# A Mini Project Report for Data Science & Big Data Analytics

**on**

**“TWITTER SENTIMENT ANALYSIS”**

****

**SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THIRD YEAR OF COMPUTER ENGINEERING**

**Submitted by**

###### 23.Pagire Siddhi Rajendra

**Under the Guidance of**

Prof. M.A.Tamboli



**DEPARTMENT OF COMPUTER ENGINEERING,**

**DR. VITHALRAO VIKHE PATIL COLLEGE OF ENGINEERING, AHILYANAGAR**

**SAVITRIBAI PHULE PUNE UNIVERSITY**

**ACADEMIC YEAR 2024-2025**

**DEPARTMENT OF COMPUTER ENGINEERING**

# CERTIFICATE

****

This is to certify that, 23.Pagire Siddhi Rajendra have successfully completed the Mini project entitled ―**Twitter Sentiment Analysis.**‖ under my guidance in partial fulfilment of the requirements for the Computer Engineering (TE) under the Savitribai Phule Pune University during the academic year 2024-2025

|  |  |
| --- | --- |
| **Prof.M.A.Tamboli** | **Prof. N. B. Vikhe** |
| **Project Guide** | **Head,**  **Department of Computer Engineering** |

#### ACKNOWLEDGMENT

With deep sense of gratitude, we would like to thank all the people who have lit our path with their kind guidance. We are very grateful to these intellectuals who did their best to help during our project work.

It is our proud privilege to express a deep sense of gratitude to **Dr. U. P. Naik,** Principal of DVVPCOE, Ahmednagar, for his comments and kind permission to complete this project. We remain indebted to **Prof. N.B. Vikhe,** Head. Computer Engineering Department for his timely suggestion and valuable guidance.

The special gratitude goes to **Prof.M.A.Tamboli** excellent and precious guidance in completion of this work. We thanks to all the colleagues for their appreciable help for our working project. With various industry owners or lab technicians to help, it has been our endeavour throughout our work to cover the entire project work.

We are also thankful to our parents who provided their wishful support for our project completion successfully. And lastly, we thank our all friends and the people who are directly or indirectly related to our project work.

23.Pagire Siddhi Rajendra

### Contents

1. **TITLE OF PROJECT:**

**“ TWITTER SENTIMENT ANALYSIS ”**

1. **ABSTRACT**
2. **INTRODUCTION**
3. **SCOPE**
4. **SPECIFIC REQUIREMENTS**
   * ***HARDWARE REQUIREMENTS***
   * ***SOFTWARE REQUIREMENTS***
5. **THEORY OF SOFTWARE USED**
   * ***PYTHON***
   * ***MySQL***
   * ***SCIKIT-LEARN***
   * ***NLTK***
6. **OUTPUT SCREEN(GUI)**
7. **SAMPLE CODE**
8. **CONCLUSION**
9. **REFERENCE**

## ABSTRACT

**Twitter Sentiment Analysis** is a data-driven project developed to extract, process, and analyze public opinions shared on Twitter, utilizing natural language processing (NLP) and machine learning techniques. The primary objective of this project is to demonstrate the application of data science principles in deriving actionable insights from unstructured text data, thereby enabling sentiment-based interpretation of social media content. This system employs Python for both data preprocessing and model development, integrating powerful libraries such as Pandas, NLTK, Scikit-learn, and Matplotlib to ensure efficient data handling, feature extraction, model training, and visualization.

The analytical workflow begins with the acquisition of real-time Twitter data via the Twitter API, followed by rigorous preprocessing steps including tokenization, stopword removal, stemming, and vectorization. These steps are critical for transforming raw textual input into structured data suitable for machine learning. The sentiment classification is performed using supervised learning models such as Logistic Regression, Naïve Bayes, or Support Vector Machines, which categorize tweets into positive, negative, or neutral sentiments based on learned patterns.

