects of a product that customers love or dislike companies can refine product development strategies, focusing on improving features that matter most to users.

* + 1. Better Customer Support:

Sentiment analysis enables businesses to address negative feedback more effectively, preventing potential customer churn and fostering a positive brand image.

* + 1. Scalability:

As businesses grow and receive feedback from multiple platforms, sentiment analysis using machine learning allows them to scale their review analysis efficiently, providing insights across a large volume of reviews.

# Chapter 2

**LITERATURE SURVEY**

1. Title: Amazon Product Reviews Sentimental Analysis using Machine Learning Classification Methods:

* Lexicon Method: Uses predefined dictionaries to determine sentiment scores but has limitations in capturing context.
* Supervised Machine Learning Method: Involves training algorithms on labeled datasets to predict sentiment based on features extracted from text.
* Hybrid Method: Combines lexicon-based and machine learning approaches to leverage the strengths of both.

2) Title: Sentiment Analysis Using Machine Learning Classification Methods:

* Bernoulli Classifier:

A type of Naive Bayes classifier that is particularly suited for binary/boolean

features. It assumes that the presence or absence of a feature is what influences the

outcome.

* SVM (Support Vector Classification):

A variant of Support Vector Machines (SVM) that allows for the adjustment of the decision boundary and is effective for classification tasks, especially in high-dimensional spaces.

* Pattern Classifiers:

These classifiers are used to identify patterns within the data and classify the

sentiments based on those patterns.

1. Title: Product Aspect Ranking Using Sentimental Analysis:

* Traditional vs. Deep Learning Models:

Recent studies indicate that deep learning models outperform traditional sentiment analysis methods. Research shows that deep learning architectures, such as Long Short-Term Memory (LSTM) networks, are particularly effective in handling long-term dependencies in text data, leading to improved sentiment classification outcomes.

* Aspect Extraction and Opinion Detection:

Various methods for aspect extraction have been explored. For instance, some

studies utilize semantic syntax trees to identify noun phrases as product features,

while others employ dictionary-based approaches to enhance aspect identification

accuracy.

Probabilistic models have also been introduced for ranking product aspects based on

their frequency and sentiment polarity derived from user reviews.

1. Sentiment Classification Techniques:
   * The Naive Bayes and Support Vector Machine (SVM) algorithms have been widely adopted for sentiment classification. These methods analyze the sentiment of reviews by calculating the polarity of words and classifying the overall sentiment as positive, negative, or neutral.
   * Hybrid approaches combining Naive Bayes with SVM have shown promising results in classifying sentiments more accurately.
2. Ranking Algorithms:
   * Probabilistic aspect ranking algorithms are commonly used to rank product features based on their importance derived from customer feedback. These algorithms consider both the frequency of mentions and the sentiment scores associated with each aspect.
   * Studies have demonstrated the effectiveness of using Term Frequency-Inverse Document Frequency (TF-IDF) for calculating the weight of opinionated terms, which aids in aspect ranking.

4) Title: Temporal And Sentimental Analysis of Customer Reviews:

1. Consumer Sentiment Analysis:
   * The significance of understanding consumer sentiment has been widely documented, with numerous studies highlighting its impact on purchasing decisions. Research shows that favorable online reviews significantly influence sales, underscoring the need for businesses to monitor and analyze customer sentiment over time.
2. Temporal Sentiment Analysis:
   * Studies have increasingly focused on temporal sentiment analysis, which examines how sentiments evolve over time. This approach allows businesses to identify trends and shifts in consumer attitudes, particularly during key events such as product launches or seasonal sales. The temporal dimension is crucial for understanding the dynamics of consumer behavior.
3. Natural Language Processing (NLP) Techniques:
   * The application of NLP techniques in sentiment analysis has gained traction, with various methods being employed to extract and classify sentiments from customer reviews. Techniques such as tokenization, sentiment lexicons, and machine learning algorithms are commonly used to analyze large datasets effectively.
4. Aspect-Based Sentiment Analysis:
   * Aspect-based sentiment analysis (ABSA) focuses on identifying specific features of products that consumers express sentiments about. Research indicates that consumers may have differing opinions on various aspects of a product, making ABSA essential for gaining nuanced insights. This method enables businesses to tailor their offerings based on specific consumer preferences.

# Chapter 3

## METHODOLOGY

### EXISTING SYSTEM

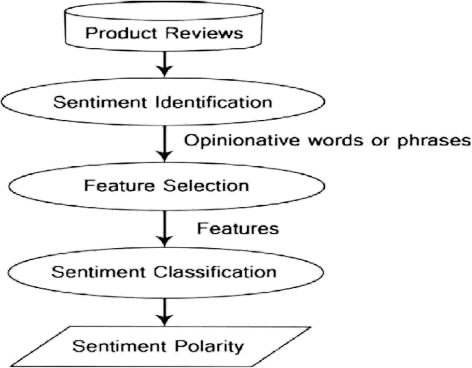


Fig: Existing Methodology of Product Review

#### 1. Data Acquisition :

Dataset collected from Kaggle website, for anyone with an interest in data science and machine learning, it is an online community platform, on Kaggle, users can solve data science challenges by competing with other data scientists, search and publish datasets and interact with other users. The dataset comprises of various features such as reviewer name, review text, overall rating about the card, have removed unnecessary columns and unnamed columns from this dataset.

#### Data Pre processing :

Preparing data is the most important step when building any machine learning model. The dataset is unstructured, therefore it might have words that are repeated and more words which are not at all needed in polarizing the reviews. Preprocessing consists of various techniques for example,

**Tokenization** :

It is defined as the process of extracting individual words from a document, such as a sentence or paragraph. These single terms are referred to as tokens and tokens may consist of single words, full sentences . For performing tokenization , first imported nltk (natural language toolkit) library in google collab notebook.

Import nltk

from nltk.tokenize import word\_tokenize,

For example, the phrase “product was awesome ” was tokenized into ‘product’, ‘was’, ‘awesome’.

