

**PROJECT REPORT**  
**ON**  
**SENTIMENT ANALYSIS USING MACHINE LEARNING**

Submitted in partial fulfillment of the requirement for the award of degree in

**MASTER OF COMPUTER APPLICATIONS**

Of the

**APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY**

Submitted by

**SREEJITH K.M**

**(NCE21MCA-2048)**

Under the guidance of

**Ms. SUMI M, MCA**

**ASSISTANT PROFESSOR**



**DEPARTMENT OF MCA**

**NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE,**

**(NAAC Re-Accredited with “A” grade)**

**PAMPADY, THIRUVILWAMALA, THRISSUR-680567**

**MAY 2023**

**PROJECT REPORT**  
**ON**  
**SENTIMENT ANALYSIS USING MACHINE LEARNING**  
Submitted in partial fulfillment of the requirement for the award of degree in  
**MASTER OF COMPUTER APPLICATIONS**  
Of the  
**APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY**



Submitted by  
**SREEJITH K.M**  
**(NCE21MCA-2048)**  
**Semester 4 MCA (2021-23)**

Under the guidance of  
**Ms. SUMI M, MCA**  
**ASSISTANT PROFESSOR**



**DEPARTMENT OF MCA**  
**NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE,**  
**MAY 2023**



## **NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE**

### **DEPARTMENT OF MCA**

#### **COLLEGE VISION**

To mold true citizens who are millennium leaders and catalysts of change through excellence in education.

#### **COLLEGE MISSION**

NCERC is committed to transform itself into a center of excellence in Learning and Research in Engineering and Frontier Technology and to impart quality education to mould technically competent citizens with moral integrity, social commitment and ethical values. We intend to facilitate our students to assimilate the latest technological know-how and to imbibe discipline, culture and spiritually, and to mould them in to technological giants, dedicated research scientists and intellectual leaders of the country who can spread the beams of light and happiness among the poor and the underprivileged.

#### **DEPARTMENT VISION**

To create a school of distinction for the PG students, prepare them to be industry-ready, and achieve Academic excellence by continuous endorsement of the Faculty team in terms of Academics, Applications & Research.

#### **DEPARTMENT MISSION**

The Department of Computer Applications strives to provide quality and competency-based education and fine-tune the younger generation through Curricular, Co-Curricular and Extra- curricular activities so as to encounter the Professional and Personnel challenges ahead with Pragmatic skills & courage , thereby 'Creating the True Citizens'.

## **DECLARATION**

I hereby declare that the project report entitled “**SENTIMENT ANALYSIS USING MACHINE LEARNING**” Submitted to the **DEPARTMENT OF MCA at Nehru College of Engineering and Research Centre** in partial fulfilment of the requirement for the award of degree in **MASTER OF COMPUTER APPLICATIONS** from **APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY**, is a record of original work done by me under the guidance of **Ms. SUMI M**, Assistant Professor of the Department of MCA, during my Fourth Semester MCA course period 2023.

**Place:Pampady**

**SREEJITH K.M**

**Date:**

**NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE, PAMPADY**



**CERTIFICATE**

This is to certify that, the project work entitled **“SENTIMENT ANALYSIS USING MACHINE LEARNING”** has been presented by **SREEJITH K.M, NCE21MCA-2048** of Fourth Semester MCA in Partial Fulfilment of the requirement for the award degree **MASTER OF COMPUTER APPLICATIONS, APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY.**

We also certify that the work done is original.

**Project Guide**

**Head of the Department**

**Principal**

**External Examiner**

## ACKNOWLEDGEMENT

First and most, I thank the **God Almighty** for showing me the path to the completion of seminar work. I thank **Prof.Dr.Karibasappa Kwadiki**, the principal **NCERC**, for providing good atmosphere for project completion and presentation. I thank **Dr.Sudheer S Marar**, Head of MCA Department ,NCERC for his valuable suggestions and support throughout the project work .I wish to acknowledge my deep sense of gratitude and heartfelt thanks to My project Guide **Ms.Sumi M**,Assistant professor ,MCA Department for his valuable suggestions, precious time, kind hearted motivations throughout the project work period. I express my immense gratitude to all my friends, without whom I would have never been able to do my project well.

## **ABSTRACT**

Sentiment analysis, an important area in Natural Language Processing, is the process of automatically detecting affective states of text. Sentiment analysis is widely applied to voice-of- customer materials such as product reviews in online shopping websites like Amazon, movie reviews or social media. It can be just a basic task of classifying the polarity of a text as being positive/negative or it can go beyond polarity, looking at sentiment states etc. Sentiment analysis refers to analyzing an opinion or feelings about something using data like text or images, regarding almost anything. Sentiment analysis helps companies in their decision- making process. For instance, if public sentiment towards a product is not so good, a company may try to modify the product or stop the production altogether in order to avoid any losses. There are many sources of public sentiment e.g., public interviews, opinion polls, surveys, etc. However, with more and more people joining social media platforms, websites like Facebook and Twitter can be parsed for public sentiment.

# CONTENTS

<b>Declaration</b>	<b>i</b>
<b>Certificate</b>	<b>ii</b>
<b>Acknowledgements</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>Contents</b>	<b>v</b>
<b>List of Figures</b>	<b>vii</b>
<b>List of Tables</b>	<b>viii</b>

<b>1. Introduction.....</b>	<b>1</b>
1.1 Background.....	2
1.1.1 Motivation.....	2
1.2 Objective.....	2
1.3 Contribution.....	3
1.4 Report Organization.....	3
<b>2. Literature Survey.....</b>	<b>4</b>
<b>3. Methodology.....</b>	<b>5</b>
3.1 Introduction.....	5
3.2 Workflow.....	5
3.3 Software Environment.....	6
3.4 Implementation.....	8
<b>4. Agile Methodology.....</b>	<b>9</b>
4.1 Introduction.....	9
4.2 User Story.....	11
4.3 Product Backlog.....	11
4.4 Project Plan.....	12
4.5 Sprint Backlog Plan.....	14
4.6 Sprint Backlog (Actual).....	16
4.7 Product backlog review.....	18
4.8 Sprint Review.....	19



4.9 Testing and Validation .....	22
4.10 Git.....	23
<b>5. Results and Discussions .....</b>	<b>24</b>
5.1 Results.....	24
<b>6. Conclusion .....</b>	<b>34</b>
6.1 Summery .....	34
6.2 Future Work .....	34
<b>References.....</b>	<b>35</b>
<b>Source Code .....</b>	<b>36</b>
<b>Publication .....</b>	<b>40</b>

## List of Figures

3.2 workflow .....	5
4.10 Git.....	23
5.1 User interface.....	24
5.2 Output1.....	25
5.3 Output2.....	25
Dataflow Diagram.....	27
Activity Diagram.....	28

## List of Tables

4.2 User Story. ....	11
4.3 Product Backlog.....	12
4.4 Project plan .....	13
4.5 Sprint Backlog (Plan).....	14
4.6 Sprint Backlog (Actual) .....	16
4.7 Product backlog review.....	18
4.8 Sprint review.....	19
4.9 Testing and Validation.....	22

# CHAPTER 1

## INTRODUCTION

Sentiment analysis using machine learning is the process of training a machine learning model to classify text data as positive, negative, or neutral based on the underlying sentiment expressed in the text. Sentiment analysis can be used to automatically classify reviews, social media posts, and other forms of text data.

The following steps are involved in sentiment analysis using machine learning:

1. **Data Collection:** The first step in sentiment analysis is to collect the data that will be used to train the machine learning model. This can be done by web scraping or by using existing datasets.
2. **Data Preprocessing:** Once the data is collected, it must be preprocessed to remove noise and irrelevant data. This can be done by removing stop words, converting all text to lowercase, and removing punctuation.
3. **Feature Extraction:** After preprocessing, features must be extracted from the text data. This can be done using techniques such as bag-of-words or TF-IDF.
4. **Model Selection:** The next step is to select a machine learning model that is best suited for the task of sentiment analysis. Popular models include Support Vector Machines (SVM), Naive Bayes, and Convolutional Neural Networks (CNN).
5. **Training the Model:** Once a model is selected, it must be trained on the preprocessed data. The data is split into training and testing sets, and the model is trained on the training data.
6. **Model Evaluation:** After training the model, it must be evaluated on the testing data to determine its accuracy and performance.
7. **Deployment:** Once the model has been trained and evaluated, it can be deployed to classify new text data.

## **1.1 BACKGROUND**

Sentiment analysis is a subfield of natural language processing (NLP) that aims to automatically identify and extract the sentiment expressed in a piece of text. Sentiment analysis has gained popularity in recent years due to the growth of social media and the need for businesses to monitor and analyze customer feedback.

