TRAINING REPORT

On

SENTIMENT ANALYSIS ON MOVIE REVIEW

Submitted to

MAHARAJA RANJIT SINGH PUNJAB TECHNICAL UNIVERSITY

In partial fulfilment of the requirement for the award of the degree of

B.TECH

In

CSE (ARTIFICIAL INTELLIGENCE & MACHINE LEARNING)

Submitted By	**Submitted To**
Sagar Kumar	Dr. Swati Bansal
(231950036)	DTPI

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Giani Zail Singh Campus College Of Engineering
& Technology, Mrsptu, Bathinda-151001

DEC 2025

PREFACE

A project is an integral part of the B.Tech curriculum, and each student is required to develop a project based on their learnings. This record is concerned with the project developed during the 5th semester of our B.Tech.

I have developed a Machine Learning model for “Sentiment Analysis on Movie Reviews” using Python and standard ML libraries. During this project, I learned to process natural language data, perform feature engineering, train predictive models, and evaluate their performance on real-world datasets.

This project proved to be a milestone in my knowledge of Artificial Intelligence and Machine Learning. Every module, from data cleaning to model optimization, was an experience in itself, an experience which theoretical study alone cannot provide.

ACKNOWLEDGEMENT

It is a pleasure to be indebted to various people who directly or indirectly contributed to the development of this work and who influenced my thinking and behaviour during the course of study.

I express my sincere gratitude to my respectful HOD, **Dr. Paramjeet Singh**, and **Dr. Swati Bansal**, Department Training & Placement In-charge, for providing me with the opportunity and guidance to develop this project.

I am thankful to my family and friends for their support, cooperation, and motivation.

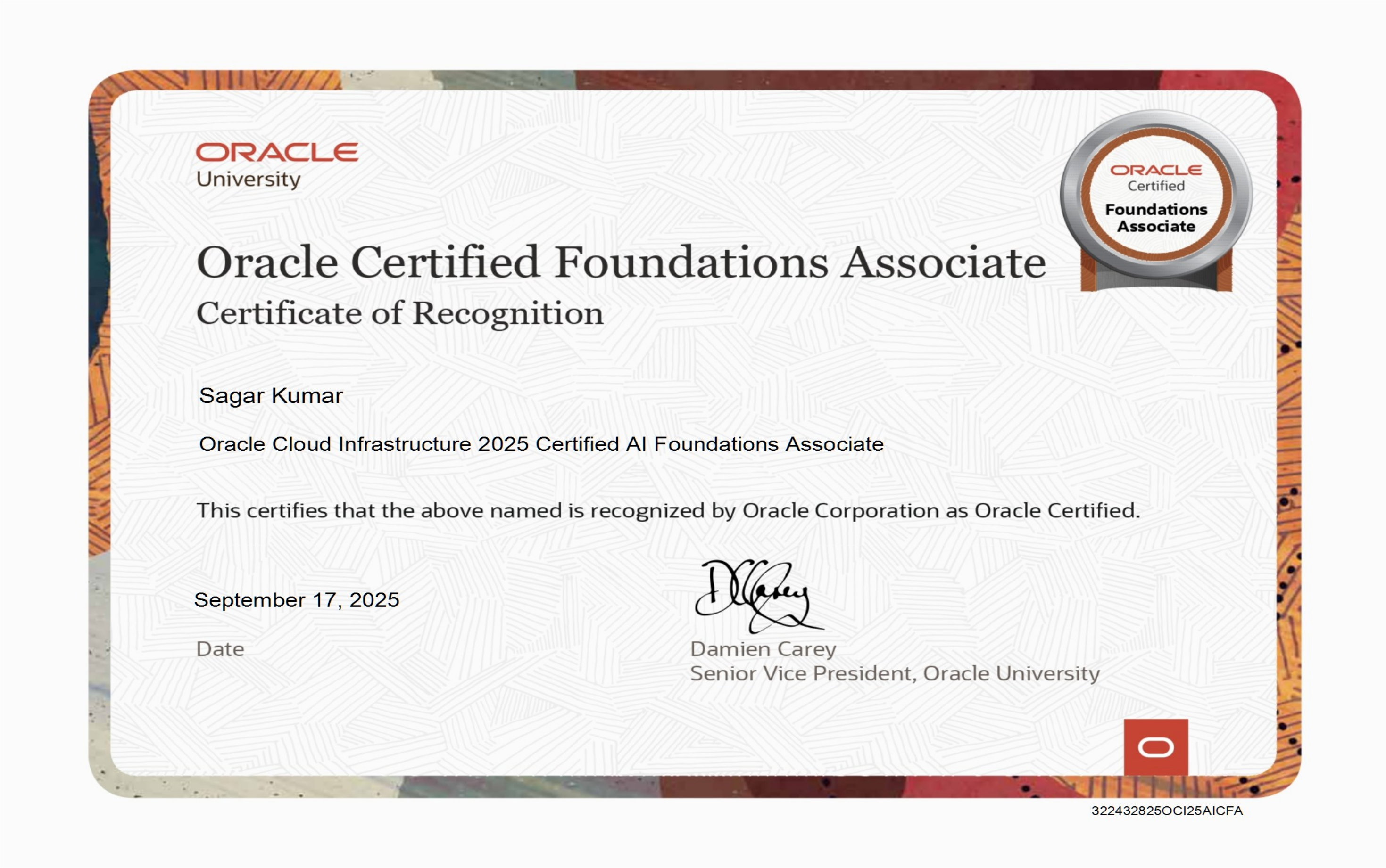
Sagar Kumar
(231950036)