**Stop words removal :**

It is defined as the process of removing common and non- meaningful words like “when” , “or” ,etc, from the textual data .Stop words can be removed using various python libraries like nltk, spacy and genism .

import nltk from nltk.corpus

import stopwords from nltk.tokenize

import word\_tokenize

Firstly identified stop words then tokenize them into a list of individual words. Then iterate through the list of tokens and remove any words that are identified as stop words.

**Stemming and Lemmatization :**

Using stemming, affixes are removed from words to obtain their base form. This technique used by search engines to index words. For example, the words walking, walks, walked have their stem as walk . For using stemming , firstly imported nltk library known as port stemmer and word\_tokenize.

import nltk from nltk.stem

import PorterStemmer from nltk.tokenize

import word\_tokenize

Firstly Start with tokenized text, then applied a stemming algorithm to each word, to reduce it to its

base form. The goal is to remove prefixes or suffixes, leaving behind the root form of the word. Lemmatization is similar to stemming but it take into account the context and returns the word to its original, meaningful form, which is known as lemma. For using lemmatization, we have imported nltk library known as WordNetLemmatizer , word\_tokenize

import nltk from nltk.stem

import WordNetLemmatizer from nltk.tokenize

import word\_tokenize

We firstly tokenized texts, then we applied lemmatization algorithm to each word, reducing it to its base form.

#### 3. Feature Engineering:

Machine learning algorithms only understand binary data so, there is a need to convert textual data

into numeric data. Feature engineering is one of the most crucial step in sentimental analysis.

##### ADVANTAGES

* Automated Review Analysis:

Efficient Processing of Large Data Volumes: Machine learning models can process vast numbers of product reviews in a fraction of the time it would take manually. This is especially useful for e-commerce platforms with thousands of reviews per product.

Real-Time Insights: With machine learning, sentiment analysis can be performed in real-time, providing immediate insights from new reviews as they are posted.

* Improved Accuracy Over Traditional Methods:

Deep Learning Capabilities: Machine learning, especially deep learning techniques like neural networks, improves the accuracy of sentiment detection by learning complex patterns in text. It captures subtle emotions like sarcasm or context that may be missed by rule-based systems.

Contextual Understanding: Machine learning models can better understand the context in which a review is written, improving sentiment classification. For instance, the phrase "surprisingly good" would be recognized as positive, despite the word "surprising."

* Improved Customer Experience:

Personalized Recommendations: By understanding the sentiment of individual reviews, machine learning models can help platforms provide better product recommendations based on customer preferences and experiences.

Faster Response to Issues: Machine learning models can flag negative reviews immediately, enabling customer service teams to address complaints or issues quickly and reduce customer churn.

* Cost-Efficient Review Management:

Reduced Need for Manual Labor: By automating the process of reading and interpreting reviews, companies save on labor costs and can scale sentiment analysis without the need for a large team of human analysts.

Continuous Learning and Improvement: Machine learning models continuously improve as they are exposed to more data. As more reviews are processed, the models refine their understanding of sentiments, leading to better results over time.

##### 3.2 DISADVANTAGES

* + - * Inaccuracy in Complex Sentiments:

Difficulty detecting sarcasm, irony or humor: Machine learning models, especially if not well-trained on nuanced language, may struggle to detect sarcasm, irony or humorous reviews, which can lead to incorrect sentiment classifications.

* Dependence on High-Quality Data:

Data preprocessing requirements: Before training machine learning models, a lot of effort is required for data cleaning, such as removing spam, redundant reviews and irrelevant text, to ensure high-quality results.

* + - * Language and Domain Dependency:

Difficulties with multilingual sentiment analysis: While machine learning models can handle multiple languages, training a model to understand various languages accurately requires large, domain-specific datasets for each language. This can be resource-intensive and may still lead to inaccurate results in some languages.

* Need for Continuous Retraining:

Concept drift: Customer sentiment can change over time, as trends and preferences evolve. Machine learning models need regular retraining to adapt to new language usage, emerging slang or changes in customer expectations. Failing to update models regularly may lead to outdated or inaccurate sentiment classification.

##### LIMITATIONS

* Negation Handling:

Negation is a significant challenge in sentiment analysis because a single negating word (like "not") can completely alter the sentiment of a sentence. For instance, "This product is good" is positive, while "This product is not good" is negative. Negation often changes the sentiments of the words that follow it, requiring the model to understand the contextual impact of negating words.

* Emotional Complexity:

Human emotions in text are rarely straightforward, often containing a mixture of sentiments. For example, a review might say, "I love the features, but I’m disappointed with the battery life." This expresses both positive and negative emotions in a single sentence, posing a challenge for simplistic models. Emotionally complex sentences require a sentiment analysis model that can interpret multiple sentiments and weigh them accordingly

* Evaluation Challenges:

Evaluating sentiment analysis models is inherently difficult due to subjective language interpretations, varying intensities of emotions, and dataset biases.Moreover, inconsistencies in labeled data, especially where the sentiment is ambiguous, can affect evaluation accuracy. Effective evaluation requires high-quality labeled data and a balanced dataset that accounts for different tones, languages, and complexities.

* Handling Sarcasm:

Sarcasm presents one of the toughest challenges in sentiment analysis because the literal meaning of words differs from the intended sentiment. For example, "Oh, just what I needed—a phone that dies after an hour!" reads positively on the surface but is actually negative. Detecting sarcasm requires an understanding of tone, context, and often external knowledge. Traditional sentiment analysis approaches struggle with this.

### PROPOSED SYSTEM

##### 3.4 INTRODUCTION

##### In the digital age, product reviews have become an essential resource for consumers making purchasing decisions, as well as for companies aiming to understand customer preferences and improve their offerings. With the sheer volume of reviews generated on e-commerce platforms, social media, and forums, manual analysis of customer sentiment has become impractical. Automated sentiment analysis, the process of determining the emotional tone behind texts, has emerged as a powerful tool in understanding and predicting consumer behavior.