The traditional approach to sentiment analysis involved the use of lexicons, which are collections of words and their associated sentiment scores. However, this approach has limitations as it is difficult to maintain and update these lexicons, and it cannot handle the nuances and complexities of human language.

Machine learning has emerged as a powerful tool for sentiment analysis as it can learn from data and adapt to new situations. Machine learning models can be trained to automatically classify text data as positive, negative, or neutral based on the underlying sentiment expressed in the text. These models can be trained on large datasets of labeled text data, which are manually annotated with sentiment labels by human annotators.

### **1.1.1 MOTIVATION**

The motivation for sentiment analysis using machine learning is to automatically classify large amounts of text data as positive, negative, or neutral based on the underlying sentiment expressed in the text.

## **1.2 OBJECTIVE**

The objective of sentiment analysis using machine learning is to automatically classify text data as positive, negative, or neutral based on the underlying sentiment expressed in the text. This is achieved by training machine learning models on large datasets of labeled text data, which are manually annotated with sentiment labels by human annotators.

### 1.3 CONTRIBUTION

Sentiment analysis using machine learning has made significant contributions in several areas, including:

1. **Improved Customer Experience:** By analyzing customer feedback, sentiment analysis can help businesses identify customer pain points and take corrective actions to improve the customer experience. This can lead to increased customer satisfaction and loyalty.
2. **Brand Reputation Management:** Sentiment analysis can help businesses monitor and manage their brand reputation by identifying and addressing negative sentiment expressed by customers or in the media. This can help prevent reputational damage and maintain brand value.
3. **Political Analysis:** Sentiment analysis can be used to analyze public opinion on political issues and to predict the outcome of elections. This can provide valuable insights into voter preferences and help political parties to refine their messaging and campaign strategies.
4. **Market Research:** Sentiment analysis can be used to analyze the sentiment expressed in product reviews, social media posts, and other forms of text data to gain insights into customer preferences and trends. This can help businesses to develop products and services that better meet customer needs.
5. **Fraud Detection:** Sentiment analysis can be used to detect fraudulent activities such as fake reviews and social media manipulation. This can help businesses and organizations to protect their reputation and prevent financial losses.

### 1.4 REPORT ORGANIZATION

The project report is divided into six sections. Section 2 describes literature survey. Section 3 describes the methodology and section 4 describes agile methodology used for implementing The project. Section 5 gives the result and discussions. Finally, section 6 gives the conclusion.

## CHAPTER 2

### LITERATURE SURVEY

A literature survey on sentiment analysis using machine learning would involve a comprehensive review of research papers, academic journals, and conference proceedings related to the topic. The survey would include a review of the following areas:

1. Sentiment analysis techniques: This involves a review of the various machine learning techniques used in sentiment analysis, such as Support Vector Machines (SVM), Naive Bayes, and Convolutional Neural Networks (CNN).
2. Feature extraction: This involves a review of the various feature extraction techniques used in sentiment analysis, such as Bag of Words (Bow), Word Embeddings, and Part of Speech (POS) tagging.
3. Sentiment lexicons: This involves a review of the various sentiment lexicons used in sentiment analysis, such as the General Inquirer and Sent WordNet.
4. Sentiment analysis applications: This involves a review of the various applications of sentiment analysis, such as customer feedback analysis, social media monitoring, brand reputation management, and political analysis.
5. Evaluation metrics: This involves a review of the various evaluation metrics used to assess the performance of sentiment analysis models, such as accuracy, precision, recall, and F1- score.

The literature survey would also involve a critical analysis of the strengths and limitations of existing research in sentiment analysis using machine learning, and identify areas for future research and development. Overall, a literature survey on sentiment analysis using machine learning would provide a comprehensive overview of the state of the art in the field and the latest advances in sentiment analysis research.

## CHAPTER 3

### METHODOLOGY

#### 3.1 INTRODUCTION

Sentiment analysis using machine learning is the process of training a machine learning model to classify text data as positive, negative, or neutral based on the underlying sentiment expressed in the text. Sentiment analysis can be used to automatically classify reviews, social media posts, and other forms of text data.

#### 3.2 WORKFLOW

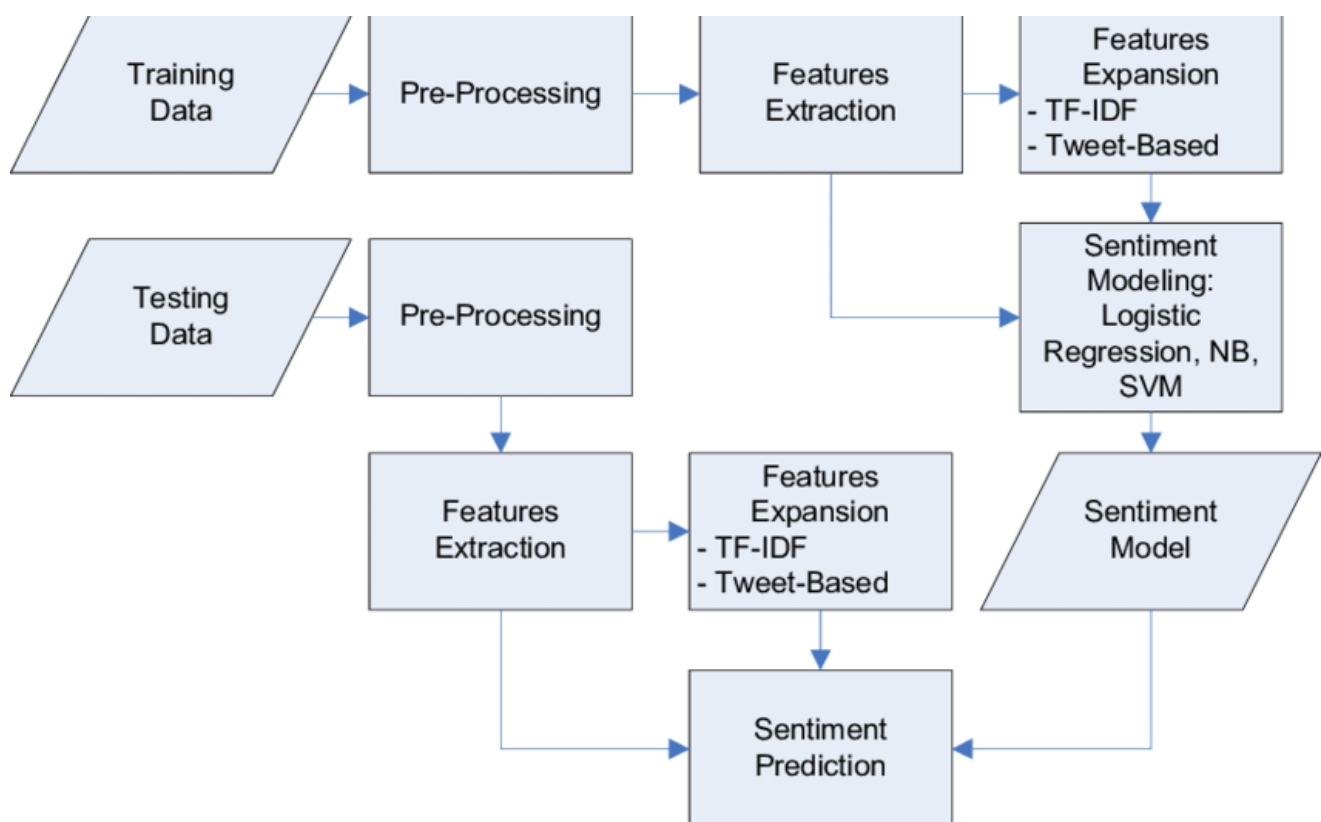


Figure 3.2: Workflow Diagram

### 3.3 SOFTWARE ENVIRONMENT

#### Python:

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** – You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** – Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

#### Python Features:

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy-to-maintain.
- **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** – You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.



## What is Python?

Python is a popular programming language. It was created in 1991 by Guido van Rossum.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

## What can Python do?

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

## Why Python?

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-orientated way or a functional way.
- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.

- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, PyCharm, NetBeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed to for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

### 3.4 IMPLEMENTATION

#### Implementation Platform and Language

Python is a general purpose, interpreted-high level language used extensively nowadays for solving domain problems instead of dealing with complexities of a system. It is also termed as the ‘batteries included language’ for programming. It has various libraries used for scientific purposes and inquiries along with number of third-party libraries for making problem solving efficient.