A key aspect of this project is its emphasis on data visualization and interpretability. Graphical representations such as word clouds, sentiment distribution plots, and confusion matrices are used to communicate findings and model performance effectively. From a data analytics standpoint, this project enables the tracking of public sentiment over time, which is valuable for trend analysis, brand monitoring, and public opinion research. Additionally, the project incorporates fundamental data security practices such as API key protection and controlled data access.

Designed with extensibility in mind, the sentiment analysis system can be enhanced with deep learning models, multilingual support, or integration into web-based dashboards, offering a scalable and insightful tool for real-world applications in social media monitoring and business intelligence.

## INTRODUCTION

##### Problem Definition

The rapid growth of social media platforms such as Twitter has led to the creation of massive amounts of textual data, which are often rich with opinions, emotions, and feedback on various topics. Analyzing this data manually is highly impractical due to its sheer volume and dynamic nature. The core challenge lies in developing an automated system capable of accurately classifying the sentiment of tweets—whether positive, negative, or neutral—while effectively handling the unstructured and noisy nature of social media text. This problem requires a robust solution that can process large datasets, utilize natural language processing techniques, and apply machine learning algorithms to classify sentiment with high accuracy.

* 1. **Problem Statement:** ―Develop an automated sentiment analysis system using machine learning and NLP techniques to classify Twitter data into positive and negative sentiments. The system should be capable of preprocessing raw data, handling large volumes of tweets, extracting meaningful features, and applying appropriate algorithms for sentiment classification.‖

##### Need for system:

###### Handling Large Volumes of Data

With millions of tweets generated every day on Twitter, manually analyzing such vast amounts of data is impractical and time-consuming. An automated sentiment analysis system allows for efficient processing and classification of large datasets in real-time, making it feasible to gain insights from social media at scale.

###### Faster Decision-Making

Automating the sentiment analysis process enables quicker decision-making. Businesses, researchers, and political analysts can immediately gauge public sentiment and adjust their strategies based on real-time feedback, whether for marketing campaigns, customer service, or public relations efforts.

###### Cost-Effective

Manual sentiment analysis requires a significant amount of human resources and time. Automating this process using machine learning reduces the need for extensive manpower, thus lowering costs and increasing overall efficiency in analyzing large volumes of textual data.

###### Consistency and Accuracy

Unlike human analysis, which can vary based on individual biases or fatigue, machine learning models can provide consistent and objective results. This ensures high accuracy in sentiment classification across large datasets, making the analysis reliable and repeatable.

###### Real-Time Insights

Automated sentiment analysis systems can provide real-time feedback, which is crucial for businesses and organizations that need to monitor trends, reactions, or changes in sentiment as events happen. This helps in proactive decision-making and timely response to emerging issues or opportunities.

## SCOPE

The scope of the **Twitter Sentiment Analysis** project involves classifying tweets into positive and negative sentiment categories using **Natural Language Processing (NLP)** and **Machine Learning (ML)** techniques. This project covers multiple key components necessary for building a sentiment classification system.

1. **Data Preprocessing**:The project begins with cleaning and preprocessing raw tweet data, which involves text cleaning, tokenization, stop word removal, and lemmatization to prepare the text for analysis.
2. **Feature Extraction**:Textual data is transformed into numerical features using TF-IDF (Term Frequency-Inverse Document Frequency) vectorization, which captures the importance of words in tweets relative to the entire dataset.
3. **Model Training and Evaluation:**The system explores multiple machine learning algorithms, such as Logistic Regression, Naive Bayes, and Support Vector Machines (SVM), for sentiment classification. The models are trained on labeled datasets and evaluated based on performance metrics like accuracy, precision, and recall.
4. **Sentiment Classification:**The primary aim of the project is to classify tweets into positive and negative categories, effectively identifying sentiment based on text input.
5. **Evaluation and Tuning:** To enhance model accuracy, different configurations and hyperparameters of the algorithms are tested, followed by cross-validation to assess model robustness.
6. **Future Enhancements:** Future improvements include adding neutral sentiment classification, expanding to multiple languages, integrating with the Twitter API for real- time analysis, and exploring advanced techniques like deep learning models for better contextual understanding.