##### Deep learning has proven especially effective in this domain, thanks to its ability to learn complex representations of text data and capture nuances of language, such as sarcasm, context, and polarity. By leveraging techniques like recurrent neural networks (RNNs), convolutional neural networks (CNNs), and transformers, deep learning models can process large volumes of textual data and classify sentiments with remarkable accuracy. This project explores the use of deep learning models to perform sentiment analysis on product reviews.

##### The goal is to develop a robust model that can accurately classify reviews into positive, negative, or neutral categories, thus providing valuable insights for businesses. By understanding the underlying emotions in customer feedback, companies can enhance customer satisfaction, improve products, and make data-driven decisions to maintain a competitive edge in the market.

##### 3.5 PROBLEM STATEMENT

##### To classify the positive, negative and neutral of a given text at the document, sentence and extract

##### key insights that can help businesses understand customer satisfaction, identify common issues, and

##### improve products.

##### 3.6 OBJECTIVES

##### To understand how customers perceive and experience their products and brands.

* To identify and extract the sentiment of specific aspects or components of a product or service

##### 

3.7 ARCHITECTURE

Methodology (Architecture):

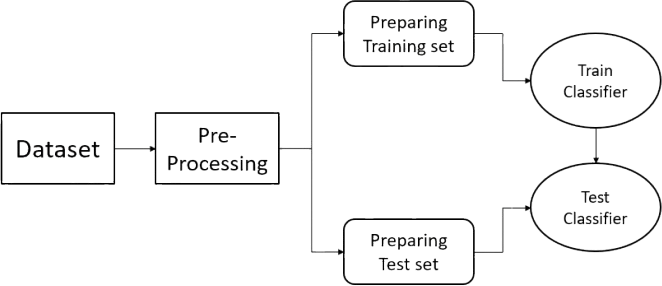


Fig: Block diagram of Sentimental analysis

#### Data Collection

* + Sources: Collect product reviews from various platforms like e-commerce sites (Amazon, eBay), social media (Twitter, Facebook) or specialized review sites.

#### Preprocessing

* + Data Cleaning: Remove noise like HTML tags, special characters and irrelevant data (e.g., advertisements).
  + Tokenization: Split reviews into individual words or tokens.
  + Stopword Removal: Remove common words (e.g., "the," "and") that don’t contribute to the sentiment.
  + Stemming/Lemmatization: Reduce words to their base forms (e.g., "running" → "run").
  + Handling Negations: Special handling of negations (e.g., “not good” → "negative sentiment") for accurate sentiment capture.

#### Feature Extraction

* + Bag of Words (BoW): Represent reviews as a frequency distribution of words.
  + TF-IDF (Term Frequency-Inverse Document Frequency): Adjust word frequency by how often they appear across documents.

#### Sentiment Classification Models

### 1) Machine Learning Models

I. Naïve Bayes

* Naïve Bayes is mostly used as a baseline for text classification tasks. It calculates the probability of each sentiment class which is given the words in a review.It is Probabilistic classifier.
* Advantages: Simple, fast, and performs surprisingly well on small datasets.
* Limitations: It does not handle complex language structures.

II. Logistic Regression

* Logistic regression is also popular baseline model for sentiment analysis. It uses weights for each word feature to predict the sentiment.It is Linear classifier.
* Advantages: Easy to interpret and implement, works well with TF-IDF or BoW.
* Limitations: It struggles with long-term dependencies in text.

III. Support Vector Machines (SVM)

* SVM is used for text classification tasks and can perform well with high-dimensional data such as TF-IDF.It is Linear classifier.
* Advantages: It is good for medium-sized datasets which gives good performance.
* Limitations: It can be slow with large datasets.

2) Deep Learning Models:

I. Convolutional Neural Networks (CNN)

* Originally used for image processing but also effective in text analysis.
* Uses filters to detect local patterns in text, such as specific phrases or n-grams.
* Suitable for capturing short-term dependencies in text.
* Fast to train due to parallel processing.

#### II. Recurrent Neural Networks (RNN)

#### Designed for sequential data processing, making it suitable for text analysis.

#### Maintains information from previous words (context) while processing the current word.

#### Struggles with long-term dependencies due to the vanishing gradient problem.

#### III. Long Short-Term Memory (LSTM)

* A variant of RNN that addresses the vanishing gradient problem.
* Capable of learning long-term dependencies in text sequences.
* Uses memory cells to remember important information over long sequences.
* Widely used in sentiment analysis for handling longer text.

#### 3) Bidirectional Encoder Representations from Transformers (BERT)

* A transformer-based model that understands context from both left-to-right and right-to-left.
* Pre-trained on large datasets and can be fine-tuned for specific tasks (transfer learning).
* Achieves state-of-the-art results in various natural language processing tasks, including sentiment analysis.
* Effective at capturing complex language patterns and nuanced sentiments.

#### Model Training & Testing

* Train-Test Split: Split the dataset into training and testing sets (e.g., 80% train, 20% test).
* Cross-Validation: Implement cross-validation to evaluate model robustness.
* Metrics: Use metrics like Accuracy, Precision, Recall, F1-score.

#### Sentiment Scoring

* Positive/Negative Labels: Classify reviews into these categories.
* Polarity Scores: Provide scores on a range (e.g., 0,1), where 0 represents negative and 1 is positive.

##### 3.8 [ADVANTAGES](#_bookmark7)

* + - * Simplicity and clarity:

Straightforward Interpretation: Binary classification simplifies sentiment analysis by categorizing

reviews into clear, easy-to-understand labels (positive or negative). This provides immediate insights

without the complexity of nuanced gradations (e.g., neutral, slightly positive).

Focus on Extremes: It captures the most polarized opinions, which are often the most important for

decision-making (e.g., highly satisfied or deeply dissatisfied customers).

* + - * Faster Computation:

Lower Computational Costs: Since binary classification only requires distinguishing between two

classes, the models are typically faster to train and deploy compared to multi-class models. This is

especially advantageous when dealing with large datasets or when real-time analysis is required.

Simpler Model Architecture: The models for binary classification can be less complex and computationally intensive, requiring fewer resources.