In this work, the Python libraries of NumPy, for scientific computation, and Matplotlib, for 2D plotting have been used. Along with this, Panda’s tool of Python has been employed for carrying out data analysis. As a development platform, PyCharm, which proves to work great due to its excellence in ‘literate programming’, where human friendly code is punctuated within code blocks, has been used.

## CHAPTER 4

### AGILE METHODOLOGY

#### 4.1 INTRODUCTION

After the initial studies it is found that agile model of software development is suitable and is the best method for the development of this system. Agile methodology mainly focused on the client satisfaction through continuous delivery. Also it sets a minimum number of requirements and turns them in to a deliverable product. As this project has many individual requirements which can be delivered in parts and the user can gradually improve their work efficiency. Agile methodology has a family of methods of which scrum is selected for the development of this project. Scrum is process framework that has been used to manage complex product development. It is not a process or technique for building products rather it is a framework within which various processes can be employed. Also, it is suitable method to support the development process. It focuses on lean software development and has in building better software effectively and efficiently.

Agile is one of the most widely used and recognized software development frameworks. The methodology those experts agreed upon was described as 'lightweight' and fast. Agile is also about being the adaptive and continuous improvement, as much as it is about constant feedback and speed of delivery.

Agile is a software development approach where a self-sufficient and cross-functional team works on making continuous deliveries through iterations and evolves throughout the process by gathering feedback from the end users. The major rules in scrum methodology are:

1. **The product owner (PO)** : Who represents the stakeholder and the business.
2. **The scrum master** : Ensures the process followed, removes obstructions, and protects the development system
3. **Development team**: Cross functional, self-organizing team who actually do the actual analysis, design implementation and testing process.

They work together in iterative time boxed durations called sprints. The first step is the creation of the product backlog by the PO. It's a to-do list of stuff to be done by the scrum team. Then the scrum team selects the top priority items and tries to finish them within the time box called a sprint. An easier way to remember all of this is to memorize the 3-3-5 framework. It means that a scrum project has 3 roles, 3 artifacts, and 5 events.

**These are:-**

1. **Roles:** Product Owner, Scrum Master, and development team.
2. **Artifacts:** Product Backlog, Sprint Backlog and Product Increment.
3. **Events:** Sprint, Sprint planning, Daily Scrum, Sprint review and Sprint retrospective

The framework begins with a simple premise start with what can be seen or known. After that the progress is tracked and tweak as necessary. The three pillars of scrum are transparency, inspection and adaptation. In scrum everyone has a role.

The Git is used as the version control system for this project. Version control is a system that records changes to a file or set of files over time so that a specific version can be recalled later. Version control systems are a category of software tools that help a software team for managing changes to source code over time. Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

## 4.2 USER STORY

A user story is a tool used in agile software development to capture a description of a software feature from an end-user perspective. The user story describes the type of user, what they want and why. A user story helps to create a simplified description of a requirement.

User story ID	As a <Type of user>	I want to perform <some task>	So that I can <achieve some goal>
1	User	Dataset Generation	Collect the required dataset
2	User	words classification	Train the dataset to classify sentiments
3	User	Data detection	Analysis of sentiments.

**Table 4.2: User Story**

## 4.3 PRODUCT BACKLOG

A product backlog is a list of the new features, changes to existing features, bug fixes, infrastructure changes or other activities that a team may deliver in order to achieve a specific outcome. The product backlog is the single authoritative source for things that a team works on. That means that nothing gets done that isn't on the product backlog. Conversely, the presence of a product backlog item on a product backlog does not guarantee that it will be delivered. It represents an option the team has for delivering a specific outcome rather than a commitment.

It should be cheap and fast to add a product backlog item to the product backlog, and it should be equally as easy to remove a product backlog item that does not result in direct progress to achieving the desired outcome or enable progress toward the outcome. The Scrum Product Backlog is simply a list of all things that needs to be done within the project. It replaces the traditional requirements specification artifacts. These items can have a technical nature or can be user-centric e.g. in the form of user stories. The product backlog of the system is given in Table 4.2

PRODUCT BACKLOG			
ID	NAME	PRIORITY	ESTIMATE[Hrs.]
1	Dataset Generation	1	60
2	words classification	2	70
3	Data detection	3	90

**Table 4.3: Product Backlog**

## 4.4 PROJECT PLAN

A project plan that has a series of tasks laid out for the entire project, listing task durations, responsibility assignments, and dependencies. Plans are developed in this manner based on the assumption that the Project Manager, hopefully along with the team, can predict up front everything that will need to happen in the project, how long it will take, and who will be able to do it. Project plan is given in Table 4.3

User story ID	Task Name	Start date	End date	Days	Status (to be filled by Scrum Master)
<b>Sprint 1</b>		<b>25-01-2023</b>	<b>17-02-2023</b>	<b>17</b>	<b>Completed</b>
1	Dataset generation	25-01-2023	02-02-2023	6	Completed
2	Coding	03-02-2023	10-02-2023	6	Completed
3	Testing	11-03-2023	17-02-2023	5	Completed
<b>Sprint 2</b>		<b>18-02-2023</b>	<b>16-03-2023</b>	<b>19</b>	<b>Completed</b>
1	words Classification	18-03-2023	01-03-2023	9	Completed
2	Coding	02-03-2023	09-03-2023	5	Completed
3	Testing	10-04-2023	18-03-2023	5	Completed
<b>Sprint 3</b>		<b>17-03-2023</b>	<b>18-04-2023</b>	<b>19</b>	<b>Completed</b>
1	Data detection	17-03-2023	29-03-2023	9	Completed
2	Testing and Validation	30-03-2023	18-04-2023	10	Completed

**Table 4.4: Project plan**

The Project has Three sprints:

#### **4.4.1 Sprint 1**

Three tasks are planned in this sprint. First one is dataset generation , next is designing and initial coding .

#### **4.4.2 Sprint 2**

Three tasks are planned in this sprint. First one is gesture classification, Coding and next one is testing.

#### **4.4.3 Sprint 3**

Two tasks are planned in this sprint. First one is Data detection and next one is testing.

## 4.5 SPRINT BACKLOG (PLAN)

The sprint backlog is a list of tasks identified by the Scrum team to be completed during the Scrum sprint. During the sprint planning meeting, the team selects some number of product backlog items, usually in the form of user stories, and identifies the tasks necessary to complete each user story. Most teams also estimate how many hours each task will take someone on the team to complete.

### Sprint 1:

Three tasks are planned in this sprint. First one is dataset generation , next is designing and initial coding .Sprint backlog (planning) for sprint 1 is given in Table 4.4.

### Sprint 2:

Three tasks are planned in this sprint. First one is gesture classification, Coding and next one is testing. Sprint backlog (planning) for sprint 2 is given in Table 4.5

### Sprint 3:

Two tasks are planned in this sprint. First one is Data detection and next one is testing. The sprint backlog for sprint 3 is given in Table 4.6

Backlog item	Completion Date	Original Estimate in Hrs	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User Story #1 Hours			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours
Dataset generation	02-02-2023	24	2	0	3	0	4	3	0	4	2	0	2	4	0	0
Coding	10-02-2023	20	2	1	2	1	0	2	4	0	0	4	0	0	2	2
Testing	17-02-2023	24	1	4	0	4	1	0	1	2	3	2	2	1	3	0
Total		68	5	5	5	5	5	5	5	6	5	6	4	5	5	2

**Table 4.4.1: Sprint Backlog (Plan)-Sprint 1**



Backlog item	Completion Date	Original Estimate in Hrs	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User Story #1 Hrs			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours
words Classification	01-03-2023	36	3	0	4	2	4	0	2	2	4	3	2	4	4	2
Coding	09-03-2023	20	2	4	1	2	0	2	4	0	0	1	0	0	2	2
Testing	16-03-2023	20	1	2	1	2	2	4	0	3	1	0	2	0	0	2
Total		76	6	6	6	6	6	6	6	5	5	4	4	4	6	6

Table 4.4.2: Sprint Backlog (Plan)-Sprint 2

Backlog item	Completion Date	Original Estimate in Hrs	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User Story #1 Hours			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours
Data detection	29-03-2023	36	4	0	3	2	2	0	4	2	4	3	2	4	4	2
Testing & Validation	18-04-2023	40	2	4	3	3	4	4	2	4	2	2	4	2	2	2
Total		76	6	4	6	5	6	4	6	6	6	5	6	6	6	4

Table 4.4.3: Sprint Backlog (Plan)-Sprint 3

## 4.6 SPRINT BACKLOG (ACTUAL)

Actual sprint backlog is what adequate sprint planning is actually done by project team there may or may not be difference in planned sprint backlog. The detailed sprint backlog (Actual) is given below.

