Phase-2 Submission

Student Name: Hariharan R

Register Number: 712523205024

Institution: PPG Institute of Technology

Department: Information Technology

Date of Submission: 09/05/2025

Github Repository Link:
https://github.com/Hari21haran/Nm_hariharan_ds

Project Title: Decoding Emotions Through Sentiment Analysis of Social Media Conversations

1. Problem Statement

Social media is overflowing with emotional expressions, yet most sentiment analysis tools reduce these to basic categories like positive or negative. This oversimplification ignores deeper emotions such as anger, joy, fear, and sadness—limiting meaningful insight. This project aims to build an advanced model that accurately identifies a wide range of emotions in real time, enabling organizations to understand public sentiment more deeply and respond with greater relevance.

It will bridge the gap between raw data and emotional intelligence, offering valuable insights for businesses, policymakers, and mental health initiatives. By decoding emotion-rich content, the system will support more empathetic and data-driven decisions.

2. Project Objectives:

- *Build a model for classifying social media text into multiple emotions (e.g., joy, anger, fear).*
- *Achieve high accuracy and F1-score in detecting diverse emotions.*
- *Use advanced NLP techniques like LSTM and BERT for emotion classification.*
- *Preprocess social media text to handle informal, noisy language.*
- *Ensure model interpretability for better decision-making.*
- *Enable real-time emotional trend tracking.*
- *Support practical use cases in business, marketing, and crisis management.*

3. Flowchart of the Project Workflow

4. Data Description

- ***Dataset name:*** *Twitter Emotion Dataset*
- ***Source:*** *Social Media Emotion Detection*
- ***Type of data:*** *Unstructured Text Data from Social Media Posts*
- ***Records and Features:*** *Over 800 text records with emotion labels and metadata*
- ***Static or dynamic dataset:*** *Both static dataset and dynamic real-time data*
- ***Target variable:*** *Emotion category (e.g., happy, sad, angry, excited)*
- ***Attributes Covered:*** *Text content, emojis, sentiment labels, hashtags, timestamps*
- ***Dataset Link:*** *https://www.kaggle.com/datasets/markmedhat/twitter*

5. Data Preprocessing

- *Removed records with empty text fields or missing labels.*
- *Deleted exact duplicate posts to prevent bias.*
- *Filtered out unusually short or long posts to maintain data quality.*
- *Converted all text to lowercase and formatted timestamps properly.*
- *Cleaned text by removing URLs, mentions, special characters, and converting emojis to words.*
- *Encoded emotion categories into numerical labels for model input.*
- *Applied TF-IDF for traditional models and BERT tokenization for deep learning models.*

6. Exploratory Data Analysis (EDA)

- ***Univariate Analysis:***
    - *Emotion countplots show imbalance, with joy and neutral being most frequent.*
    - *Most tweets are under 30 words, as shown in tweet length histograms.*
    - *Boxplots reveal longer tweets are often associated with anger and sadness.*
- ***Bivariate/Multivariate Analysis:***
    - *Correlation matrix shows slight relationships between polarity and subjectivity.*

    - *Pairplots indicate clustering among similar emotions like sadness and fear.*
    - *Scatterplots show high subjectivity is often tied to strong emotions like joy and anger.*

- ***Insights Summary:***

    - *The dataset has imbalanced emotion classes that may affect model accuracy.*
    - *Features like polarity, subjectivity, tweet length, and emoji count are likely important.*
    - *Emotionally intense tweets are typically longer and more expressive.*

7. Feature Engineering

- *Extract meaningful features like word count, emoji presence, punctuation usage, hashtags, and mentions to capture emotional expressions common in social media posts*
- *Use sentiment lexicons (e.g., VADER, NRC) to assign emotion or polarity scores, adding domain-driven features that highlight emotional intensity.*

- *Derive features such as hour, day, or weekday from timestamps (if available) to analyze how sentiment changes over time or during specific events.*

- *Apply techniques like TF-IDF, Word2Vec, or BERT embeddings to convert text into numerical form while preserving context and semantics.*

- *Use methods like correlation analysis, PCA, or feature importance scores to reduce redundancy and improve model efficiency and performance.*

8. Model Building

- ***Model Selection:***
  Implemented and compared two machine learning models: Logistic Regression and Random Forest, selected for their proven performance in text classification tasks and ability to handle high-dimensional data.

- ***Justification of Models:***
  Logistic Regression offers simplicity and interpretability, ideal as a baseline. Random Forest provides robustness, handles non-linearity well, and mitigates overfitting through ensemble learning.

- ***Data Splitting:***
  Split the dataset into training and testing sets using stratified sampling to ensure the class distribution is preserved, especially important for imbalanced emotion classes.

- ***Model Training:***
  Trained both models on the training set, using processed features such as TF-IDF vectors and sentiment scores derived from lexicons.

- ***Performance Evaluation:***
  Evaluated model performance using classification metrics: accuracy, precision, recall, and F1-score, with special focus on F1-score to handle class imbalance effectively.

9. Visualization of Results & Model Insights

- ***Confusion Matrix:***
  Displayed confusion matrices for each model to visualize how well emotions were classified. These helped identify which emotions were frequently misclassified and guided model improvement.

- ***ROC Curve:***
  Plotted ROC curves for multi-class classification (using one-vs-rest approach) to assess the models' ability to distinguish between emotion categories. AUC scores were compared to evaluate overall performance.

- ***Feature Importance Plot:***
  For Random Forest, visualized the most influential features contributing to emotion prediction, such as specific keywords, emojis, or punctuation patterns, providing insight into model decision-making.

- ***Model Comparison Charts:***
  Used bar charts and line plots to compare performance metrics (accuracy, precision, recall, F1-score) across different models, making it easier to interpret which model performed best.

- ***Interpretation of Results:***
  Explained what each visualization reveals about model behaviour and data patterns—for instance, which features strongly indicate joy or anger, and how well the model captures subtleties in emotional tone.

10. Tools and Technologies Used

- ***Programming & Environment:*** *Python, using Google Colab for development and execution.*

- ***Key Libraries Used:*** *pandas, numpy, scikit-learn, matplotlib, seaborn, and nltk for data processing, modeling, and visualization.*

- ***Visualization & Deployment Tools:*** *Plotly for interactive plots; optional deployment with Streamlit for showcasing results*

11. Team Members and Contributions

Team Member	*Role*	*Responsibility*
Danish P	*Model Development*	*Responsible for building machine learning models and guiding model selection.*
Atsaya D	*Data Cleaning*	*Handles preprocessing tasks, including managing missing values and improving data quality.*
Karthika V	*Feature Engineering*	*Focuses on creating and refining features to enhance model performance.*
Abishek S	*Exploratory Data Analysis*	*Conducts data exploration to identify patterns and extract key insights.*
Hariharan R	*Documentation Support*	*Maintains clear and organized documentation throughout the project.*