* + - * Real time Analysis:

Quick Results for Decision-Making: With simpler models, binary classification enables real-time

sentiment analysis, helping businesses respond quickly to shifting customer sentiments (e.g., during

product launches or marketing campaigns).

* Improved customer engagement:

Targeted Marketing Campaigns: Businesses can identify highly satisfied customers and use positive

reviews for testimonials, case studies or social proof. Negative reviews, on the other hand, can be addressed through personalized customer service interactions.

* Enhancing User Experience: By quickly identifying negative feedback businesses can improve the

user experience, resulting in better customer retention and loyalty.

* 1. APPLICATIONS

#### Customer Feedback Analysis

* + Product Improvement: Companies can analyze positive and negative feedback to understand customer sentiment toward specific product features. This helps prioritize product improvements and innovation based on what customers like or dislike.
  + Detecting Common Complaints: Negative reviews can highlight recurring problems with a product (e.g., defects, poor quality), helping businesses address these issues promptly.

#### Reputation Management

* + Brand Monitoring: Companies can use binary sentiment analysis to monitor the overall sentiment toward their brand. A surge in negative reviews may indicate a brand reputation issue that needs to be addressed through improved customer service or crisis management.
  + Competitive Benchmarking: Businesses can compare the sentiment of their product reviews with those of competitors to understand where they stand in the market and which areas need improvement to outperform the competition.

#### Automated Customer Support

* Ticket Prioritization: By flagging negative reviews or comments, businesses can automatically prioritize customer support tickets for unhappy customers, ensuring that their issues are addressed promptly.
* Proactive Customer Service: Companies can proactively reach out to customers who leave negative reviews to resolve issues, enhance the customer experience, and prevent churn.

Chapter 4

## SYSTEM DESIGN

##### DATA FLOW DIAGRAM

* DFD 0

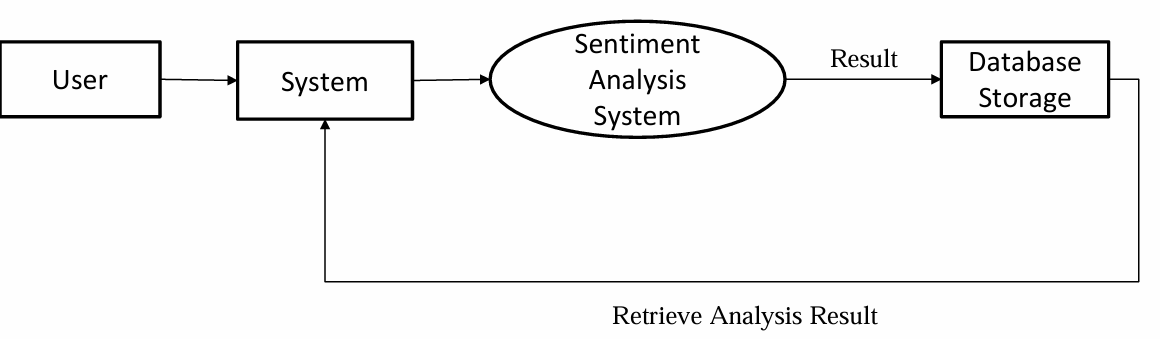


Figure 4.1: DFD 0

##### DFD 1

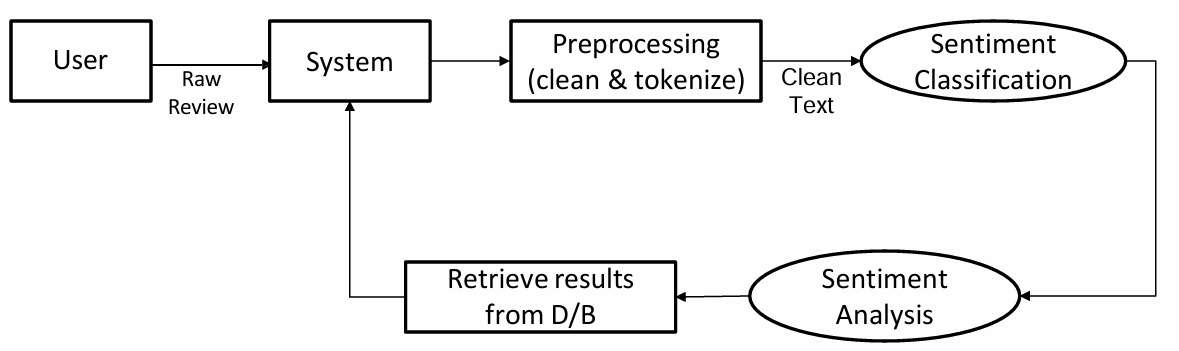


Figure 4.2: DFD 1

What Is DFD

A Data Flow Diagram (DFD) is a visual representation of how data flows within a system. It illustrates the inputs, processes, storage points, and outputs of a system in a simplified, diagrammatic form. DFDs are commonly used in system analysis and design to understand and document how a system operates, the

data it processes, and the way information flows between components. Key Components of a DFD

1. External Entities: Represent sources or destinations of data that interact with the system but are outside its boundaries, such as users, other systems, or external organizations. Typically depicted as rectangles.
2. Processes: Represent actions or transformations applied to data, converting inputs into outputs. Processes are often drawn as circles or ovals.
3. Data Stores: Represent places where data is held within the system, such as databases or files. Data stores are typically shown as open-ended rectangles.
4. Data Flows: Represent the movement of data between entities, processes, and data stores. They are usually depicted as arrows labeled with the type of data being moved.

Levels of DFDs

DFDs are often broken down into levels to represent the system's complexity in stages:

* + Level 0 DFD (Context Diagram): Shows the entire system as a single process with external entities and the main data flows.
  + Level 1 DFD: Breaks down the main process into sub-processes, showing more detail about how data flows within the system.

Benefits of Using DFDs

* + Clarity: Provides a clear, easy-to-understand visualization of system processes and data flows.
  + Documentation: Serves as a helpful document for system development, user training, and system troubleshooting.
  + Analysis and Design: Assists in identifying data handling inefficiencies or redundancies, making it useful for system improvement and optimization.