Backlog item	Completion Date	Original Estimate in Hrs	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User Story #1 Hours			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours
Dataset generation	02-02-2023	24	24	2	0	3	0	4	3	0	4	2	0	2	4	0
Coding	10-02-2023	24	24	1	4	0	4	1	0	1	2	3	2	2	1	3
Testing	17-02-2023	20	20	2	1	2	1	0	2	4	0	0	4	0	0	2
Total		68	68	5	5	5	5	5	5	5	6	5	6	4	5	5

**Table 4.6.1: Sprint Backlog (Actual)-Sprint 1**

Backlog item	Completion Date	Original Estimate in Hrs	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User Story #1 Hrs			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours
words Classification	01-03-2023	36	3	0	4	2	4	0	2	2	4	3	2	4	4	2
Coding	09-03-2023	20	2	4	1	2	0	2	4	0	0	1	0	0	2	2
Testing	16-03-2023	20	1	2	1	2	2	4	0	3	1	0	2	0	0	2
Total		76	6	6	6	6	6	6	6	5	5	4	4	4	6	6

Table 4.6.2: Sprint Backlog (Actual)-Sprint 2

Backlog item	Completion Date	Original Estimate in Hrs	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User Story #1 Hours			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours
Data detection	29-03-2023	36	4	0	3	2	2	0	4	2	4	3	2	4	4	2
Testing & Validation	18-04-2023	40	2	4	3	3	4	4	2	4	2	2	4	2	2	2
Total		76	6	4	6	5	6	4	6	6	6	5	6	6	6	4

Table 4.6.3: Sprint Backlog (Actual)-Sprint 3

## 4.7 PRODUCT BACKLOG REVIEW

### REVIEW FORM

#### SPRINT 1

**Version:1.0**

**Date:17/02/2023**

User story ID	Comments From Scrum master if any	Comments from Product Owner if any
1	Developer should have a easy access process.	User friendly
2	Effective dataset	If there is relevant dataset

**Table 4.7.1: Product Backlog Review (Sprint1)**

#### SPRINT 2

**Version:1.0**

**Date:16/03/2023**

User story ID	Comments from Scrum master if any	Comments From Product Owner if any
3	should check the data inserted correctly	Inserted
4	Classify the words	Should classifies different models

**Table 4.7.2: Product Backlog Review (Sprint2)**

## SPRINT 3

**Version:1.0**

**Date:18/04/2023**

User story ID	Comments from Scrum master if any	Comments From Product Owner if any
5	should check corresponding data	Check result
6	Detecting the data	Compare dataset

**Table 4.7.3: Product Backlog Review (Sprint3)**

### 4.8 SPRINT REVIEW

At the end of each sprint a Sprint Review meeting is held. During this meeting the Scrum Team shows which Scrum Product Backlog items they completed (according to the Definition of Done) during the sprint. This might take place in the form of a demo of the new features. Backlog items that are not completed shall not be demonstrated. Otherwise, this might suggest that these items are finished as well. Instead, incomplete items/remaining activities shall be taken back into the Scrum Product Backlog, re-estimated and completed in one of the following sprints. The Sprint Review meeting should be kept very informal. No PowerPoint slides should be used and time for preparation and performing the meeting should be limited. During the meeting the Scrum Product Owner inspects the implemented backlog entries and accepts the solution or adds new stories to the Scrum Product Backlog to adapt the functionality. Participants in the sprint review typically include the Scrum Product Owner, the Scrum Team and the Scrum Master. Additionally, management, customers, and developers from other projects might participate as well.

**REVIEW FORM****SPRINT 1****Version:1.0****Date:17/02/2023**

User story ID	Comments From Scrum master if any	Comments from Product Owner if any
1	Developer should have a easy access process	satisfied
2	effective system	successful

**Table 4.8.1: Sprint Review (Sprint1)****SPRINT 2****Version:1.0****Date:16/03/2023**

User story ID	Comments from Scrum master if any	Comments from Product Owner if any
3	should check the data inserted correctly	Successful
4	Classify the words	Satisfied

**Table 4.8.2: Sprint Review (Sprint2)**

**SPRINT 3****Version:1.0****Date:18/04/2023**

User story ID	Comments from Scrum master if any	Comments from Product Owner if any
5	should check corresponding data	successful
6	Detecting the data	Successful

**Table 4.8.3: Sprint Review (Sprint3)**

## 4.9 TESTING AND VALIDATION

### Sprint 1

Test #	Date	Action	Expected Result	Actual Result	Pass ? <Yes/ No>
1	01/03/2023	Dataset Generation	Dataset generation successful	successful	Yes

**Table 4.9.1: Testing and Validation(Sprint1)**

### Sprint 2

Test #	Date	Action	Expected Result	Actual Result	Pass ? <Yes/ No>
1	08/04/2023	Words Classification	Classify the word classification	Done	Yes

**Table 4.9.2: Testing and Validation(Sprint2)**

### Sprint 3

Test #	Date	Action	Expected Result	Actual Result	Pass ? <Yes/ No>
1	03/05/2023	Data Detection	Detect the original data	Done	Yes

**Table 4.9.3: Testing and Validation(Sprint3)**



## 4.10 GIT

The Git is used as the version control system for this project. Version control is a system that records changes to a file or set of files over time so that a specific version can be recalled later. changes to source code over time. Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

Git is a free and open-source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. To show the continuous development of the project the Gitlab histories are shown in below:

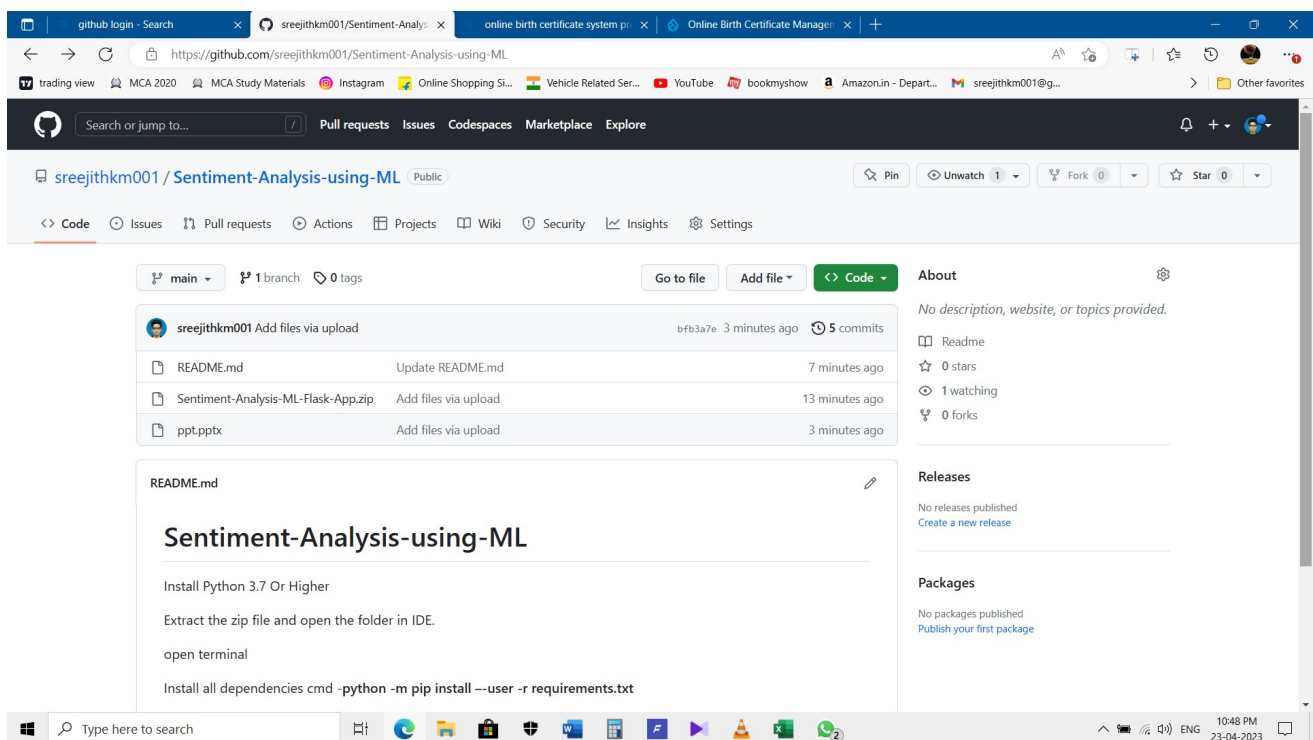
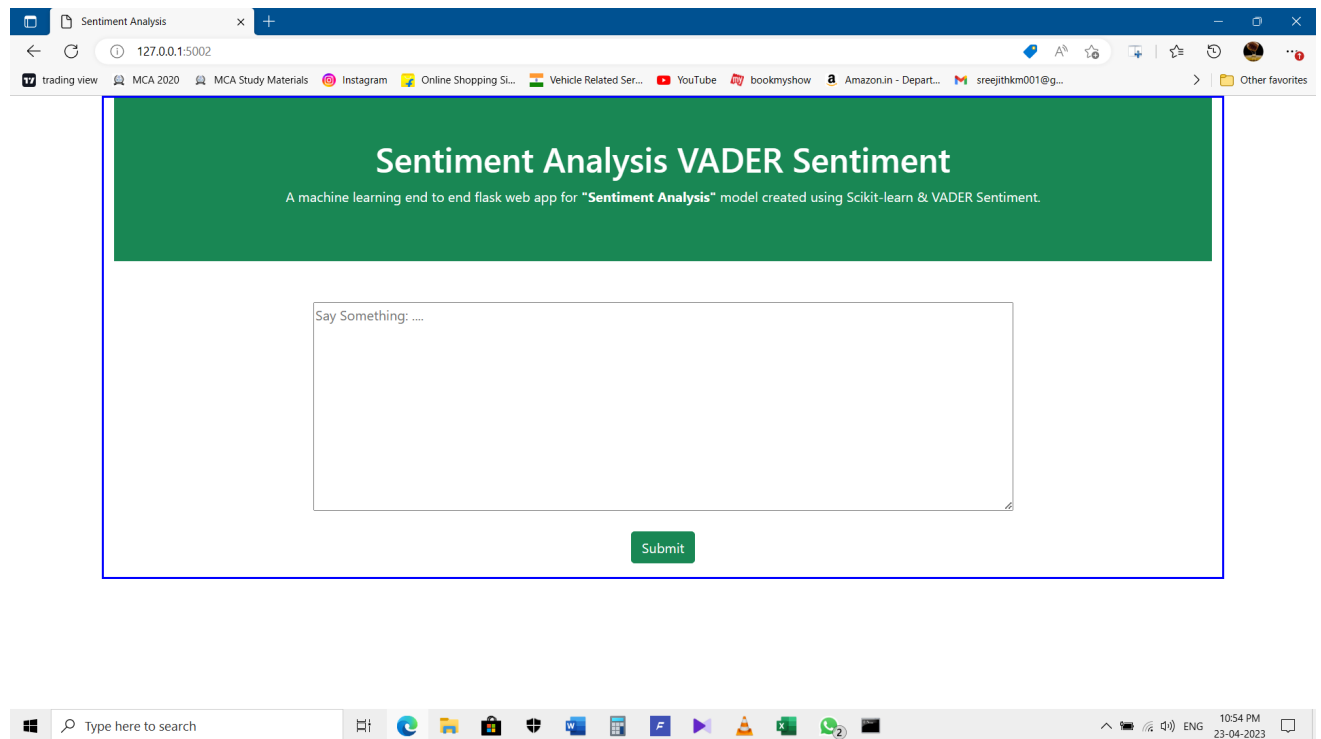


