

## Phase-2 Submission

**Student Name:** Ravi kumaar B

**Register Number:** AU410723104065

**Institution:** Dhanalakshmi College of Engineering

**Department:** computer science engineering

**Date of Submission:** 07-May-25

**GitHub Repository link:**

[https://github.com/ravikumaarb/Nm\\_ravikumaards/blob/main/Phase-1%20nm%20ravikumaar%5B1%5D.pdf](https://github.com/ravikumaarb/Nm_ravikumaards/blob/main/Phase-1%20nm%20ravikumaar%5B1%5D.pdf)

---

# Project Title: Decoding Emotions Through Sentiment Analysis of Social Media Conversation

## 1. Problem Statement

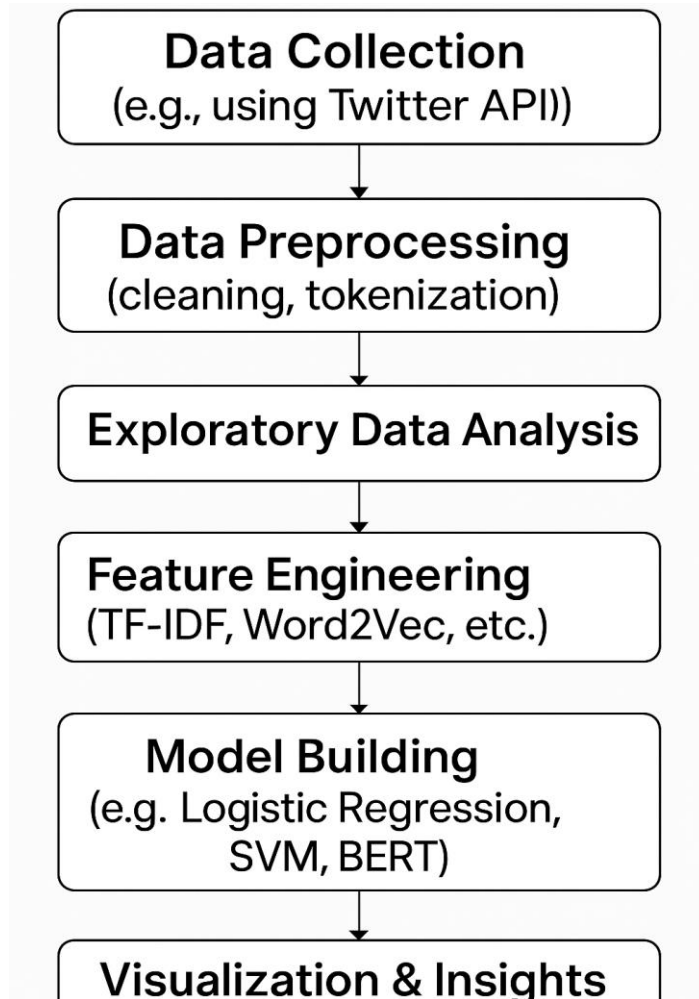
*With the exponential growth of social media platforms like Twitter, Facebook, and Instagram, people now express their thoughts, opinions, and emotions online more than ever. These platforms have become a rich repository of user-generated content, which provides unique opportunities to understand public sentiment across various domains such as politics, mental health, marketing, and customer service.*

## 2. Project Objectives

1. *To collect a diverse dataset of social media text data such as tweets or Facebook comments.*
2. *To preprocess the textual data by cleaning, normalizing, and tokenizing the content.*
3. *To build models capable of performing sentiment classification with high accuracy and robustness.*
4. *To visualize emotional trends across time, location, or topics to better understand public sentiment.*
5. *To address challenges such as sarcasm detection, handling class imbalance, and dealing with multilingual content*