#### SPECIFIC REQUIRMENTS

The system analysis contains a planning and design phases where a logical design of system is developed and to work accordingly a plan is established. Also the requirements of system are identified and the operating environment is identified.

##### Hardware Requirements

* + **RAM**: 4 GB or more
  + **Processor**: Intel Core i5 or higher
  + **Memory Space**: 1 GB free space
  + **Operating System**: Windows 10 / Linux / macOS

##### Software Requirements

* + **Programming Language**: Python 3.x
  + **IDE/Platform**: Jupyter Notebook / Google Colab

###### Libraries:

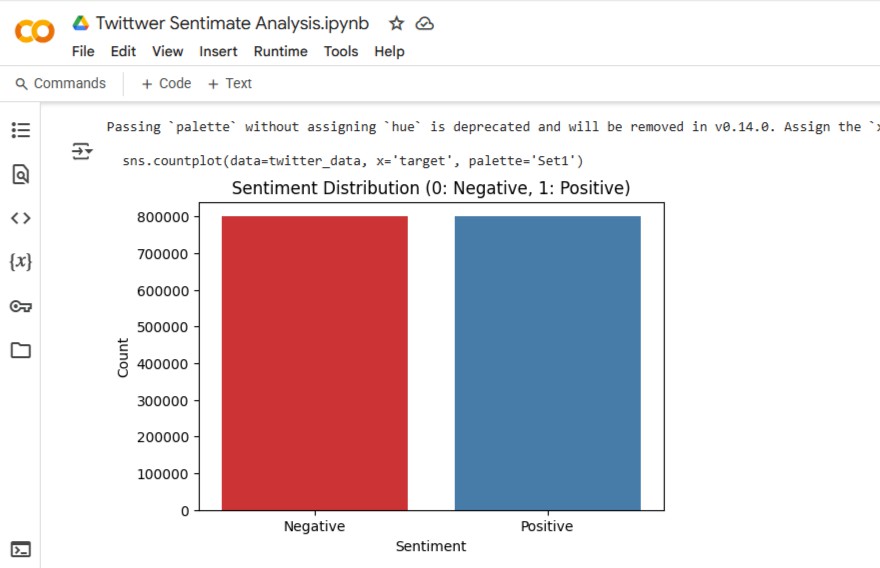
* + - **Pandas** (for data manipulation)
    - **NumPy** (for numerical computations)
    - **Scikit-learn** (for machine learning algorithms and model evaluation)
    - **NLTK** (for text processing and NLP tasks)
    - **Matplotlib** (for data visualization)
    - **re** (for regular expressions in text cleaning)
  + **Database (optional)**: MySQL (for data storage and management, if used)