Figure 4.10 Git

## CHAPTER 5

### RESULT AND DISCUSSIONS



**Figure 5.1 User Interface**

## Final Result:

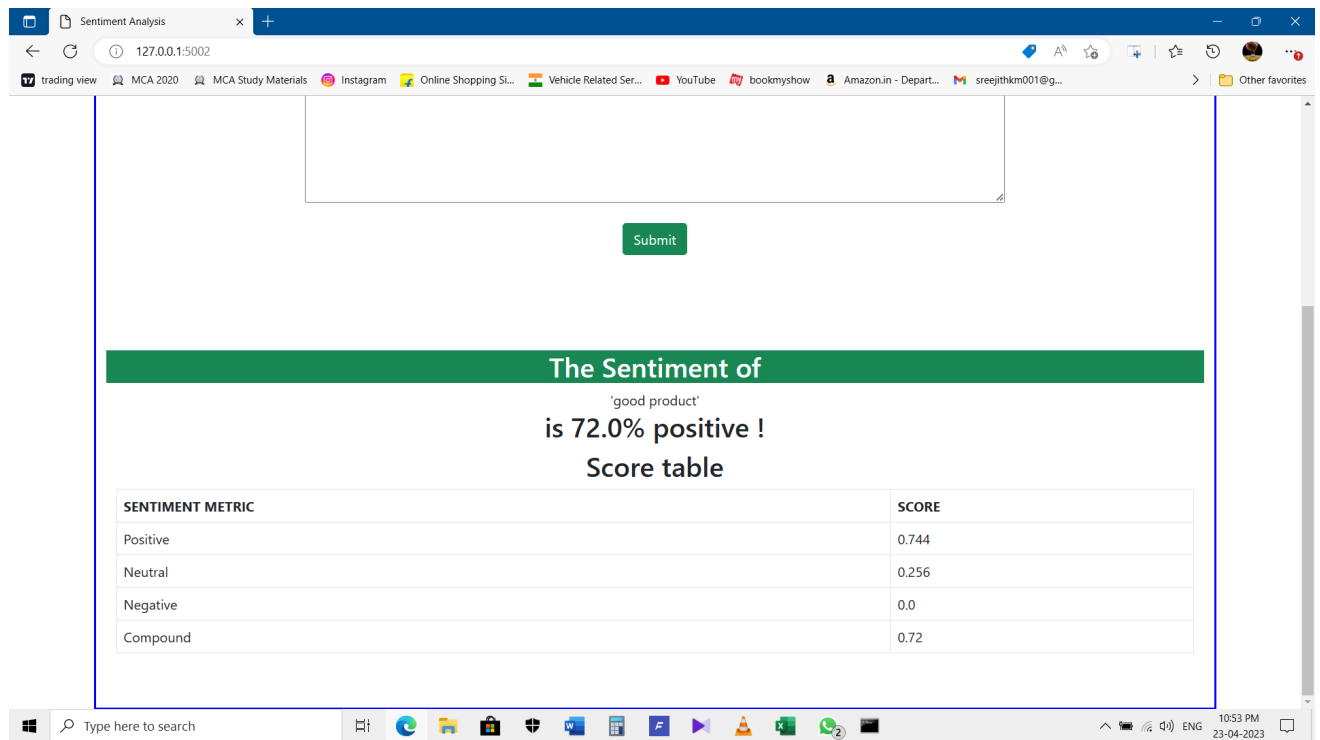


Figure 5.2 Output1

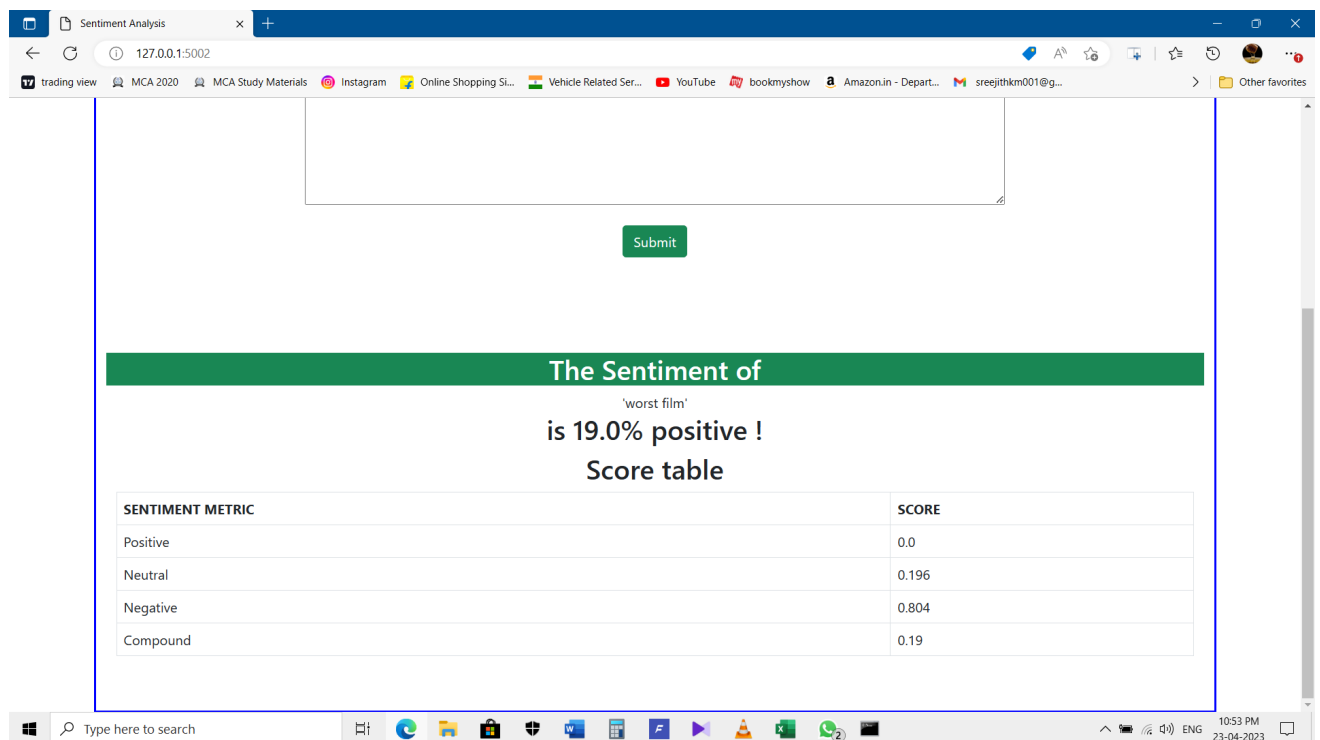
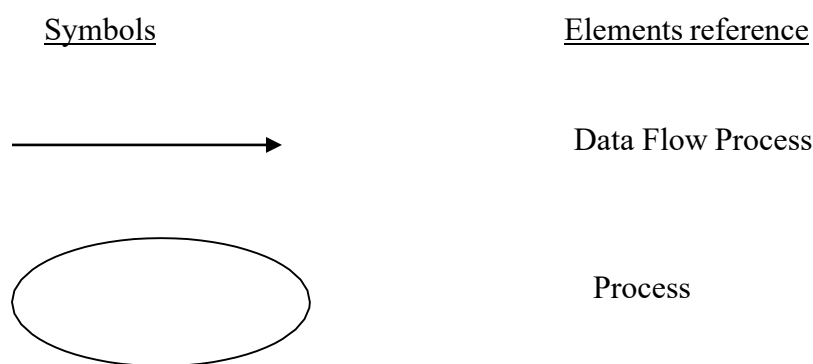