##### 4.2 USE CASE DIAGRAM

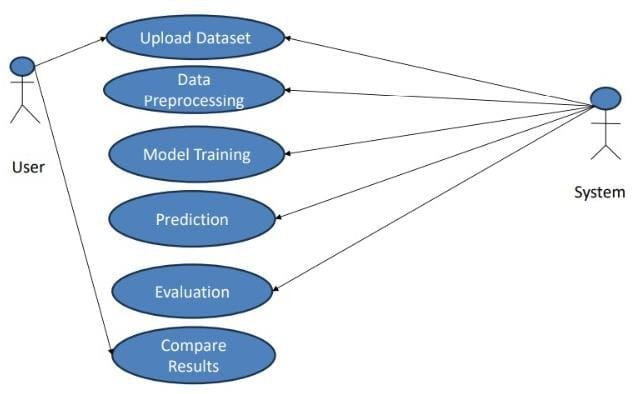


Figure 4.3: USECASE DIAGRAM

A Use Case Diagram is a type of behavioral diagram in the Unified Modeling Language (UML) that visually represents the functional requirements of a system. It shows the interactions between users (or "actors") and the system itself, specifically illustrating what the system does in response to user actions. Use case diagrams are commonly used in the initial stages of system design to capture system requirements and define the scope.

Key Elements of a Use Case Diagram

1. Actors: Represent users or external systems that interact with the system. Actors can be humans, other systems, or hardware devices, and are typically drawn as stick figures.
2. Use Cases: Represent the actions or functions that the system performs in response to an actor's input, capturing specific system functionalities (e.g., "Login," "Submit Order," "Generate Report"). Use cases are drawn as ovals labeled with the action.
3. System Boundary: Represents the limits of the system being modeled, enclosing all use cases within a box to separate internal processes from external actors.
4. Relationships:
   1. Association: A line that connects actors to the use cases they interact with.
   2. Include: Represents a use case that is always included within another use case, indicating a mandatory interaction or subprocess

(e.g., "Verify Payment" included in "Process Order").

* 1. Extend: Represents an optional or conditional relationship where a use case is extended by another if specific conditions are met (e.g., "Add Gift Wrapping" as an extension to "Process Order").
  2. Generalization: Represents inheritance relationships between use cases or actors, where one actor or use case is a specialized version of another.

Purpose of a Use Case Diagram

Use case diagrams are used to:

* Define Functional Requirements: Identify what the system should do in response to user actions, outlining the main features.
* Clarify User Roles: Show who will interact with the system and what actions they can perform.
* Establish System Boundaries: Define which parts of the functionality belong to the system and which are external.
* Provide a High-Level Overview: Give a quick, high-level visualization of the system's functionality and user interactions.

Benefits of Use Case Diagrams

* User-Focused: Centers on what users need from the system, making it easier to identify key features.
* Clear Communication: Provides a straightforward way for developers and stakeholders to discuss system requirements.
* Foundation for Detailed Design: Lays out the groundwork for more detailed system design and testing.

##### CLASS DIAGRAM

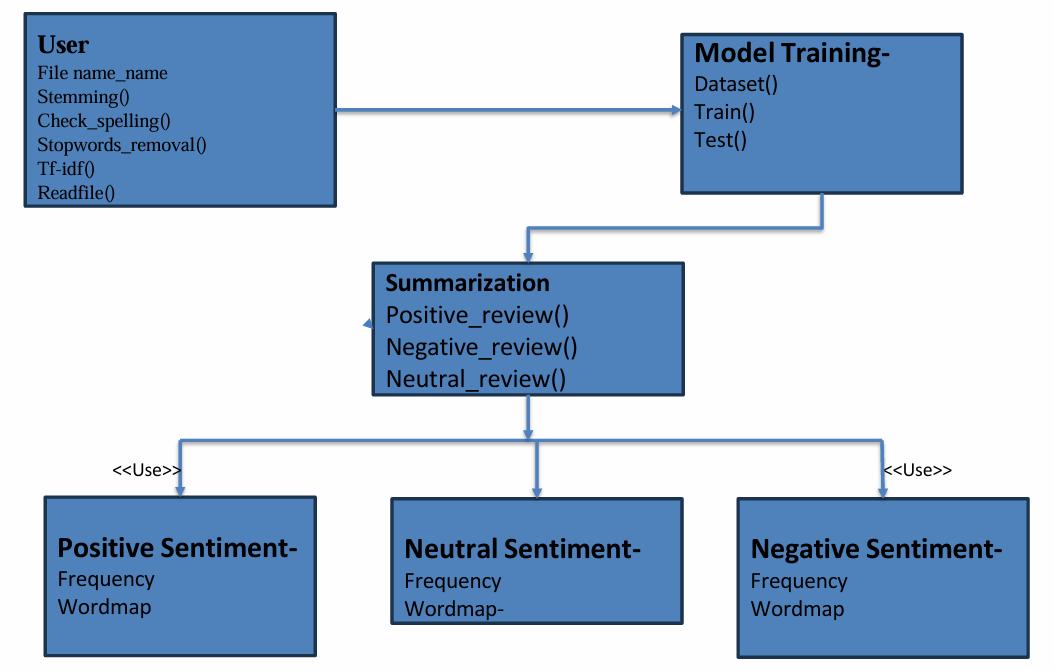


Figure 4.4: CLASS DIAGRAM

A Class Diagram is a type of structural diagram in the Unified Modeling Language (UML) that represents the static structure of a system. It shows the system's classes, their attributes, methods (functions), and the relationships between them. Class diagrams are widely used in object-oriented design to illustrate the system’s organization and serve as a blueprint for the code structure.

