Phase-1 Submission Template

Student Name: MADHUMITHA S

Register Number: 731123106030

Institution: GOVERNMENT COLLEGE OF ENGINEERING, ERODE

Department: ELECTRONICS AND COMMUNICATION ENGINEERING

Date of Submission: 02/05/2025

• **Problem Statement**

Social media is an emotional minefield—raw, chaotic, and laced with slang, sarcasm, and vibes AI still can't crack. We're building a smart engine to decode real-time emotional signals from posts, turning noise into nuanced insight.

• **Objectives of the Project**

1. **Emotion Detection:** Accurately identify and classify a wide range of beyond simple sentiment polarity.

2. **Contextual Understanding:** Handle sarcasm, slang, abbreviations, and emojis to interpret emotional tone in real-world, informal language.

3. **Real-time Analysis:** Build a pipeline that processes and analyzes live or recent social media data streams.

4. **Multi-Modal Integration (optional for stretch goals):** Incorporate

text, emojis, and even images or memes to enhance emotion detection accuracy.

5.Visualization & Insights: Create a dashboard or visual tool to present emotional trends, spikes, or anomalies over time.

6.Scalability & Speed: Ensure the system can handle high-volume data efficiently during live events or viral trends.

•Scope of the Project

This project aims to build a tool that can understand people's emotions—like happiness, anger, sadness, or fear—by analyzing what they post on social media (like tweets or comments). It will focus only on text, and try to make sense of slang, emojis, and even sarcasm.

The tool will show results in an easy-to-use dashboard so you can see emotional trends in real time. For this hackathon, it will work only with English text and public posts (like from Twitter). Things like images, other languages, or deep long-term analysis are not included in this version but could be added later.

4.Data Sources

For this project, we will use the GoEmotions dataset, a publicly available dataset developed by Google Research, and hosted on GitHub. It contains over 58,000 English Reddit comments labeled with 27 different emotions such as joy, anger, sadness, and surprise, along with a neutral category. This dataset is well-suited for training and evaluating emotion detection models.

- Source: GitHub (by Google Research)

- Access Type: Public

- Type: Static (downloaded once)

- Language: English

- Format: CSV/TSV

If real-time emotion detection is implemented, we may also use the Twitter API to collect recent tweets using specific hashtags or keywords. This secondary data source is:

- Source: Twitter API (X Developer Platform)

- Access Type: Public (with API key)

- Type: Dynamic (live data)

- Usage: For testing real-time predictions after training the model

5.High-Level Methodology

1. Data Collection

- Use the **GoEmotions dataset** (public, static) with 58,000 Reddit comments labeled by emotion.

- Collect **real-time tweets** using the **Twitter API** for testing and live emotion detection.
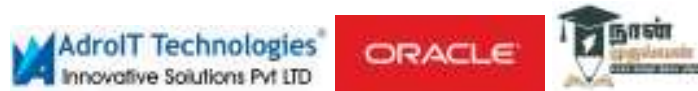
2. Data Cleaning

- **Remove duplicates** and handle **missing values** to ensure data quality.
- **Normalize text** by converting to lowercase, removing special characters, emojis, and unnecessary whitespace.

3. Exploratory Data Analysis (EDA)

- Visualize **emotion distribution** and **word frequency** using bar charts and word clouds.

- Analyze **text length** and **sentiment polarity** to understand comment structure and tone.

4. Feature Engineering

- Create features like **text length**, **word count**, and **sentiment score**.

- Transform text using **TF-IDF**, **word embeddings**, or **BERT tokenizer** for model input.

5. Model Building

Train baseline models like **Naive Bayes** and **SVM** for quick testing.
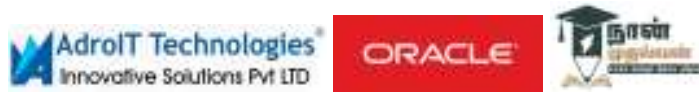
5. Model Building

Train baseline models like **Naive Bayes** and **SVM** for quick testing.

Use **LSTM** and **BERT-based models** for more accurate, context-aware emotion classification.

6. Model Evaluation

- Evaluate using **accuracy**, **precision**, **recall**, and **F1-score**.

- Use **confusion matrix** and **cross-validation** to assess performance and stability.

7. Visualization & Interpretation

- Present results through **charts, graphs**, and **confusion matrices**.

- Show **emotion trends** over time and model predictions in an **interactive dashboard**.

8. Deployment

- Deploy as a web app using Streamlit or Flask.

- Host on Heroku or AWS for public access and real-time use.

6.Tools and Technologies

🖥️ Programming Language

- **Python**: The primary language due to its strong ecosystem for data science, NLP, and machine learning.

🧠 Notebook / IDE

- **Google Colab**: For easy collaboration, GPU support, and cloud-based execution.

- **VS Code** *(optional)*: For local development and deployment setup.

📦 Libraries

Data Processing

- **pandas**: For data cleaning and manipulation.

- **numpy**: For numerical operations and array handling.

Text Processing / NLP

- **NLTK / spaCy**: For text cleaning, tokenization, and language processing.

- **Transformers (Hugging Face)**: To use pretrained models like **BERT** for emotion classification.

  - **TextBlob / VADER**: For extracting sentiment scores (optional features).

Modeling

- **scikit-learn**: For baseline models (Naive Bayes, SVM) and evaluation metrics.

- **TensorFlow / Keras**: For deep learning models (LSTM, neural nets).

- **PyTorch** *(optional)*: If using transformer models via Hugging Face.

Visualization

- **matplotlib** and **seaborn**: For plotting charts, graphs, and emotion distributions

- **wordcloud**: For generating emotion-based word clouds.

- **plotly**: For interactive graphs and dashboards (optional).

🌐 Optional Tools for Deployment

- **Streamlit**: To build and deploy an interactive web app for real-time emotion detection.

- **Flask**: For creating a lightweight web API to serve the model.

- **Heroku / AWS**: For cloud hosting and public access to the app.

7.Team Members and Roles

1. Team Leader – [SATHYAPRIYA M]

- Plans and manages the project.

- Oversees progress and helps in model selection and integration.

2. Data Handler –[MADHUMITHA S]

- Collects and cleans the data (from GoEmotions and Twitter).

- Prepares the dataset for model training.

3. Model Developer –[YAZHINI T]

- Builds and tests machine learning models.

- Evaluates model accuracy and performance.

4. App & UI Developer – [SRI DHARSHIKA S]

- Creates the web app using tools like Streamlit or Flask.

- Adds visualizations and deploys the final project.