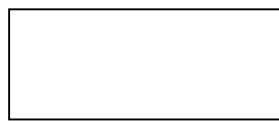
Figure 5.3 Output2

## SYSTEM DESIGN

### DATAFLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

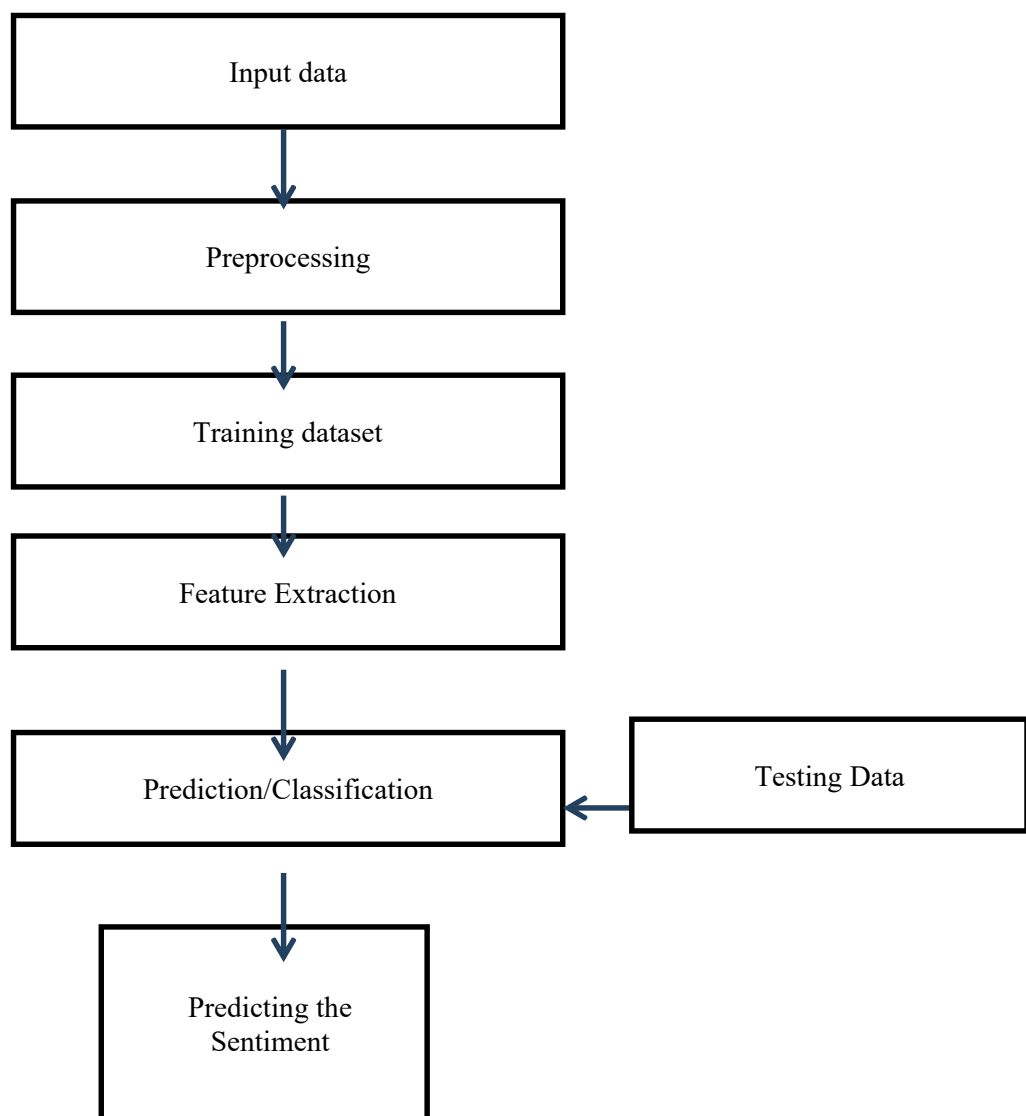




Source link



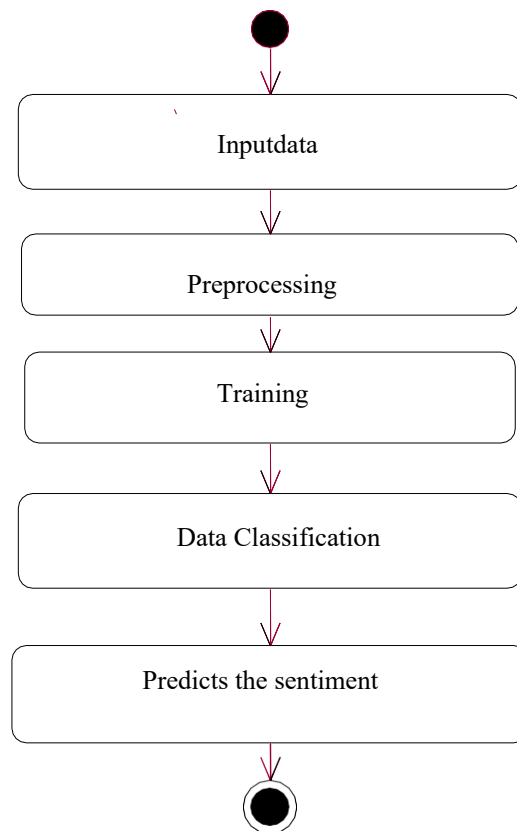
Data Store



**Dataflow Diagram**

## ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



**Activity Diagram**

## SYSTEM REQUIREMENTS

### Hardware Requirements:

- System : Intel Core i5.
- Hard Disk : 1TB.
- Monitor : 15 LED
- Ram : 4 GB

### Software Requirements:

- Operating system : Linux/Windows 10.
- Coding Language : Python
- IDE : PyCharm/VS Code

## PYTHON LIBRARIES :

### 1. FLASK:

Flask is an API of Python that allows us to build up web-applications. It was developed by Armin Ronacher. Flask's framework is more explicit than Django's framework and is also easier to learn because it has less base code to implement a simple web-Application. A Web-Application Framework or Web Framework is the collection of modules and libraries that helps the developer to write applications without writing the low-level codes such as protocols, thread management, etc. Flask is based on WSGI(Web Server Gateway Interface) toolkit and Jinja2 template engine.

### 2. SKLEARN

Scikit-Learn is a free machine learning library for Python. It supports both supervised and unsupervised machine learning, providing diverse algorithms for classification, regression, clustering, and dimensionality reduction. The library is built using many libraries you may already be familiar with, such as NumPy and SciPy. It also plays well with other libraries, such as Pandas and Seaborn.

