A

PROJECT REPORT

ON

"Twitter (X) Sentiment Analysis "

BY

Diksha Dipak Gole

Submitted in partial fulfilment of

BSc in Data Science

Under

Savitribai Phule Pune University

For the academic year

2025-2026

UNDER THE GUIDANCE OF

Mr. Ravikiran Pawar

Department of Technology,

Savitribai Phule Pune University,

Ganeshkhind, Pune 411007

CERTIFICATE

This is to certify that

Diksha Dipak Gole

Have successfully completed their project on

"Twitter (X) Sentimental Analysis"

In partial fulfilment of

BSc in Data Science

Under

Department of Technology,

Savitribai Phule Pune University,

For the academic year 2025-2026

Dr. Manisha Bharti
(Course Coordinator)

Dr.Aditya Abhyankar
(H.O.D)

Mr. Ravikiran Pawar
(Project Guide)

Signed By
(External Examiner)

DECLARATION

I undersigned **Diksha Gole** the student of BSc Data Science , Department of Technology, Savitribai Phule Pune University. Declare that the research project titled **"Twitter (X) Sentiment Analysis"** is a result of my own work and my indebtedness to other work publications, references, if any have been duly acknowledged. If I found guilty of copying any other report or published information and showing as our original work, we understand that we shall be liable and punishable by Institute or University, which may include failure in examination, repeat study and re-submission of the report or any other punishment that institute or University may decide.

Diksha Gole

BSC23DS46

Signature:

ACKNOWLEDGEMENT

We would like to extend our sincere appreciation to the each and every individual and the Department of Technology who played pivotal roles in the successful completion of this project. First and foremost, we are deeply grateful to our project guides **Mr. Ravikiran Pawar** for their exceptional guidance and mentorship, which significantly enriched the project's depth and direction. We are also very thankful for the timely guidance and motivation of **Dr Manisha Bharti ma'am**. To our peers, collaborators, and the open-source community, thank you for your contributions, insightful discussions, and assistance in fostering a collaborative research environment. We appreciate the efforts of all those involved in data collection and sharing, as well as the reviewers and anyone is engaged with the project, offering valuable feedback and insights.

Finally, our thanks to the readers and tribes of Maharashtra who will benefit from the findings presented herein. Your collective support, expertise, and encouragement were indispensable on this journey. We are also thankful to the HOD **Dr. Aditya Abhyankar** sir for his guidance and motivation for the project. Last but not the least we would thank each and everyone directly or indirectly involved in the project.

INDEX

Twitter (X) Sentiment Analysis

CHAPTER 1 : INTRODUCTION

1.1 Overview

Sentiment analysis is one of the most widely used Natural Language Processing (NLP) techniques used to identify the emotional tone behind written text. In the digital era, social media platforms—especially Twitter (now X)—have become powerful sources of public expression where millions of users share opinions every minute. Organizations, businesses, and governments increasingly depend on sentiment analysis systems to understand trends, monitor public mood, detect incidents, and make data-driven decisions.

This project focuses on building a **Twitter Sentiment Analysis System** using a modern, hybrid approach combining **Sentence-BERT (SBERT)** embeddings and a **Light Gradient Boosting Model (LightGBM)** classifier. Unlike traditional machine learning techniques that rely solely on TF-IDF or Bag-of-Words, SBERT generates deep semantic representations of text, making the model more robust and context-aware. When paired with LightGBM—a highly efficient gradient boosting algorithm—the system achieves strong performance even with noisy social media data.

The final output is a fully functional model that accepts a tweet and predicts whether the sentiment is **Positive**, **Negative**, or **Neutral**. The model is optimized for academic use, lightweight deployment, and reproducibility.

The full project, including the model and application, is hosted on GitHub:

- https://github.com/5Diksha/Project_Minor

1.2 Motive

The motivation behind this project stems from the increasing global reliance on digital communication. Twitter acts as a real-time reflection of public opinion. Organizations frequently analyze Twitter discussions to:

- gauge customer satisfaction,
- monitor brand reputation,
- detect sentiment patterns across trending topics,
- identify negative feedback early,
- improve public relations.

Manual analysis of thousands of tweets is time-consuming, inconsistent, and error-prone. Therefore, a reliable automated sentiment analysis system is necessary.

Additionally, this project aims to explore more advanced NLP models such as SBERT, which provide significantly richer semantic information than classical models. The goal is not only to build a sentiment classifier but also to understand the practical pipeline of modern NLP systems used in industry today.

1.3 Problem Statement

Millions of tweets are generated daily, containing slang, abbreviations, emojis, hashtags, and informal writing styles. This irregularity makes sentiment classification challenging. Traditional ML models struggle to capture contextual meaning, leading to lower accuracy and generalization issues.