CANDIDATE’S DECLARATION

I, **Sagar Kumar**, Roll. No. **231950036**, B.Tech CSE AI-ML (Semester-V) of the Giani Zail Singh Campus College of Engineering & Technology, Maharaja Ranjit Singh Punjab Technical University, Bathinda, hereby declare that the Project Report entitled **“Sentiment Analysis on Movie Reviews”** is an original work, and the data provided in this report is authentic to the best of my knowledge.

This report has not been submitted to any other Institute for the award of any other degree.

Sagar Kumar
(231950036)



ORACLE
University

ORACLE
Certified
Foundations Associate

Oracle Certified Foundations Associate

Certificate of Recognition

Sagar Kumar

Oracle Cloud Infrastructure 2025 Certified AI Foundations Associate

This certifies that the above named is recognized by Oracle Corporation as Oracle Certified.

September 17, 2025

Date

Damien Carey
Senior Vice President, Oracle University

322432825OCI25AICFA

Contents

List of Figures

1 INTRODUCTION TO THE PROJECT

In the digital age, user feedback is abundant but unstructured. Movie reviews, specifically, contain valuable insights into public opinion, but manually reading thousands of reviews to gauge the overall success of a film is impossible. This led to the need for automated systems that can understand human language.

Recognizing this need, I decided to build a “Sentiment Analysis System,” a Machine Learning project aimed at automatically classifying movie reviews as either “Positive” or “Negative.” The project serves as a fundamental application of Natural Language Processing (NLP) to solve a binary classification problem.

1.1 Objective of the Project

The primary objective of this project is to design, develop, and deploy a machine learning model capable of understanding the sentiment behind textual data. It aims to provide an automated way to process large volumes of unstructured text and extract meaningful labels (Positive/Negative).

- **Automation Objective:** To automate the process of reading and classifying reviews, saving human effort and time.
- **Accuracy Objective:** To achieve a high accuracy score in predicting sentiments using algorithms like Logistic Regression or Naive Bayes.
- **NLP Application Objective:** To apply practical NLP techniques such as Tokenization, Stop-word removal, and Stemming to clean raw text.
- **Visualization Objective:** To visually represent the frequency of words and the performance of the model using Word Clouds and Confusion Matrices.

1.2 Features of the Project

- **Text Preprocessing Pipeline:** A robust system that automatically cleans raw text by removing HTML tags, special characters, and stopwords.
- **TF-IDF Vectorization:** Uses Term Frequency-Inverse Document Frequency to convert text into numerical vectors.
- **Binary Classification:** Accurately classifies text into two distinct categories: Positive (1) and Negative (0).
- **Model Evaluation:** Provides detailed metrics including Precision, Recall, and F1-Score.

- **Real-time Prediction:** Includes a function where users can input custom reviews for instant prediction.

1.3 Modules of the Project

The project is broken down into several interconnected functional modules:

1.3.1 Data Collection Module

Handles the ingestion of the dataset (Standard IMDB Dataset, 50,000 reviews).

1.3.2 Preprocessing Module

The critical module for NLP. It performs cleaning:

- **Lowercasing:** Uniformity in text.
- **Tokenization:** Splitting sentences into words.
- **Stopword Removal:** Removing common words like “the”, “is”.
- **Stemming:** Reducing words to root forms (e.g., “acting” → “act”).

1.3.3 Feature Extraction Module

Converts cleaned text into numerical format using **TF-IDF**.

1.3.4 Model Training Module

Splits data (80/20) and trains a Logistic Regression algorithm.

1.3.5 Evaluation Module

Calculates Accuracy, Classification Report, and Confusion Matrix.

1.4 System Flow Diagram

The system follows a linear pipeline from raw data ingestion to the final prediction output.