### 3. REQUESTS:

Request's library is one of the integral part of Python for making HTTP requests to a specified URL. Whether it be REST APIs or Web Scraping, requests is must to be learned for proceeding further with these technologies. When one makes a request to a URI, it returns a response. Python requests provides inbuilt functionalities for managing both the request and response.

### 4. NLP:

Natural language processing (NLP) is a field that focuses on making natural human language usable by computer programs. NLTK, or Natural Language Toolkit, is a Python package that you can use for NLP. A lot of the data that you could be analyzing is unstructured data and contains human-readable text. Before you can analyze that data programmatically, you first need to pre-process it. In this tutorial, you'll take your first look at the kinds of text pre-processing tasks you can do with NLTK so that you'll be ready to apply them in future projects. You'll also see how to do some basic text analysis and create visualizations. If you're familiar with the basics of using Python and would like to get your feet wet with some NLP, then you've come to the right place.

### 5. VADER:

VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. VADER uses a combination of A sentiment lexicon is a list of lexical features (e.g., words) which are generally labelled according to their semantic orientation as either positive or negative. VADER not only tells about the Positivity and Negativity score but also tells us about how positive or negative a sentiment is.



## **SYSTEM TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

## **TYPES OF TESTS**

### **Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

### **Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successful unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

## **System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

## **White Box Testing**

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

## **Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

## **Unit Testing:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

## **Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

## **Test objectives**

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

## Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

## Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

## Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

## CHAPTER 6

### 6.1 CONCLUSION:

This project tackles a fundamental problem of sentiment analysis, sentiment polarity categorization. Companies or Markets can use this project to take user reviews about their services or products. This is user friendly, free of cost and easy to use where the customers can type their sentiments or opinions about any product or services related to any company or market. Based on the sentiments, percentage strength (positive sentiments) and the percentage of weakness (negative sentiments) will be calculated and it will be displayed with accuracy results. By investigating and analyzing customer sentiments, these brands are able to get an inside look at consumer behaviors and, ultimately, better serve their audiences with the products, services and experiences they offer.

### 6.2 FUTURE WORK:

Sentiment analysis is a rapidly growing field, and there are several exciting areas of future work for sentiment analysis using machine learning. Here are some of them:

1. Fine-grained sentiment analysis: Current sentiment analysis techniques classify texts into positive, negative, or neutral categories. However, human emotions are more nuanced, and fine-grained sentiment analysis aims to capture this. Future work could involve developing machine learning models that can recognize and classify a wider range of emotions and sentiments.
2. Contextual sentiment analysis: Context plays a vital role in determining the sentiment of a text. Future work could involve developing machine learning models that can understand the context in which a text was written and adjust their sentiment analysis accordingly.
3. Multilingual sentiment analysis: As businesses expand globally, there is a growing need for sentiment analysis in multiple languages. Future work could involve developing machine learning models that can accurately analyze sentiments across different languages and cultures.
4. Domain-specific sentiment analysis: Different domains have different language styles and terminologies, making it difficult to apply general sentiment analysis models to specific domains. Future work could involve developing machine learning models that are trained on specific domains, such as social media, news, or product reviews.
5. Incorporating user feedback: Sentiment analysis models can be refined and improved by incorporating user feedback. Future work could involve developing machine learning models that can learn from user feedback and adapt their analysis accordingly.

## REFERENCES

- [1] J. Li, S. Fong, Y. Zhuang, and R. Khoury, “ Hierarchical Classification in Text Mining for Sentiment Analysis,” in 2014 International Conference on Soft Computing and Machine Intelligence, september. 2014, p. 46-51, doi: 10.1109/ISCMI.2014.37.
- [2] H. Parveen and S. Pandey, “Sentiment analysis on Twitter Dataset using Naive Bayes algorithm ,” in 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), july. 2016, p. 416-419, doi: 10.1109/ICATCCT.2016.7912034.
- [3] S. S. and P. K.v., “ Sentiment analysis of malayalam tweets using machine learning techniques ,” ICT Express, april. 2020, doi: 10.1016/j.icte.2020.04.003.
- [4] R. A. Tuhin, B. K. Paul, F. Nawrine, M. Akter, and A. K. Das, “ An Automated System of Sentiment Analysis from Bangla Text using Supervised Learning Techniques ,” in 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), february. 2019, p.360-364, doi: 10.1109/CCOMS.2019.8821658.
- [5] A. Poornima and K. S. Priya, “ A Comparative Sentiment Analysis Of Sentence Embedding Using Machine Learning Techniques ,” in 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), march 2020, p. 493-496, doi: 10.1109/ICACCS48705.2020.9074312.
- [6] H. S and R. Ramathmika, “ Sentiment Analysis of Yelp Reviews by Machine Learning ,” in 2019 International Conference on Intelligent Computing and Control Systems (ICCS), may 2019, p. 700-704, doi: 10.1109/ICCS45141.2019.9065812.
- [7] R. B. Shamantha, S. M. Shetty, and P. Rai, “Sentiment Analysis Using Machine Learning Classifiers: Evaluation of Performance,” in 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), f'evr. 2019, p. 21-25, doi:10.1109/CCOMS.2019.8821650.

## APPENDIX

### Source Code

#### App.py

```
from flask import Flask, request, render_template, session, flash
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
import nltk
from string import punctuation
import re
from nltk.corpus import stopwords

nltk.download('stopwords')

set(stopwords.words('english'))

app = Flask(__name__)

@app.route('/')
def my_form():
    return render_template('form.html')

@app.route('/', methods=['POST'])
def my_form_post():
    stop_words = stopwords.words('english')

    #convert to lowercase
    text1 = request.form['text1'].lower()

    text_final = ".join(c for c in text1 if not c.isdigit())

    #remove punctuations
    #text3 = ".join(c for c in text2 if c not in punctuation)

    #remove stopwords
    processed_doc1 = '.join([word for word in text_final.split() if word not in stop_words])

    sa = SentimentIntensityAnalyzer()
    dd = sa.polarity_scores(text=processed_doc1)
    compound = round((1 + dd['compound'])/2, 2)
```

```
return render_template('form.html', final=compound,  
text1=text_final,text2=dd['pos'],text5=dd['neg'],text4=compound,text3=dd['neu'])
```

```
if_name == "_main_":  
    app.run(debug=True, host="127.0.0.1", port=5002, threaded=True)
```

### **login.html**

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <meta http-equiv="X-UA-Compatible" content="ie=edge">  
    <title>Medlabs</title>  
    <link href="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/css/bootstrap.min.css"  
rel="stylesheet" id="bootstrap-css">  
    <script  
src="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/js/bootstrap.min.js"></script>  
    <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>  
</head>  
<style>  
  
.login-container{  
    margin-top: 5%;  
    margin-bottom: 5%;  
}  
.login-logo{  
    position: relative;  
    margin-left: -41.5%;  
}  
.login-logo img{  
    position: absolute;  
    width: 20%;  
    margin-top: 19%;  
    background: #282726;  
    border-radius: 4.5rem;  
    padding: 5%;  
}  
.login-form-1 {  
    padding: 9%;  
    background:#282726;  
    box-shadow: 0 5px 8px 0 rgba(0, 0, 0, 0.2), 0 9px 26px 0 rgba(0, 0, 0, 0.19);
```

```

}
.login-form-1 h3 {
  text-align: center;
  margin-bottom: 12%;
  color: #fff;
}
.login-form-2 {
  padding: 9%;
  background: #f05837;
  box-shadow: 0 5px 8px 0 rgba(0, 0, 0, 0.2), 0 9px 26px 0 rgba(0, 0, 0, 0.19);
}
.login-form-2 h3 {
  text-align: center;
  margin-bottom: 12%;
  color: #fff;
}
.btnSubmit {
  font-weight: 600;
  width: 50%;
  color: #282726;
  background-color: #fff;
  border: none;
  border-radius: 1.5rem;
  padding: 2%;
}
.btnForgetPwd {
  color: #fff;
  font-weight: 600;
  text-decoration: none;
}
.btnForgetPwd:hover {
  text-decoration: none;
  color: #fff;
}
.fon {
  color: white;
  text-align: center;
}
</style>
<body>

<div class="container login-container">
  <div class="row">
    <div class="col-md-6 login-form-1">

```



```

<h3>Application Login</h3>
<h6 class="font-weight-normal">Login here to access 3 Minute Depression Test</h6>