Thus, the problem addressed in this project is:

"To design and develop a robust sentiment analysis model for Twitter data using SBERT embeddings and LightGBM, capable of accurately predicting positive, neutral, and negative sentiments."

The solution must be reliable, scalable, and capable of handling noisy real-world text data.

1.4 Objectives

The major objectives of this project are:

1. To collect and preprocess a high-quality Twitter sentiment dataset.
2. To clean and normalize tweet text for improved model performance.
3. To generate contextual embeddings using Sentence-BERT.
4. To train a LightGBM classifier with optimized hyperparameters.
5. To evaluate the model using standard performance metrics.
6. To develop a simple user interface (Streamlit) for real-time predictions.
7. To provide a clear and structured report showcasing technical understanding and implementation.

CHAPTER 2 :DATASET

2.1 Data Source

The dataset used in this project is the **TweetEval Sentiment Dataset**, a well-known and academically recognized benchmark for Twitter sentiment classification. It is provided by CardiffNLP and hosted on HuggingFace. The dataset includes **train**, **validation**, and **test** splits and contains tweets labeled as:

- **0 → Negative**
- **1 → Neutral**
- **2 → Positive**

This dataset is widely used in NLP research due to its authenticity, real-world tweet content, and challenging nature involving abbreviations, sarcasm, and informal language.

2.2 Dataset Examples

Sample tweets from the dataset include:

Tweet	Label
“I hate waking up so early 😩”	Negative
“Going to college now.”	Neutral
“This is the best news ever!”	Positive

These examples reflect the typical variety of emotional expressions on Twitter.

2.3 Dataset Preparation

Before training the model, the dataset was thoroughly cleaned and standardized:

Steps involved:

1. **Removed URLs:**
   Links do not contribute to sentiment.
2. **Removed user mentions (@username):**
   They are irrelevant to the emotional tone.
3. **Removed extra whitespace and punctuation.**
4. **Handled contractions and normalised text:**
   Example: *"don't" → "do not"*
5. **Generated VADER (Valence Aware Dictionary) scores:**
   These scores act as additional sentiment features.
6. **Encoded labels into numeric format (0, 1, 2).**

   The cleaned dataset is stored in the project folder as:

- clean_train.csv
- clean_valid.csv
- clean_test.csv

The final dataset sizes:

Split	Samples
Train	45,562
Validation	2,000
Test	12,278

CHAPTER 3 : METHODOLOGY

This chapter explains the complete sentiment analysis pipeline used in the project. The approach combines traditional preprocessing with modern deep learning embeddings.

3.1 Data Preprocessing

Twitter text is noisy. Preprocessing ensures clean, consistent inputs for SBERT and LightGBM.

Steps performed:

- Lowercasing the text
- Removing HTML tags
- Removing hashtags and usernames
- Filtering special symbols
- Lemmatization using spaCy
- Optional removal of stopwords
- Emoji normalization

A well-preprocessed dataset improves embedding quality and reduces noise during training.

3.2 VADER Feature Extraction

VADER sentiment analyzer is a lexicon and rule-based sentiment scoring system. It is particularly effective for short social media text.

The VADER "compound" score ranges from **–1 (very negative)** to **+1 (very positive)**.

Why include VADER?
Because VADER captures surface-level sentiment cues such as emojis and slang, which complements SBERT's deeper semantic meaning. Together they provide a more holistic representation.

3.3 SBERT Embedding Extraction

Sentence-BERT (SBERT) is a modification of BERT designed to generate fixed-length vectors for sentences. Unlike classical BERT outputs, SBERT is optimized for sentence similarity and semantic tasks.

Why SBERT?

- Captures contextual meaning
- Provides 384-dimensional embedding vectors
- Efficient for downstream ML models
- Works excellently for short text like tweets

SBERT is loaded using the "all-MiniLM-L6-v2" model, which is lightweight and fast while delivering strong performance.

3.4 LightGBM Training Pipeline

LightGBM is a gradient boosting framework that:

- trains extremely fast,
- handles high-dimensional data,
- naturally supports multiclass classification,
- provides excellent generalization.

Training Process:

1. SBERT vectors generated for every tweet
2. VADER score appended as an additional feature
3. LightGBM receives **385 total features**
4. Dataset split into train/validation/test
5. Model trained with early stopping
6. Hyperparameters tuned for balanced performance

The final model achieved:

- **Validation Accuracy:** 67.65%
- **Test Accuracy:** 65.66%

Given the challenge of tweet language, this performance is aligned with academic research results for non-transformer classifier models.

3.5 Workflow Diagram

(Insert diagram in Word)
The workflow should visually show:

1. Raw tweet →
2. Preprocessing →
3. SBERT Embedding →
4. VADER Score →
5. Feature Vector →
6. LightGBM Classifier →
7. Output: Sentiment Label
8.