## 3. Flowchart of the Project Workflow

- *Data Collection: Using APIs like Twitter API or extracting data from public datasets on Kaggle.*
- *Data Preprocessing: Cleaning text by removing noise (URLs, hashtags, mentions), normalizing, and tokenizing.*
- *Exploratory Data Analysis: Identifying patterns in data, distribution of sentiments, and word usage.*
- *Feature Engineering: Creating numerical representations of text using techniques like TF-IDF, Word2Vec, or BERT embeddings.*
- *Model Building: Training machine learning models such as Logistic Regression, SVM, or deep learning models like LSTM and BERT.*
- *Model Evaluation: Assessing models using metrics such as accuracy, recall, F1-score, and confusion matrix.*
- *Visualization and Insights: Plotting sentiment trends, keyword importance, and evaluating emotional responses across different dimensions.*



## 4. Data Description

**Dataset Source:** Data is collected from the Twitter API, Reddit threads, or publicly available sentiment datasets on platforms like Kaggle.

**Data Type:** Unstructured text data containing user-generated content.

**Features:** Tweet/comment text, timestamps, user metadata (optional), sentiment labels.

**Target Variable:** Sentiment classification—commonly Positive, Negative, and Neutral.

**Nature of Data:** Can be static (archived datasets) or dynamic (real-time data from social APIs).

## 5. Data Preprocessing

**Text Cleaning:** Remove punctuation, numbers, stop words, URLs, hashtags, and emojis.

**Text Normalization:** Lowercasing, correcting spelling errors, and expanding contractions.

**Tokenization and Lemmatization:** Breaking text into individual words and reducing them to their root form.

**Handling Imbalanced Classes:** Apply oversampling (e.g., SMOTE) or under sampling techniques if sentiment categories are not evenly distributed.

## 6. Exploratory Data Analysis (EDA)

- Analyze most frequently occurring words across sentiment classes.
- Generate word clouds to visualize dominant words in positive, negative, and neutral texts.
- Assess the distribution of sentiments to understand biases in the data.
- Study trends over time or across hashtags to identify sentiment shifts.

## 7. Feature Engineering

**Text Vectorization:** Convert text into numerical format using TF-IDF, Bag of Words, or advanced embeddings like Word2Vec or GloVe.

**Contextual Embeddings:** Use pre-trained language models like BERT to capture contextual relationships in sentences.

**Linguistic Features:** Add parts of speech tags, sentiment lexicons, or n-gram frequency counts.

## 8. Model Building

*Choose appropriate models such as:*

- *Logistic Regression and SVM for baseline performance*
- *Random Forest for robustness and interpretability*
- *LSTM and BERT for deep learning-based performance*
- *Fine-tune models using cross-validation and hyperparameter tuning.*
- *Evaluate performance using metrics:*
  - *Accuracy*
  - *Precision, Recall, and F1-score*
  - *Confusion Matrix and ROC/AUC*

## 9. Visualization of Results and Model Insights

**Confusion Matrix:** *Visual representation of prediction performance.*

**ROC/AUC Curve:** *To analyze model performance at various classification thresholds.*

**Sentiment Over Time:** *Line plots to visualize how sentiment changes over time.*

**Keyword Contribution:** *Identify key terms that contribute most to each sentiment class.*

## 10. Tools and Technologies Used

**Programming Language:** *Python*

**NLP Libraries:** *NLTK, spaCy, TextBlob, HuggingFace Transformers*

**Data Handling:** *pandas, numpy*

**Modeling:** *scikit-learn, TensorFlow, Keras*

**Visualization:** *matplotlib, seaborn, plotly*

**IDEs:** *Google Colab, Jupyter Notebook*

**Version Control:** *GitHub for tracking project development*

## 11. Team Members and Contributions

<b><i>Name</i></b>	<b><i>Role</i></b>	<b><i>Responsibilities</i></b>
<i>Umesh</i>	<i>Presentation Designer</i>	<i>Designed and structured the project presentation for clear and effective delivery.</i>
<i>Srikanth</i>	<i>Content Researcher</i>	<i>Researched sentiment analysis techniques and contributed to writing project content.</i>
<i>Yashwanth</i>	<i>Content Researcher</i>	<i>Helped gather insights and develop detailed write-ups for each section of the project.</i>
<i>Ravikumaar</i>	<i>Technical Support &amp; Review</i>	<i>Assisted with reviewing the technical workflow, validating data processing steps, and ensuring overall quality control.</i>