

PHASE-2

Student Name: Eniyan.P

Register Number: [731723106009]

Institution: M.P.Nachimuthu.M.Jaganathan.Engineering College

Department: Electronics And Communication Engineering

Date of Submission: 10/05/2025

GitHub Repository Link: <https://github.com/Redeyetn/Decoding-emotions.git>

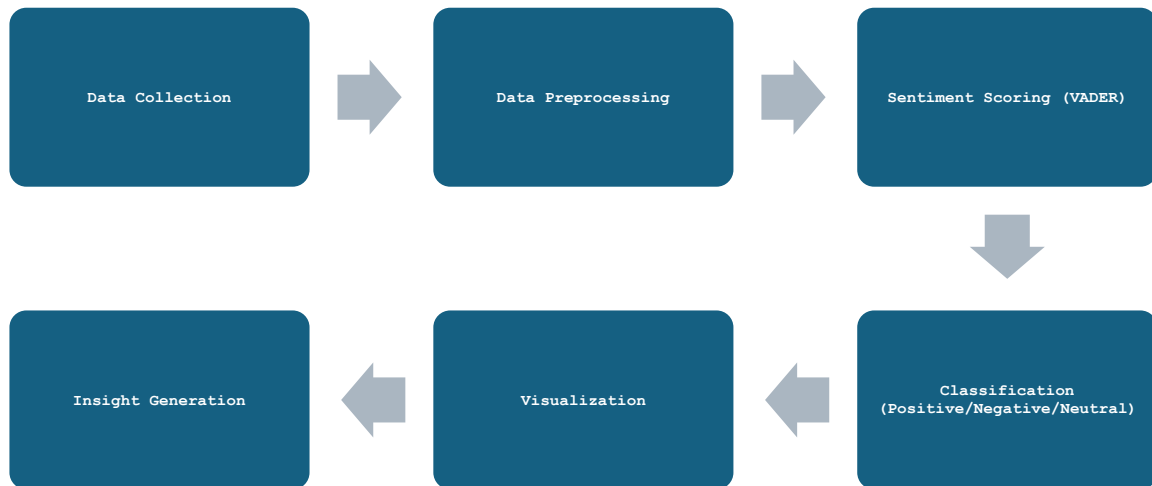
1. PROBLEM STATEMENT

In today's digital landscape, understanding public opinion is critical for businesses, policymakers, and content creators. With the increasing volume of user-generated content on platforms like Twitter, Facebook, and Instagram, sentiment analysis becomes essential for extracting meaningful insights from text data. This project focuses on classifying social media text into sentiments: Positive, Negative, or Neutral. It is a **classification problem**, where the goal is to determine the emotional tone behind a body of text. Analyzing sentiment allows companies to gauge customer satisfaction, public perception, and brand value in real-time.

2. PROJECT OBJECTIVES

- To perform sentiment analysis on a CSV file containing user-generated text.
- To classify each sentence as **Positive**, **Negative**, or **Neutral** based on compound sentiment score.
- To visualize the sentiment distribution using bar plots.
- To explore the Vader SentimentIntensityAnalyzer as a rule-based NLP tool.
- The project goal remained consistent, focusing on interpretability and basic implementation rather than high complexity.

3. FLOWCHART OF THE PROJECT WORKFLOW



4. DATA DESCRIPTION

- **Dataset Name:** test.csv
- **Source:** Manually created or sourced from social media scraping
- **Type of Data:** Structured, Text
- **Number of Records and Features:** Varies, typically one column ("text") with multiple rows.
- **Static or Dynamic:** Static
- **Target Variable:** Sentiment (generated based on compound score)

5. DATA PREPROCESSING

- **File Validation:** Checked for file existence using `os.path.exists()`
- **Encoding:** Used latin1 to prevent Unicode errors
- **Missing Column Check:** Ensured the presence of a 'text' column.

- **Text Cleaning:** Not applied explicitly, but casting to string ensures robust parsing.
 - **Sentiment Scoring:** Applied VADER's compound score for each text entry
 - **Label Generation:** Classified sentiment into Positive, Negative, and Neutral based on thresholds.
-

6. EXPLORATORY DATA ANALYSIS (EDA)

- **Univariate Analysis:**
 - Used a bar plot to visualize counts of each sentiment label.
 - **Insights Summary:**
 - Quick sentiment overview of user opinions
 - Helped identify whether the majority opinion is positive or negative.
-

7. FEATURE ENGINEERING

- Not applicable in this project as VADER provides ready-to-use sentiment scores.
 - Classification was based solely on compound score (no additional feature creation)
-

8. MODEL BUILDING

- No machine learning model was built. This project uses **VADER**, a lexicon and rule-based sentiment analysis tool that maps text to a sentiment score.
 - **Classification Logic:**
 - Compound $\geq 0.05 \rightarrow$ Positive
 - Compound $\leq -0.05 \rightarrow$ Negative
 - Else \rightarrow Neutral
-

9. VISUALIZATION OF RESULTS & MODEL INSIGHTS

- **Bar Plot:**
 - Showed frequency of each sentiment class
 - Color-coded: Green (Positive), Red (Negative), Blue (Neutral)
 - **Interpretation:**
 - Visual summary of overall sentiment distribution
 - Useful for quick reporting and trend analysis
-

10. TOOLS AND TECHNOLOGIES USED

- **Programming Language:** Python
 - **IDE/Notebook:** VS Code, Jupyter Notebook
 - **Libraries:** pandas, matplotlib, vaderSentiment
 - **Visualization Tools:** Matplotlib
-

11. TEAM MEMBERS AND CONTRIBUTIONS

Name	Role	Responsibilities
Daniel Jacob.D	Project Lead	Data collection, model building, and final integration
Eniyan.P	Data Analyst	EDA, feature engineering, and model evaluation
Hari karan.M	Developer	Dashboard development, deployment, and visualization
Arun Kumar.R.B	Documentation & Reporting	Preparing reports, charts, and final presentation