Key Elements of a Class Diagram

1. Classes: Represent the main building blocks of an object-oriented system. Each class is depicted as a rectangle divided into three parts:
   * Class Name: The name of the class, displayed in the top section.
   * Attributes: Characteristics or properties of the class, listed in the middle section.
   * Methods (Operations): Actions or functions that the class can perform, listed in the bottom section.
2. Relationships:
   * Association: Represents a link between two classes that shows how they are related. Associations can have roles, multiplicities, and direction.
   * Aggregation: A "whole-part" relationship where one class is made up of multiple instances of another class (e.g., a Library has Books). Depicted with an empty diamond.
   * Composition: A stronger form of aggregation where the part cannot exist independently of the whole (e.g., a House has Rooms). Depicted with a filled diamond.
   * Inheritance (Generalization): Represents an "is-a" relationship where one class inherits from another (e.g., Dog inherits from Animal). Depicted with an open arrow pointing from the subclass to the superclass.
   * Dependency: Represents a "uses-a" relationship where one class depends on another to function (e.g., a Printer depends on Ink). Depicted with a dashed arrow.
3. Multiplicity: Indicates how many instances of a class can be associated with another class (e.g., "1," "0.1," "*," "1..*").

Purpose of a Class Diagram

Class diagrams are used to:

* Define System Structure: Model the data structure and organization of the system's components.
* Represent Relationships: Show how different classes interact and are related to each other.
* Guide Implementation: Serve as a reference for developers when coding, helping ensure that classes and their relationships align with design.
* Facilitate Maintenance: Provide a visual representation that can be used for documentation, understanding, and maintaining code.

Benefits of Class Diagrams

* Clarity in Structure: Provides a clear overview of the system’s architecture.
* Code Blueprint: Acts as a blueprint for developers during system design and implementation.
* Better Communication: Helps developers and stakeholders understand system organization and design choices**.**

##### 4.3 SEQUENCE DIAGRAM

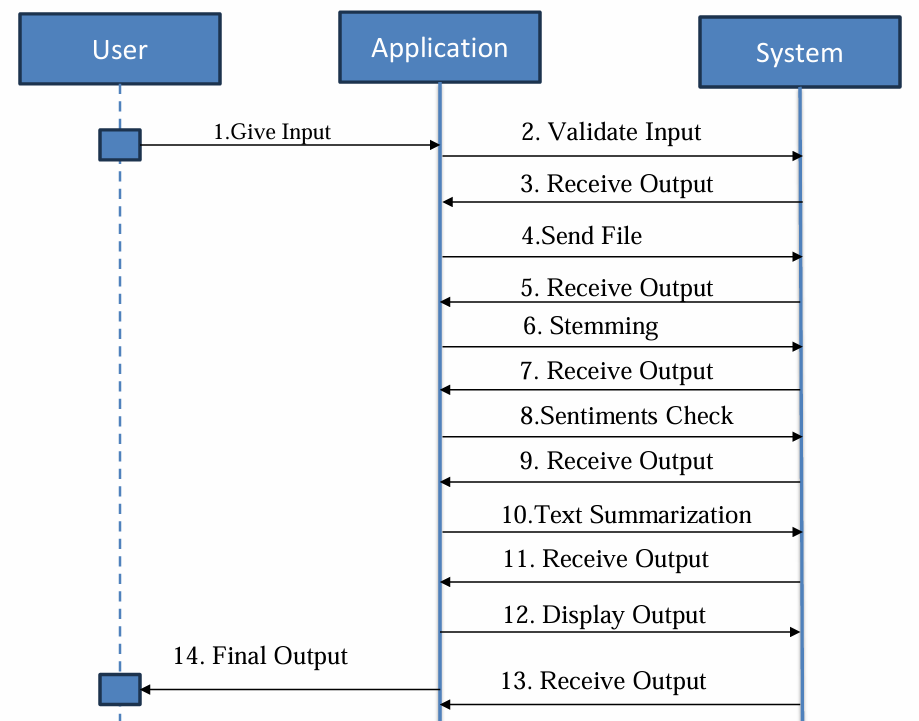


Figure 4.5: SEQUENCE DIAGRAM

A Sequence Diagram is a type of interaction diagram in the Unified Modeling Language (UML) that models the sequence of messages exchanged between objects or components in a system over time. It is used to represent how objects or participants interact in a particular scenario or use case, focusing on the order of interactions (i.e., the flow of messages between objects).

Key Elements of a Sequence Diagram

1. Actors/Objects: Represent the participants in the interaction, typically depicted as rectangles or lifelines at the top of the diagram. An actor is a user or external system that interacts with the system, while objects represent components or instances within the system.
2. Lifeline: Represented as dashed vertical lines, lifelines show the existence of an object during the interaction and the object's timeline over the course of the sequence.
3. Messages: Horizontal arrows between lifelines represent the communication or messages passed between objects. The message may indicate a method call, data transfer, or event. Arrows are usually labeled with the name of the operation or message.
   * Synchronous Message: Represented by a solid line with a filled arrowhead, indicating that the sender is waiting for a response from the receiver.
   * Asynchronous Message: Represented by a line with an open arrowhead, indicating that the sender does not wait for a response before proceeding.
   * Return Message: Represented by a dashed line with an open arrowhead, showing the return of a result or value from a method call.
4. Activation (Focus of Control): Represented by thin rectangles on the lifelines, showing the period during which an object is performing an action or method call.
5. Conditions and Loops: Sequence diagrams can include conditions (if-else branches) or loops to represent logic in the sequence of interactions (e.g., repeating messages or conditional branches).

Purpose of a Sequence Diagram Sequence diagrams are used to:

* Model Interaction: Represent how objects interact with each other to complete a particular task or use case.
* Clarify Message Flow: Show the exact sequence and timing of messages, helping to understand the flow of control and data.
* Document System Behavior: Provide a visual representation of system behavior for stakeholders, helping to communicate and document the system's functionality.
* Analyze Timing and Dependencies: Help identify the dependencies between objects and detect potential bottlenecks or performance issues.
* Clear Understanding of Process Flow: Sequence diagrams provide a step-by-step view of how objects interact in a particular scenario, making it easier to understand system behavior.
* Improved Communication: They serve as a valuable tool for communication between developers, designers, and stakeholders by visually mapping out the interactions.
* Easier Debugging: Helps identify errors or inefficiencies in communication between

objects and ensures the system is working as expected.