#### SOFTWARE USED

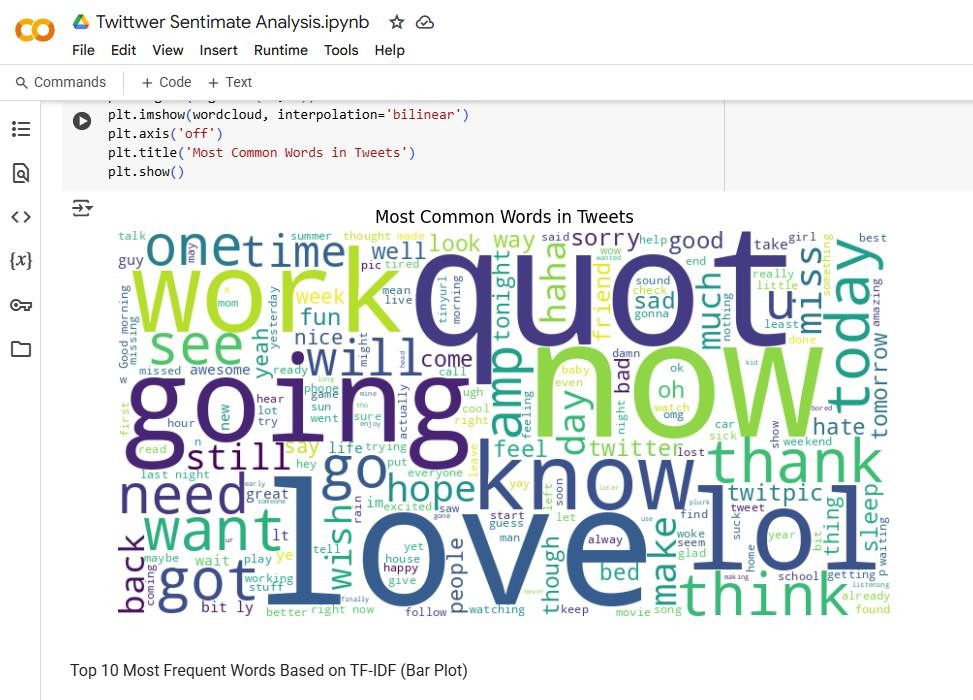
1. **Python :** Python is a versatile, high-level programming language that is widely used for data analysis, machine learning, and web development. It is known for its simplicity and readability, making it ideal for building machine learning models for tasks like sentiment analysis. Python's extensive libraries, such as Pandas for data manipulation, Scikit-learn for machine learning algorithms, and NLTK for natural language processing (NLP), make it a perfect choice for this project.
2. **MySQL :** MySQL is an open-source relational database management system (RDBMS) that is used for storing structured data in a systematic way. In this project, MySQL is employed for managing and retrieving data such as tweet content, user details, and sentiment analysis results. Its speed, reliability, and support for large datasets make it suitable for handling data throughout the project's pipeline.
3. **Scikit-learn :** Scikit-learn is a powerful machine learning library in Python that provides tools for data mining and data analysis. It offers a wide range of algorithms for classification, regression, and clustering tasks. For this sentiment analysis project, Scikit-learn is used to train and evaluate models that classify tweets into positive or negative categories, using techniques like TF-IDF vectorization and various machine learning algorithms such as Naive Bayes and Logistic Regression.
4. **NLTK (Natural Language Toolkit) :** NLTK is a Python library designed for working with human language data (text). It includes tools for text preprocessing tasks such as tokenization, lemmatization, stopword removal, and stemming, which are essential steps for preparing the tweet data before applying machine learning models. NLTK makes it easier to perform natural language processing tasks and ensures that the text data is clean and structured for analysis.

