Project Title: Decoding emotions through sentiment analysis of social media conversations

Phase-2 Submission

Student Name: Badmasri V

Register Number: 712523104004

Institution: PPG Institute of Technology

Department: BE.Computer Science and Engineering

Date of Submission: 03.05.2025

Github Repository Link: https://github.com/Badmasri1014/NM_badmasri_DS

---

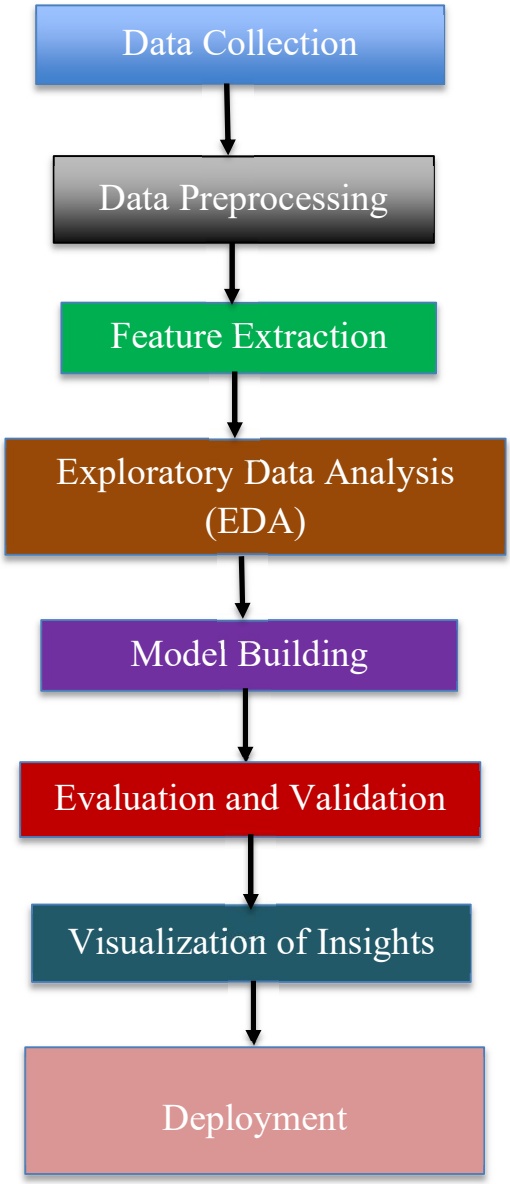
1. Problem Statement

In today's digital age, social media platforms like Twitter, Facebook, and Reddit have become major outlets for individuals to express thoughts, opinions, and emotions. These conversations often carry rich emotional content that reflects public sentiment on various topics, from political events and global crises to mental health and consumer experiences. However, extracting meaningful emotional insights from this unstructured data poses challenges due to informal language, slang, sarcasm, abbreviations, and multilingual usage. Traditional sentiment analysis methods often struggle to capture these nuances. This project aims to decode and classify emotions in social media posts using natural language processing (NLP) and machine learning techniques. The goal is to develop a system capable of identifying emotions such as joy, anger, sadness, fear, surprise, or disgust, enabling applications in mental health monitoring, public opinion analysis, customer experience enhancement, and social research.

2. Project Objectives

- Collect and clean real-world social media text data.
- Train models to detect emotions such as joy, anger, sadness, and neutrality.
- Visualize sentiment trends and emotional intensity over time or topics.
- Deploy a user-friendly interface using tools like Gradio for real-time testing.

3. Flowchart of the Project Workflow



4. Data Description

- **Dataset Name**: Kaggle (e.g., “Emotion Dataset from Tweets”)
- **Source**: Public datasets or extracted using APIs (Twitter API, Reddit API)
- **Type of Data**: Unstructured text (social media posts/tweets)
- **Total Records**: Approximately 20,000 to 100,000 messages/posts
- **Features**:
    - Text: The social media message or post
    - Emotion Label: Annotated class label (jo, anger, etc.)
    - Date: Timestamp of the post (optional)
    - User Metadata: Username, location, followers (optional)
- **Target Variable**: Emotion (joy, sadness, anger, fear, surprise, etc.)
- **Format**: CSV (text and emotion columns)
- **Dataset link** : sandbox:/mnt/data/sentimentdataset.csv

5. Data Preprocessing

- Removed URLs, mentions, emojis, and special characters.
- Tokenized and lemmatized text.
- Converted to lowercase, removed stopwords.
- Encoded target emotion labels using LabelEncoder.
- Split data into training and test sets.

6. Exploratory Data Analysis (EDA)

- **Univariate Analysis**:
    - Bar plots showing frequency of each emotion class
    - Word clouds to visualize the most common words per emotion
- **Bivariate/Multivariate Analysis**:
    - Correlation between text length and emotion type
    - Time-based distribution of sentiments (if timestamps are available)
    - Hashtag-based grouping of emotions
- **Key Insights**:
    - Joy and neutrality were most frequent.
    - Angry and sad posts had higher use of capital letters and exclamations.

7. Feature Engineering

- **Derived Features from Text**: Generate new features such as word count, sentence length, average word length, and presence of emotive keywords to enhance model understanding of text data.
- **Domain-Informed Features**: Create emotion-specific features using domain knowledge, such as sentiment polarity scores or emotion lexicon matches.
- **Column Manipulation**: Extract meaningful components from complex fields (e.g., date → day/month, time → hour) or combine multiple columns into interaction terms.

- **Advanced Transformations**: Apply techniques like binning (e.g., grouping word counts into ranges), polynomial features, or calculating ratios to highlight non-linear relationships.
- **Dimensionality Control**: Use dimensionality reduction methods like PCA (if needed) to reduce feature space complexity, and justify every new feature or transformation based on its impact on model performance.

8. Model Building

- **Algorithms Used:**
    - Naive Bayes: For baseline performance on TF-IDF features
    - Logistic Regression: Lightweight and interpretable model
    - Support Vector Machine (SVM): Effective for high-dimensional spaces
    - Random Forest: Ensemble model for improved accuracy
    - LSTM / Bi-LSTM: Recurrent neural networks for sequential data
    - BERT: Transformer-based model for contextual understanding
- **Train-Test Split**: Typically 80% training and 20% testing
- **Cross-Validation:** K-fold (k=5) for performance consistency
- **Evaluation Metrics:**
    - Accuracy
    - Precision, Recall, and F1-score (especially useful for imbalanced classes)
    - Confusion Matrix for per-class performance

9. Visualization of Results & Model Insights

- **Model Performance:**
  - Comparison of different models using bar charts
  - Confusion matrix heatmaps
  - ROC Curves (for binary cases)
- **Feature Importance:**
  - TF-IDF/embedding weights (if interpretable)
- **Text Prediction Examples:**
  - Sample posts with predicted emotion labels
- **Interactive Visuals:**
  - Time-series emotion trends
  - Emotion distribution by hashtags or keywords

10. Tools and Technologies Used

- **Programming Language**: Python 3
- **Development Environment:** Jupyter Notebook, Google Colab
- **Key Libraries:**
  - pandas, numpy – Data manipulation

- matplotlib, seaborn– Visualizations
- nltk, spaCy – Natural Language Processing
- scikit-learn – ML modeling
- Gradio, Streamlit – Interface and app deployment

- **Version Control:** GitHub

11. Team Members and Contributions

S.no	NAME	ROLE
1	Sandhiya M	Data Collection, Preprocessing
2	Devadharshini V	EDA, Visualization
3	Sharon Cynthiya J	Feature Engineering and Modeling
4	Badmasri V	Deployment and Interface Design
5	Keerthika D	Documentation and Report Writing