</div>
<div class="col-md-6 login-form-2">
  <div class="login-logo">
    
  </div>
  <h3>Login</h3>
  <form action="/login" method="POST">
    <div class="form-group">
      <input type="text" name="username" class="form-control"
placeholder="Your Username *" value="" />
    </div>
    <div class="form-group">
      <input type="password" name="password" class="form-control"
placeholder="Your Password *" value="" />
    </div>
    <div class="form-group">
      <input type="submit" class="btnSubmit" value="Login" />
    </div>
    <div class="form-group">
      <a href="#" class="btnForgetPwd" value="Login">Forget
Password?</a>
    </div>
  </form>
</div>
</div>
</body>

```

# **PUBLICATION**

## **SENTIMENT ANALYSIS USING MACHINE LEARNING**

Sreejith K M, MCA Scholar

Ms Sumi M, Assistant Professor

Department of MCA, Nehru College of Engineering and Research Centre, Thrissur, Kerala

### **ABSTRACT**

Sentiment analysis, an important area in Natural Language Processing, is the process of automatically detecting affective states of text. Sentiment analysis is widely applied to voice-of- customer materials such as product reviews in online shopping websites like Amazon, movie reviews or social media. It can be just a basic task of classifying the polarity of a text as being positive/negative or it can go beyond polarity, looking at sentiment states etc. Sentiment analysis refers to analyzing an opinion or feelings about something using data like text or images, regarding almost anything. Sentiment analysis helps companies in their decision- making process. For instance, if public sentiment towards a product is not so good, a company may try to modify the product or stop the production altogether in order to avoid any losses. There are many sources of public sentiment e.g., public interviews, opinion polls, surveys, etc. However, with more and more people joining social media platforms, websites like Facebook and Twitter can be parsed for public sentiment.

### **INTRODUCTON**

Sentiment analysis using machine learning is the process of training a machine learning model to classify text data as positive, negative, or neutral based on the underlying sentiment expressed in the text. Sentiment analysis can be used to automatically classify reviews, social media posts, and other forms of text data.

The following steps are involved in sentiment analysis using machine learning:

1. **Data Collection:** The first step in sentiment analysis is to collect the data that will be used to train the machine learning model. This can be done by web scraping or by using existing datasets.
2. **Data Preprocessing:** Once the data is collected, it must be preprocessed to remove noise and irrelevant data. This can be done by removing stop words, converting all text to lowercase, and removing punctuation.
3. **Feature Extraction:** After preprocessing, features must be extracted from the text data. This can be done using techniques such as bag-of-words or TF-IDF.
4. **Model Selection:** The next step is to select a machine learning model that is best suited for the task of sentiment analysis. Popular models include Support Vector Machines (SVM), Naive Bayes, and Convolutional Neural Networks (CNN).
5. **Training the Model:** Once a model is selected, it must be trained on the preprocessed data. The data is split into training and testing sets, and the model is trained on the training data.
6. **Model Evaluation:** After training the model, it must be evaluated on the testing data to determine its accuracy and performance.
7. **Deployment:** Once the model has been trained and evaluated, it can be deployed to classify new text data.

## **LITERATURE SURVEY**

A literature survey on sentiment analysis using machine learning would involve a comprehensive review of research papers, academic journals, and conference proceedings related to the topic. The survey would include a review of the following areas:

1. **Sentiment analysis techniques:** This involves a review of the various machine learning techniques used in sentiment analysis, such as Support Vector Machines (SVM), Naive Bayes, and Convolutional Neural Networks (CNN).
2. **Feature extraction:** This involves a review of the various feature extraction techniques used in sentiment analysis, such as Bag of Words (Bow), Word Embeddings, and Part of Speech (POS) tagging.

3. Sentiment lexicons: This involves a review of the various sentiment lexicons used in sentiment analysis, such as the General Inquirer and Sent WordNet.
4. Sentiment analysis applications: This involves a review of the various applications of sentiment analysis, such as customer feedback analysis, social media monitoring, brand reputation management, and political analysis.
5. Evaluation metrics: This involves a review of the various evaluation metrics used to assess the performance of sentiment analysis models, such as accuracy, precision, recall, and F1-score.

The literature survey would also involve a critical analysis of the strengths and limitations of existing research in sentiment analysis using machine learning, and identify areas for future research and development. Overall, a literature survey on sentiment analysis using machine learning would provide a comprehensive overview of the state of the art in the field and the latest advances in sentiment analysis research.

## **FUTURE WORK:**

Sentiment analysis is a rapidly growing field, and there are several exciting areas of future work for sentiment analysis using machine learning. Here are some of them:

1. Fine-grained sentiment analysis: Current sentiment analysis techniques classify texts into positive, negative, or neutral categories. However, human emotions are more nuanced, and fine-grained sentiment analysis aims to capture this. Future work could involve developing machine learning models that can recognize and classify a wider range of emotions and sentiments.
2. Contextual sentiment analysis: Context plays a vital role in determining the sentiment of a text. Future work could involve developing machine learning models that can understand the context in which a text was written and adjust their sentiment analysis accordingly.
3. Multilingual sentiment analysis: As businesses expand globally, there is a growing need for sentiment analysis in multiple languages. Future work could involve developing machine learning models that can accurately analyze sentiments across different languages and cultures.

4. Domain-specific sentiment analysis: Different domains have different language styles and terminologies, making it difficult to apply general sentiment analysis models to specific domains. Future work could involve developing machine learning models that are trained on specific domains, such as social media, news, or product reviews.
5. Incorporating user feedback: Sentiment analysis models can be refined and improved by incorporating user feedback. Future work could involve developing machine learning models that can learn from user feedback and adapt their analysis accordingly.

## **CONCLUSION:**

This project tackles a fundamental problem of sentiment analysis, sentiment polarity categorization. Companies or Markets can use this project to take user reviews about their services or products. This is user friendly, free of cost and easy to use where the customers can type their sentiments or opinions about any product or services related to any company or market. Based on the sentiments, percentage strength (positive sentiments) and the percentage of weakness (negative sentiments) will be calculated and it will be displayed with accuracy results. By investigating and analyzing customer sentiments, these brands are able to get an inside look at consumer behaviors and, ultimately, better serve their audiences with the products, services and experiences they offer.

## REFERENCES:

- [1] J. Li, S. Fong, Y. Zhuang, and R. Khoury, “ Hierarchical Classification in Text Mining for Sentiment Analysis,” in 2014 International Conference on Soft Computing and Machine Intelligence, september. 2014, p. 46-51, doi: 10.1109/ISCMI.2014.37.
- [2] H. Parveen and S. Pandey, “Sentiment analysis on Twitter Dataset using Naive Bayes algorithm ,” in 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATecT), july. 2016, p. 416-419, doi: 10.1109/ICATCCT.2016.7912034.
- [3] S. S. and P. K.v., “ Sentiment analysis of malayalam tweets using machine learning techniques ,” ICT Express, april. 2020, doi: 10.1016/j.icte.2020.04.003.
- [4] R. A. Tuhin, B. K. Paul, F. Nawrine, M. Akter, and A. K. Das, “ An Automated System of Sentiment Analysis from Bangla Text using Supervised Learning Techniques ,” in 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), february. 2019, p.360-364, doi: 10.1109/CCOMS.2019.8821658.
- [5] A. Poornima and K. S. Priya, “ A Comparative Sentiment Analysis Of Sentence Embedding Using Machine Learning Techniques ,” in 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), march 2020, p. 493-496, doi: 10.1109/ICACCS48705.2020.9074312.
- [6] H. S and R. Ramathmika, “ Sentiment Analysis of Yelp Reviews by Machine Learning ,” in 2019 International Conference on Intelligent Computing and Control Systems (ICCS), may 2019, p. 700-704, doi: 10.1109/ICCS45141.2019.9065812.
- [7] R. B. Shamantha, S. M. Shetty, and P. Rai, “Sentiment Analysis Using Machine Learning Classifiers: Evaluation of Performance,” in 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), f'evr. 2019, p. 21-25, doi:10.1109/CCOMS.2019.8821650.



# NECTAR

## 2023

Nehru e-Conference on  
Technologies Annexing Reality  
Version 4 of NeCTAR Series

# Certificate of Participation

*This is to certify that*

*Sreejith K M*

*has presented a paper titled*

*SENTIMENT ANALYSIS USING MACHINE LEARNING*

*in Nehru e-Conference on Technology Annexing Reality  
held during May-June 2023 on Hybrid mode*

*Your Hardwork Achivement and Dedication will be cherished.*

*Your Article has been included in the Conference Proceedings bearing*

**ISBN 978-81-959223-5-2**



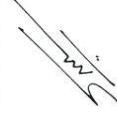
**Ashish L**

**Technical Head**



**Dr. Deepa A**

**Session Co.**



**Dr. Vineetha K R**

**Publication Co.**



**Prof. Dr. Sudheer Marar**

**Conference Chair**