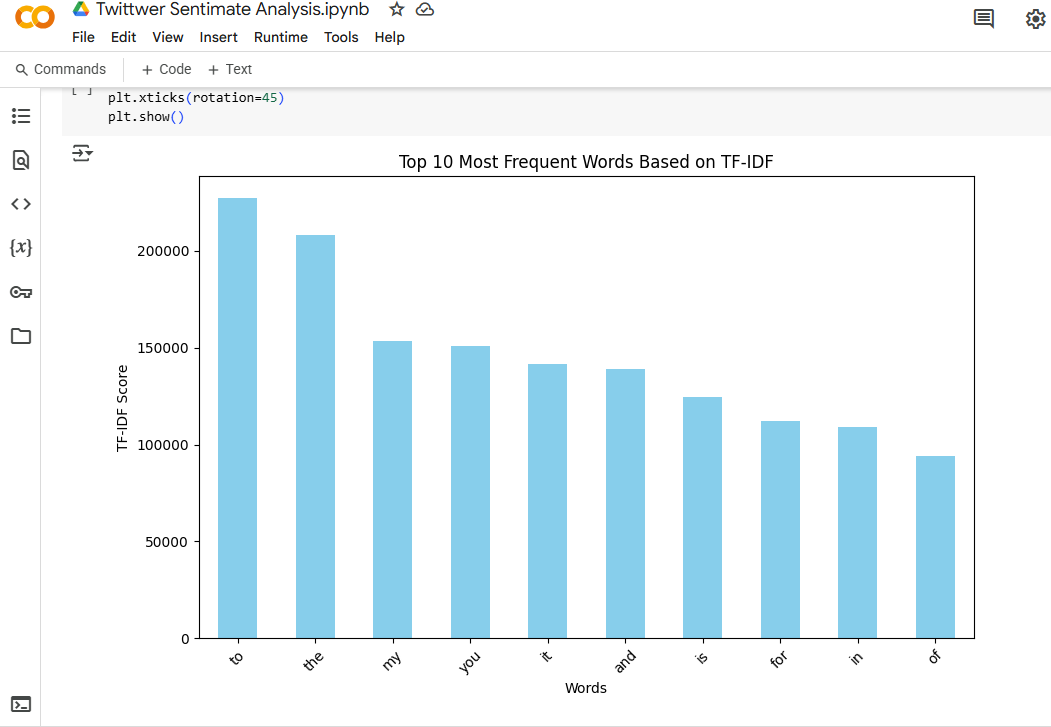
#### SCREEN OUTPUT

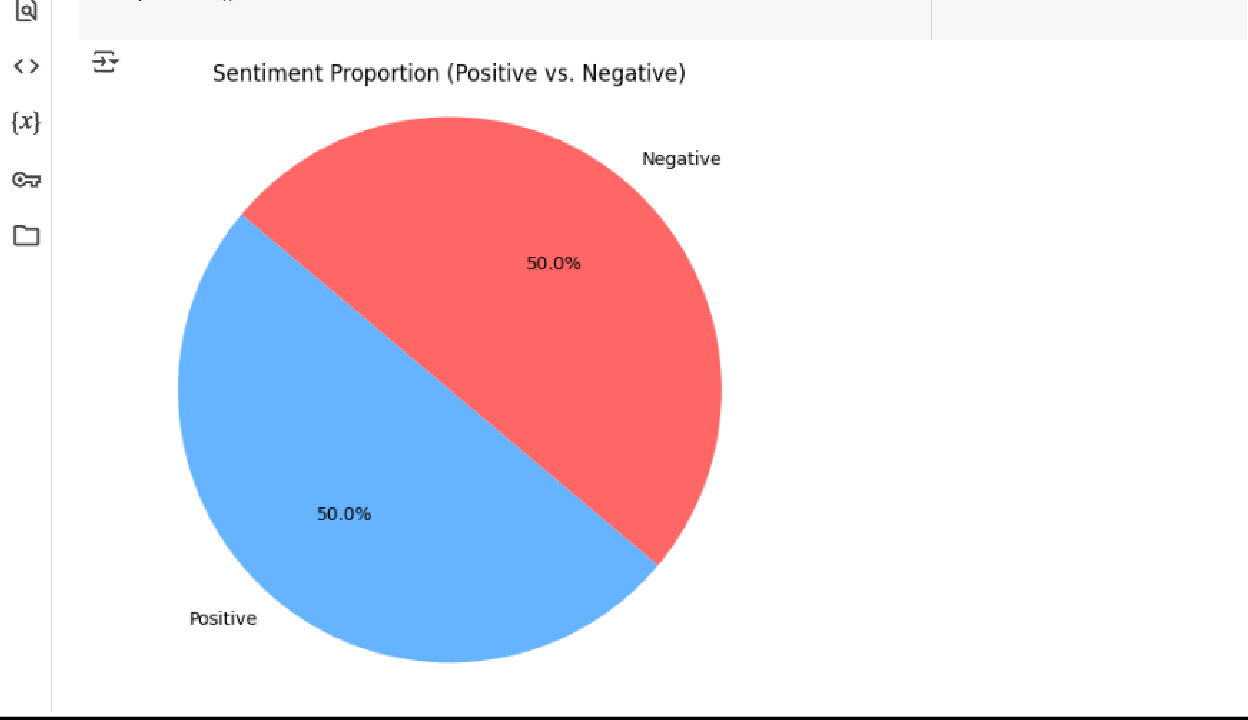
1. **Sentiment Distrubution Graph:**

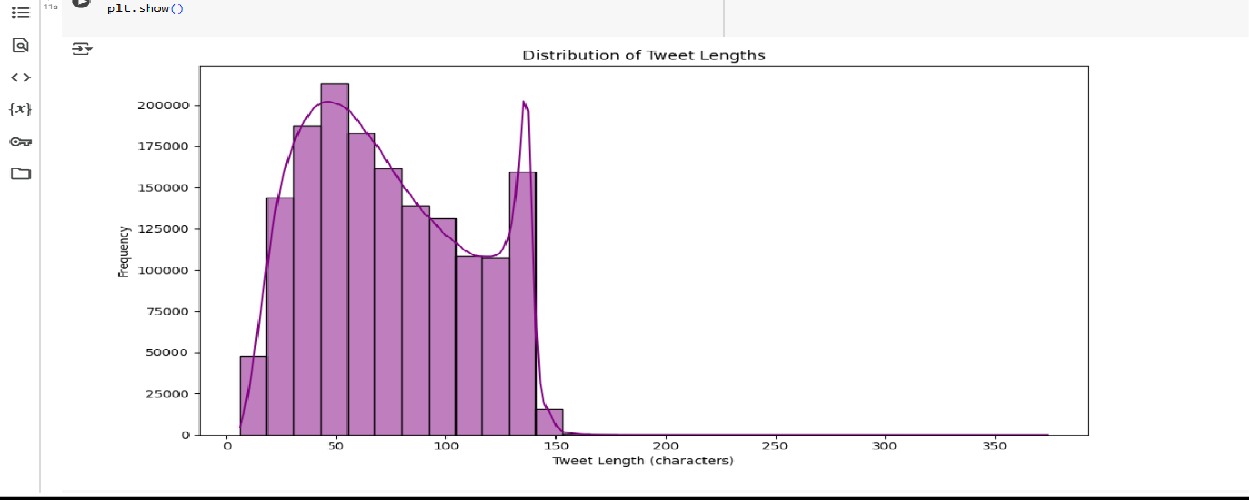
****

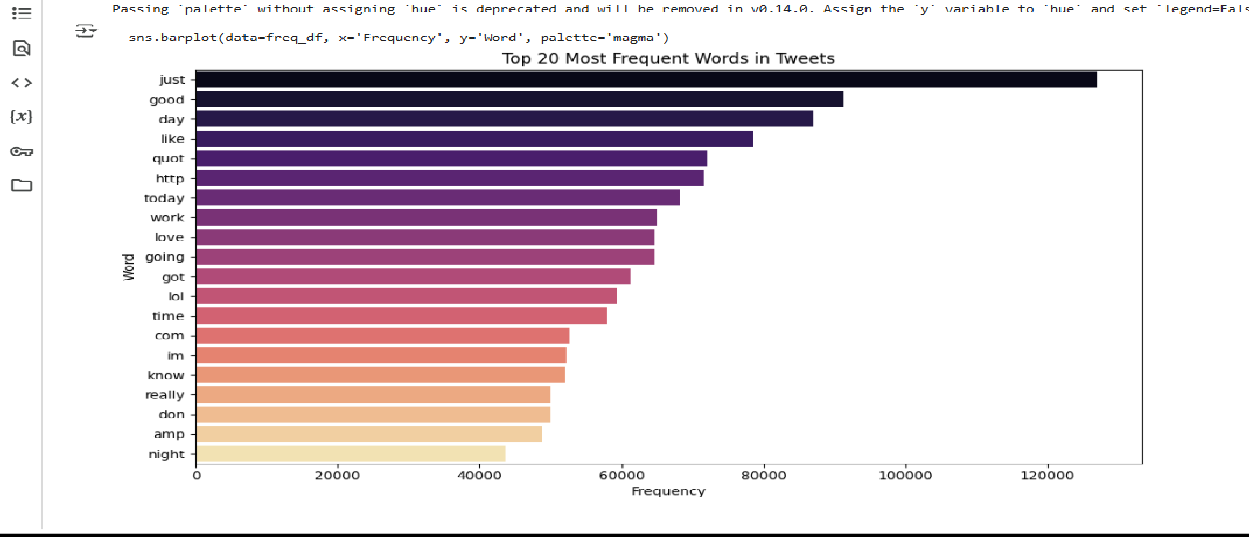
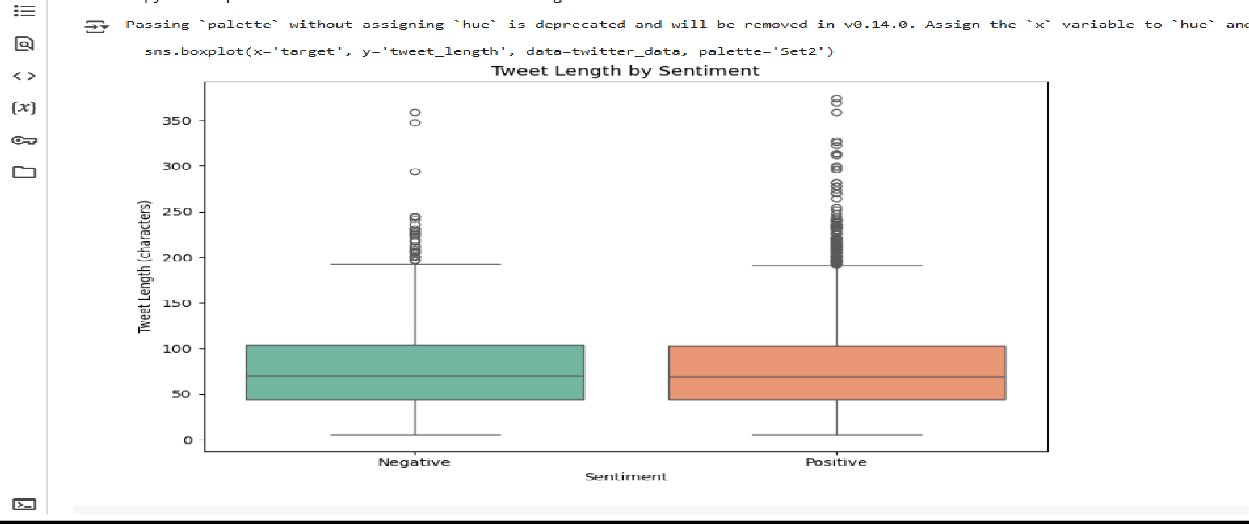
1. **Word Cloud-Most Common Words In Tweets:**

****

1. **Top 10 Most Frequent Words Based on TF-IDF (Bar-Plot)**

****



****

#### SAMPLE CODE

# install kaggle library

! pip install kaggle

# configuring the path of kaggle.json file

!mkdir -p ~/.kaggle

!cp kaggle.json ~/.kaggle/

!chmod 600 ~/.kaggle/kaggle.json

# API to fix the dataset from kaggle

!kaggle datasets download -d kazanova/sentiment140

# extracting the compressed dataset from zipfile import ZipFile

dataset = '/content/sentiment140.zip' with ZipFile(dataset,'r') as zip: zip.extractall()

print('The dataset is extracted')

#Importing Dependencies import numpy as np import pandas as pd import re

from nltk.corpus import stopwords

from nltk.stem.porter import PorterStemmer

from sklearn.feature\_extraction.text import TfidfVectorizer from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LogisticRegression from sklearn.metrics import accuracy\_score

import nltk nltk.download('stopwords')

#printing the stopwords in english print(stopwords.words('english'))

# loading the data from csv file to pandas dataframe twitter\_data =

pd.read\_csv('/content/training.1600000.processed.noem oticon.csv',encoding = 'ISO-8859-1')

# checking the number of rows and columns twitter\_data.shape

#print firsr 5 rows of data frame twitter\_data.head()