* Supports System Design: Sequence diagrams assist in designing the interaction between components and defining the system's behavior at a more granular level.

# Chapter 5

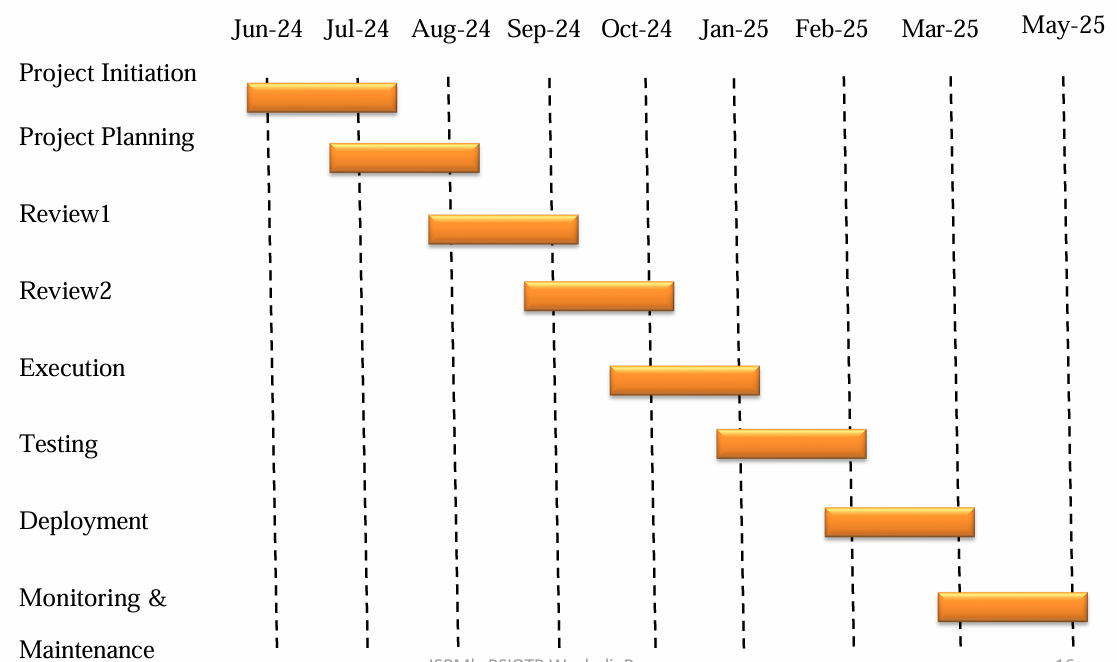
## PLANNING AND SCHEDULING

##### ACTION PLAN

|  |  |
| --- | --- |
| Action plan outline | Goal |
| Requirement gather | Dataset from Kaggle, IEEE Research Papers, Jupyter Notebook |
| Designing | DFD Diagrams, UML Diagrams |
| Implementation | Coding of project |
| Testing | Testing of project |
| Deployment | Deployment of project |
| Maintenance | Maintaing the project |

Table 5.1: Action Plan

* 1. MONTH WISE PLAN



# Chapter 6

## REQUIREMENT ANALYSIS

##### SOFTWARE REQUIREMENT

1. Core i5/i7 processor
2. At least 8GB RAM
3. Anaconda Navigator
4. Jupyter Notebook
5. Kaggle

##### HARDWARE REQUIREMENT

1. Core i5/i7 processor
2. At least 8GB RAM Device

##### FUNCTIONAL REQUIREMENTS

##### Functional requirements for a sentiment analysis system specify the features and

##### capabilities needed to fulfill its purpose effectively. Here are key functional

##### requirements:

1. Text Processing:

This foundational step involves preparing raw text for analysis by cleaning, tokenizing,

and normalizing it. Techniques such as removing stop words, stemming, and

lemmatization reduce noise and ensure that the sentiment-bearing words are accessible

for the model, thus improving accuracy and efficiency.

1. Sentiment Detection:

The core task of identifying sentiment polarity (positive, negative, neutral) or emotions

(e.g., joy, anger) in text. While simple models handle basic polarity, advanced deep

learning models like LSTMs and BERT capture nuanced expressions, enabling more

accurate sentiment detection in complex or ambiguous sentences.

1. Handling of Mixed Sentiments:

Mixed sentiment handling identifies when a text expresses multiple sentiments, such as

positive and negative feelings toward different aspects of a product. Advanced methods

like aspect-based analysis or attention mechanisms allow models to capture

and separately categorize each sentiment within a single text.

1. Multi-language Support:

Effective sentiment analysis often requires understanding multiple languages. Multi-

language support can be achieved through machine translation or multilingual models

like BERT, enabling sentiment detection in diverse linguistic contexts and improving

accessibility for global applications.

1. Sentiment Output and Categorization:

This step involves generating interpretable output from the model. Basic outputs include

binary or ternary classification (positive, neutral, negative), while more sophisticated

models may provide intensity scores or categorize nuanced emotions, helping in detailed

sentiment analysis.

1. Aspect-based Sentiment Analysis:

ABSA identifies specific product or service aspects (e.g., “battery life” or “camera”) and

analyzes the sentiment toward each. This approach provides granular insights into which

features customers like or dislike, aiding in targeted improvements and customer

satisfaction analysis.

1. Scalability and Performance:

Scalability ensures that sentiment analysis systems can handle large volumes of data,

often using efficient deep learning models or optimized computational methods. High-

performance systems are essential for real-time analysis, particularly for applications

requiring immediate insights, such as customer feedback monitoring on social media.

##### NON-FUNCTIONAL REQUIREMENTS

##### Non-functional requirements for a sentiment analysis system using machine learning and deep learning focus on the system's quality attributes, ensuring efficient and reliable performance. Key requirements include:

1. Performance:

Performance measures the system’s ability to process and analyze data quickly and

efficiently. In sentiment analysis, this involves fast data processing, minimal latency,

and optimized computational resources. High-performing systems enable real-time

analysis, which is critical for applications like social media monitoring or customer

service interactions.

