

- github link: <https://github.com/Arunachalam8525/Phase2-project.git>
-
- **TOPIC : DECODING EMOTIONS THROUGH SENTIMENT ANALYSIS OF SOCIAL MEDIA CONVERSATIONS**

- **Problem Statement**

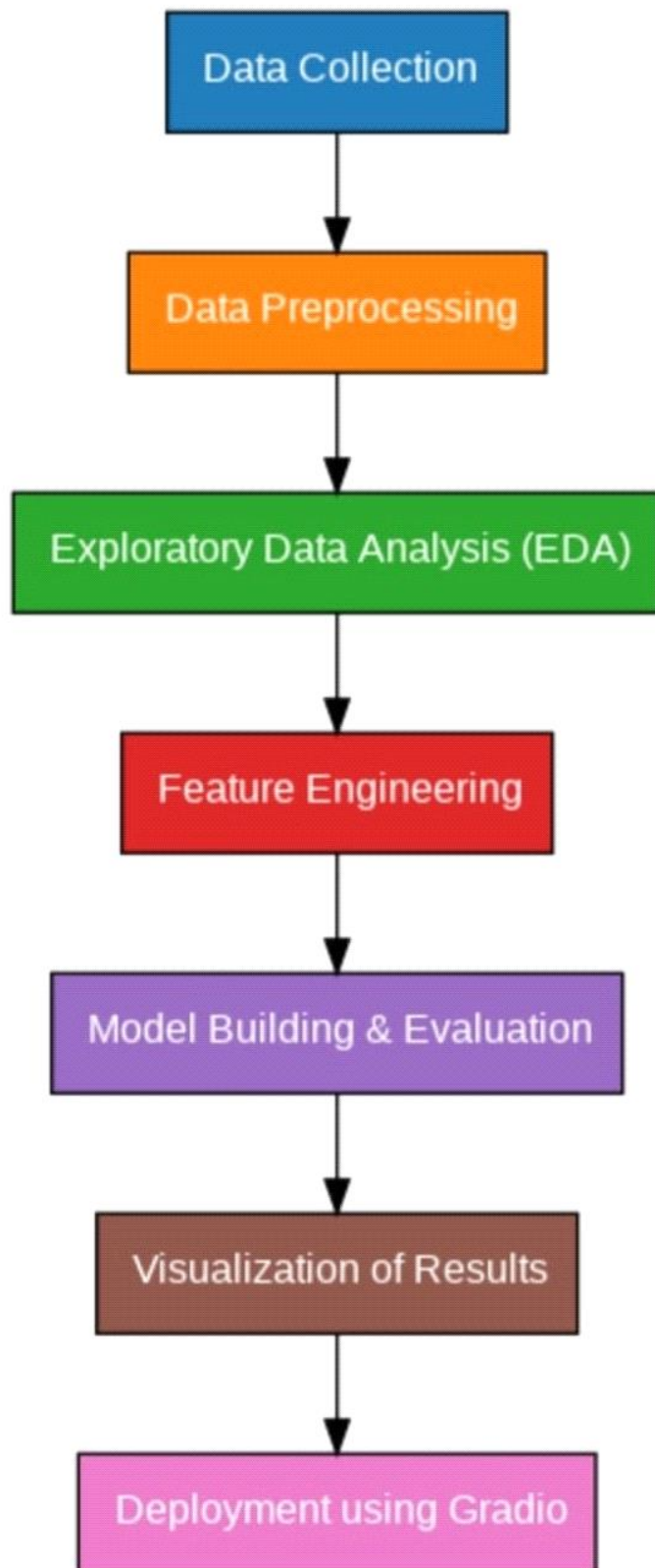
Sentiment Analysis aims to determine the emotional tone behind a series of words, often used to understand the attitudes, opinions, or emotions expressed within online reviews, social media, and more. The goal of this project is to build a robust model that can accurately classify text as positive, negative, or neutral based on its sentiment.

- **Project Objectives**

- Develop a machine learning model that accurately predicts the sentiment of text data.
- Identify the most influential features that impact sentiment classification.
- Provide insights into how linguistic patterns and word usage reflect sentiment.
- Ensure model interpretability and usability in real-world applications like social media monitoring, brand analysis, and feedback systems.

- **Flowchart of the Project Workflow**

3. Flowchart of the Project Workflow



- Data Collection
- Data Preprocessing (Tokenization, Stop-word Removal, Stemming)
- Exploratory Data Analysis (EDA)
- Feature Engineering (TF-IDF, Word Embeddings)
- Model Building (Logistic Regression, SVM, Neural Networks)
- Model Evaluation (Accuracy, Precision, Recall, F1-Score)
- Deployment and Testing
- **Data Description**

- Dataset Name: Sentiment140 Dataset
- Source: Kaggle
- Type of Data: Textual data
- Records and Features: 1.6 million tweets with attributes such as tweet content, sentiment label, and user information.
- Target Variable: Sentiment (0 = Negative, 2 = Neutral, 4 = Positive)
- Static or Dynamic: Static dataset for initial training, dynamic updates possible through API.
- Dataset link : <https://www.kaggle.com/search?q=sentiment+analysis>
- Dataset : https://www.kaggle.com/datasets/abhi8923shriv/sentiment-analysis-dataset?utm_source=chatgpt.com

- **Data Preprocessing**

- Removed URLs, mentions, and special characters.

- Handled missing values and duplicates.
- Tokenized tweets into individual words.
- Performed lemmatization and stop-word removal.
- Applied TF-IDF and Word2Vec for vector representation.

- **Exploratory Data Analysis (EDA)**

- Visualized sentiment distribution with bar plots.
- Analyzed word frequency for positive and negative sentiments.
- Explored hashtag sentiment correlations.

- **Feature Engineering**

- Extracted key features like:
 - N-grams (bi-grams and tri-grams)
 - Part of Speech (POS) tagging
 - Sentiment lexicon features
 - Topic modeling with LDA
- Applied dimensionality reduction techniques (PCA, LSA).

- **Model Building**

- Algorithms Used:
 - Logistic Regression: Baseline model for interpretability.
 - Support Vector Machine (SVM): Captures non-linear patterns.
 - LSTM (Long Short-Term Memory): For sequential text analysis.
- Train-Test Split:

- 80% training, 20% testing
- Evaluation Metrics:
 - Accuracy, Precision, Recall, F1-Score
 - Confusion Matrix to visualize classification performance.
- **Visualization of Results & Model Insights**
 - Displayed confusion matrices for model comparison.
 - Feature importance analysis for interpretability.
 - Visualized model performance over epochs for neural networks.
- **Tools and Technologies Used**
 - Programming Language: Python 3
 - Notebook Environment: Jupyter Notebook / Google Colab
 - Key Libraries:
 - pandas, numpy for data handling
 - matplotlib, seaborn, plotly for visualizations
 - scikit-learn, TensorFlow, Keras for modeling
- **Team Members and Contributions**
 - **C. Annamalai:** Data Collection and Cleaning
 - **P. Arunachalam:** Exploratory Data Analysis (EDA), Feature Engineering
 - **T. Deepakraj:** Model Development, Documentation, and Reporting