# naming the columns and reading the data again

column\_name = ['target','id','data','flag','user','text'] twitter\_data =

pd.read\_csv('/content/training.1600000.processed.noem oticon.csv',names=column\_name,encoding = 'ISO- 8859-1')

#print firsr 5 rows of data frame twitter\_data.head()

#counting the number of missing values in dataset twitter\_data.isnull().sum()

# checking the distrubution of tagert column twitter\_data['target'].value\_counts()

# Visualizing Sentiment Distribution (0: Negative, 1: Positive) plt.figure(figsize=(6, 4))

sns.countplot(data=twitter\_data, x='target', palette='Set1') plt.title('Sentiment Distribution (0: Negative, 1: Positive)') plt.xlabel('Sentiment')

plt.ylabel('Count')

plt.xticks([0, 1], ['Negative', 'Positive']) plt.show()

# Combine all the text data into one string all\_text = ' '.join(twitter\_data['text'])

# Generate the word cloud

wordcloud = WordCloud(width=800, height=400, background\_color='white').generate(all\_text)

# Plot the word cloud plt.figure(figsize=(10, 6))

plt.imshow(wordcloud, interpolation='bilinear') plt.axis('off')

plt.title('Most Common Words in Tweets') plt.show()

# Vectorizing the text data using TF-IDF vectorizer = TfidfVectorizer(max\_features=20) X = vectorizer.fit\_transform(twitter\_data['text'])

# Get the feature names (words)

features = vectorizer.get\_feature\_names\_out()

# Convert the TF-IDF matrix to a DataFrame

tfidf\_df = pd.DataFrame(X.toarray(), columns=features)

# Calculate the sum of the TF-IDF values for each word word\_freq = tfidf\_df.sum(axis=0).sort\_values(ascending=False)

# Plotting the most frequent terms based on TF-IDF plt.figure(figsize=(10, 6)) word\_freq.head(10).plot(kind='bar', color='skyblue') plt.title('Top 10 Most Frequent Words Based on TF-IDF') plt.xlabel('Words')

plt.ylabel('TF-IDF Score') plt.xticks(rotation=45) plt.show()

#### CONCLUSION

The Twitter Sentiment Analysis project illustrates the effective application of Data Science and Big Data Analytics (DSBDA) techniques to derive insights from unstructured social media data. By using Natural Language Processing (NLP) and machine learning algorithms such as Logistic Regression, Naive Bayes, and Support Vector Machines, we successfully built a model capable of classifying tweets into positive and negative sentiments. The project covered the entire data science pipeline — including data collection, cleaning, preprocessing, feature extraction using TF-IDF, model training, and performance evaluation. These steps emphasized the importance of handling real-time, large-scale data and applying analytics to extract patterns and trends that are otherwise difficult to detect manually.

Beyond technical implementation, the project showcased the real-world relevance of sentiment analysis in areas like marketing, customer feedback, brand monitoring, and political opinion tracking. With further enhancements, such as real-time API integration and the use of deep learning for better contextual understanding, the system can be scaled to handle even larger datasets and more complex sentiment variations. Overall, the project reflects the core values of DSBDA by demonstrating how data-driven methods can transform raw social media content into valuable, actionable insights.

**REFERENCES**

### Online References

1. [**www.nltk.org**](http://www.nltk.org/)

Natural Language Toolkit (NLTK). [Online]. Available at: [https://www.nltk.org](https://www.nltk.org/) (Accessed: 13 April 2025)

1. [**www.kaggle.com**](http://www.kaggle.com/)

Kaggle - Sentiment140 Dataset. [Online]. Available at: <https://www.kaggle.com/datasets/kazanova/sentiment140> (Accessed: 13 April 2025)

1. [**www.scikit-learn.org**](http://www.scikit-learn.org/)

Scikit-learn Documentation. [Online]. Available at: [https://scikit-learn.org](https://scikit-learn.org/) (Accessed: 13 April 2025)

### Books References

###### Machine Learning Yearning

By – Andrew Ng

###### Python for Data Analysis

By – Wes McKinney

###### Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow

By – Aurélien Géron