1. Scalability:

Scalability refers to the system’s capability to handle increasing amounts of data or user

demand. Scalable sentiment analysis systems can expand to accommodate larger

datasets, such as growing social media posts or user-generated reviews, ensuring

consistent performance under heavy loads. Scalable infrastructure often leverages cloud

resources, distributed computing, or parallel processing.

1. Accuracy and Precision:

Accuracy and precision are essential metrics for sentiment analysis, reflecting the

model's ability to correctly identify sentiments. High accuracy means that the model

captures the correct sentiment overall, while high precision ensures it consistently

categorizes sentiments with minimal false positives. Both metrics are crucial for

maintaining trustworthy analysis results in business or research applications.

1. Reliability and Availability:

Reliability ensures that the sentiment analysis system consistently produces accurate

results, even under varying conditions. Availability refers to the system’s readiness and

uptime, ensuring it’s accessible to users whenever needed. High reliability and

availability are especially critical in real-time applications, where downtime can disrupt

workflows or delay insights.

1. Security and Data Privacy:

Security and data privacy are critical in protecting user data and maintaining

compliance with regulations (e.g., GDPR). Sentiment analysis systems often handle

sensitive customer data, making encryption, secure data handling practices, and access

controls essential to prevent unauthorized access and safeguard user information.

## RESULTS AND DISCUSSION

Analyses are done on unstructured data collected from various resources like Kaggle dataset or e-commerce websites. The require for analysis are as follows:

1. Precision:

Precision measures the accuracy of positive predictions. It is the ratio of correctly predicted positive instances to the total instances predicted as positive. Precision is useful for evaluating how many of the positive predictions made by the model are actually correct.

𝑃𝑟𝑒𝑐𝑖𝑠𝑖𝑜𝑛 =

1. Recall:

Recall (also known as sensitivity or true positive rate) measures the ability of the model to identify all relevant positive instances. It is the ratio of correctly predicted positive instances to the total actual positive instances. Recall is crucial in scenarios where capturing all relevant instances is important.

𝑅𝑒𝑐𝑎𝑙𝑙 =

1. **F1-Score:**

The F1-score is the harmonic mean of precision and recall, providing a single metric that balances both. It is especially useful when there is an uneven class distribution or when precision and recall are both critical. A high F1-score indicates a balance between precision and recall, making it a preferred metric in scenarios where both false positives and false negatives need to be minimized.

𝐹 − 𝑀𝑒𝑎𝑠𝑢𝑟𝑒 =

Where,

* TP(True-positive): a true positive means that the model identified an instance as belonging to the positive class and it actually does belong to that class.,
* TN(True-negative): A true negative occurs when the model correctly identifies an instance as not belonging to the positive class
* FP (False-positive): A false positive occurs when the model labels an instance as belonging to the positive class, but it actually belongs to the negative class
* FN (False-negative): A false negative occurs when the model labels an instance as belonging to the negative class, but it actually belongs to the positive class

## CONCLUSION AND FUTURE SCOPE

User reviews are very important and they influence the purchase decisions. Sentiment analysis provides the users emotion towards the product and their services. Sentimental analysis can be implemented using various techniques and the results range accordingly to the conditions and the factors that influence them. In this paper, we have built and tested a model using logic on datasets of product reviews to find the sentiments of the reviews. The system performs well with the given dataset and with applied conditions

Deep learning significantly improves the accuracy of sentiment analysis in product reviews through several key ways such as automatic feature extraction, understanding context, handling large vocabulary and complex language. By using Machine Learning algorithms system can efficiently analyzes large volumes of data and provide valuable insights into customer opinions and attitude towards product.

The future scope of sentiment analysis in product reviews includes optimizing models for better accuracy, expanding to multilingual support for global reach, and incorporating aspect-based analysis for insights into specific product attributes. Real-time sentiment tracking and integration with recommendation systems could enhance customer experience, while advances in explainable AI will improve transparency and trust. Together, these advancements can make sentiment analysis a more valuable tool for business intelligence and customer insight.

# Chapter 9

## REFERNCES

1. P. Rethina Sabapathi, Dr. K.P.Kaliyamurthie . “Analysis of Customer Review and Predicting Future Release of the Product using machine learning concepts”. 2023 International Conference on Communication, Computing and Internet of Things.
2. Yallanki Vikas, K. Nagendra chary. “Sentiment Analysis Using Machine Learning”. 2022 International Journal of Innovative Research in Technology.
3. Shrey arora, Siddharth Mahapatra, Dr. Anil Jadav, Manojeet Barla, Nishant Mallick.” Temporal and Sentimental Analysis of Customer Reviews”.2024 14th International Conference of Data Science.
4. Yuvashree E, Preethika S, Nirupama A, Cloudin S.” Product Aspect Ranking Using Sentimental Analysis”. 2021 International Conference on System, Computation, Automation and Networking.
5. Gitanshu Chauhan, Akash Sharma, Nripendra Dwivedi. “Amazon Product Reviews Sentimental Analysis using Machine Learning “.2024 IEEE International Conference on Computing, Power and Communication Technologies.
6. Hanan Alasmari.” Sentimental Visualization: Semantic Analysis of Online Product Reviews Using Python and Tableau”. 020 IEEE International Conference on Big Data (Big Data) | 978-1-7281-6251-5/20/$31.00 ©2020 IEEE |.
7. Mukherjee, Subhabrata, and Pushpak Bhattacharyya. "Feature specific sentiment analysis for product reviews." International Conference on Intelligent Text Processing and Computational Linguistics. Springer, Berlin, Heidelberg, 2012.
8. Somprasertsri, Gamgarn, and Pattarachai Lalitrojwong. "Extracting product features and opinions from product reviews using dependency analysis." 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery. Vol. 5. IEEE, 2010

# Appendices – A

Plagiarism report :