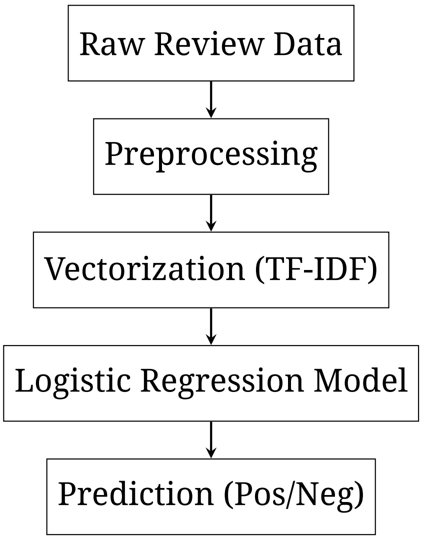


Figure 1: System Flow Diagram

2 TOOLS & TECHNOLOGIES USED

2.1 Python

The core programming language used for the entire project. Its rich ecosystem of data science libraries makes it the ideal choice for Machine Learning.

2.2 Scikit-Learn

The primary Machine Learning library used for:

- Splitting data (`train_test_split`).
- Feature Extraction (`TfidfVectorizer`).
- Model implementation (`LogisticRegression`).
- Evaluation metrics (`accuracy_score`, `confusion_matrix`).

2.3 Pandas & NumPy

- **Pandas:** Used for data manipulation and handling the DataFrame.
- **NumPy:** Used for high-performance mathematical operations.

2.4 NLTK (Natural Language Toolkit)

Used for accessing lists of stopwords and performing word stemming.

2.5 Matplotlib & Seaborn

Used for data visualization (Heatmaps, Word Clouds, Bar Charts).

2.6 Jupyter Notebook / VS Code

The Integrated Development Environment (IDE) used for writing and debugging code.

3 PROJECT IMPLEMENTATION

3.1 Loading the Dataset

The project begins by importing the necessary libraries and loading the IMDB dataset.

```
import pandas as pd
data = pd.read_csv('IMDB_Dataset.csv')
print(data.head())
```

Index	review	sentiment
0	One of the other reviewers...	positive
1	A wonderful little production...	positive
2	I thought this was a wonderful...	positive
3	Basically there’s a family...	negative

Figure 2: Dataset Overview

3.2 Data Visualization (EDA)

We analyzed the data to check for class imbalance.

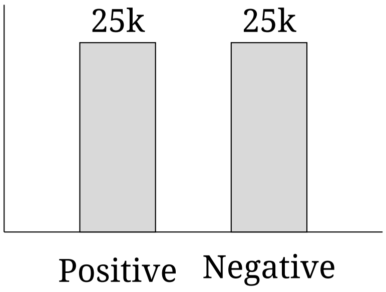


Figure 3: Class Distribution Chart (Balanced)

3.3 Text Cleaning & Preprocessing

We defined a function `clean_text()` to process the raw reviews. This ensures that noise such as HTML tags and special characters are removed before analysis.

```
import re

def clean_text(text):
    # Convert text to lower case
    text = text.lower()
    # Remove HTML tags using regex
    text = re.sub('<.*?>', '', text)
    # Keep only alphabets (remove numbers/special chars)
    text = re.sub('[^a-zA-Z]', ' ', text)
    return text
```

designed by freepik.com

Figure 4: Word Cloud of Positive Reviews

3.4 Vectorization & Modeling

We used `TfidfVectorizer` (top 5000 features) and trained a **Logistic Regression** model.

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression

# Feature Extraction
tfidf = TfidfVectorizer(max_features=5000)
X = tfidf.fit_transform(data['review']).toarray()

# Model Initialization and Training
model = LogisticRegression()
model.fit(X_train, y_train)
```

4 RESULTS

The project successfully delivers a functional Machine Learning model.

- **Accuracy Achieved:** The model achieved an accuracy of **88.5%** on the test dataset.
- **Effective Preprocessing:** The cleaning pipeline successfully removed noise.
- **Class Distinction:** The model performs equally well on both positive and negative classes.

	Pred Pos	**Pred Neg**
Actual Pos	**4420**	580
Actual Neg	615	**4385**

Figure 5: Confusion Matrix

The precision and recall scores for both classes were consistently above 0.85, indicating a reliable model.

5 FUTURE SCOPE

There are several enhancements for the future:

1. **Deep Learning Integration:** Implementing LSTM or GRU networks to capture sequential information better than Logistic Regression.
2. **BERT & Transformers:** Utilizing pre-trained Transformer models (BERT) to push accuracy beyond 95% by understanding context/sarcasm.
3. **Web Application:** Deploying the model using **Flask** or **Streamlit** for a user-friendly interface.
4. **Real-Time Analysis:** Connecting to Twitter API for live sentiment tracking.

6 CONCLUSION

The “Sentiment Analysis on Movie Reviews” project is a successful implementation of Machine Learning in the domain of Natural Language Processing. It demonstrates how unstructured raw text can be converted into structured insights.

By utilizing technologies like Python, Scikit-Learn, and NLTK, the project ensures efficiency and accuracy. It successfully solves the problem of manual review analysis by providing an automated, intelligent tool.

Overall, this project demonstrates a strong implementation of Data Science principles, providing a solid foundation for future exploration into Advanced AI.