CHAPTER 4 : IMPLEMENTATION

4.1 Tools & Technologies Used

Tool / Library	Purpose
Python	Programming Language
Sentence-BERT	Embedding generation
LightGBM	Classifier
NLTK	Preprocessing, VADER
Pandas	Data handling
Streamlit	User Interface
HuggingFace Datasets	Dataset fetching
scikit-learn	Metrics and support utilities

4.2 Model Training

The model training process includes:

- Loading the cleaned dataset
- Computing SBERT embeddings
- Appending VADER scores
- Training LightGBM with optimized parameters
- Saving model and embeddings to /models directory

Training took ~7 minutes on CPU due to efficient batching.

4.3 Streamlit Application

A minimal Streamlit application was implemented:

- Accepts user text input
- Filters/cleans the text
- Generates SBERT embedding
- Adds VADER score
- Runs LightGBM prediction
- Displays label + probability

This allows users to test their sentences instantly.

4.4 Final Model Architecture

(**Insert diagram in your report**)

The final model block diagram:

- **Input:** User text
- **Processing:** Cleaning → VADER → SBERT
- **Model:** LightGBM
- **Output:** Sentiment Class

CHAPTER 5 : EVALUATION

5.1 Metrics Used

To evaluate the classifier, the following metrics were calculated:

- Accuracy
- Precision
- Recall
- F1-score
- Weighted averages
- Confusion matrix

These give a comprehensive view of model performance.

5.2 Results Analysis

Metric	Validation	Test
Accuracy	67.65%	65.66%
Positive F1	0.72	0.62
Neutral F1	0.65	0.66
Negative F1	0.60	0.66

Key Observations:

- The model performs best for Positive and Neutral tweets.
- Negative tweets are harder due to sarcasm and ambiguous tone.
- SBERT improves semantic understanding significantly over traditional models.
- LightGBM handles high-dimensional embeddings efficiently.

5.3 Limitations

1. Twitter language is often inconsistent and sarcastic.
2. Emojis and slang may require deeper contextual models.
3. Classical boosting models cannot match end-to-end fine-tuned transformers.
4. Dataset contains writing inconsistencies that affect model performance.

5.4 Future Scope

1. Transformer fine-tuning (RoBERTa, BERTweet).
2. Larger datasets and multilingual support.
3. Deploying full backend + Android app.
4. Real-time tweet monitoring pipeline.
5. Adding Explainable AI (SHAP) to interpret predictions.

5.5 Ethical Considerations

- Sentiment models should avoid biases across language groups.
- User privacy must be respected when analyzing personal tweets.
- Models must be transparent about limitations.

CONCLUSION :

This project successfully developed a sentiment classification system using SBERT embeddings and LightGBM. The system achieves strong performance on real Twitter data and demonstrates the effectiveness of combining deep semantic embeddings with gradient boosting classifiers. The final model is fast, lightweight, and suitable for academic use and real-world applications. The work provides a solid foundation for future enhancements such as transformer fine-tuning and full-scale deployment.

REFERENCES :

1) Reimers, N., & Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks.* Proceedings of EMNLP-IJCNLP.
2) Barbieri, F., Camacho-Collados, J., Espinosa-Anke, L., & Neves, L. (2020). *TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification.* Proceedings of Findings of EMNLP.
3) Chen, T., & Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System.* Proceedings of KDD.
4) Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., … Liu, T. Y. (2017). *LightGBM: A Highly Efficient Gradient Boosting Decision Tree.* Advances in Neural Information Processing Systems (NeurIPS).
5) Hutto, C., & Gilbert, E. (2014). *VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text.* Proceedings of ICWSM.
6) Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* NAACL-HLT.
7) Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., … Rush, A. (2020). *Transformers: State-of-the-art Natural Language Processing.* Proceedings of EMNLP. HuggingFace.
8) Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space.* arXiv preprint arXiv:1301.3781.
9) Pennington, J., Socher, R., & Manning, C. D. (2014). *GloVe: Global Vectors for Word Representation.* Proceedings of EMNLP.
10) Pang, B., & Lee, L. (2008). *Opinion Mining and Sentiment Analysis.* Foundations and Trends in Information Retrieval.
11) Liu, B. (2012). *Sentiment Analysis and Opinion Mining.* Morgan & Claypool Publishers.
12) Mohammad, S. M., & Turney, P. D. (2013). *Crowdsourcing a Word–Emotion Association Lexicon.* Computational Intelligence.
13) Twitter Developers. (2023). *Twitter API Documentation.* Available at developer.twitter.com
14) HuggingFace Datasets. (2022). *TweetEval Dataset Documentation.* Available at https://huggingface.co/datasets
15) Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research.