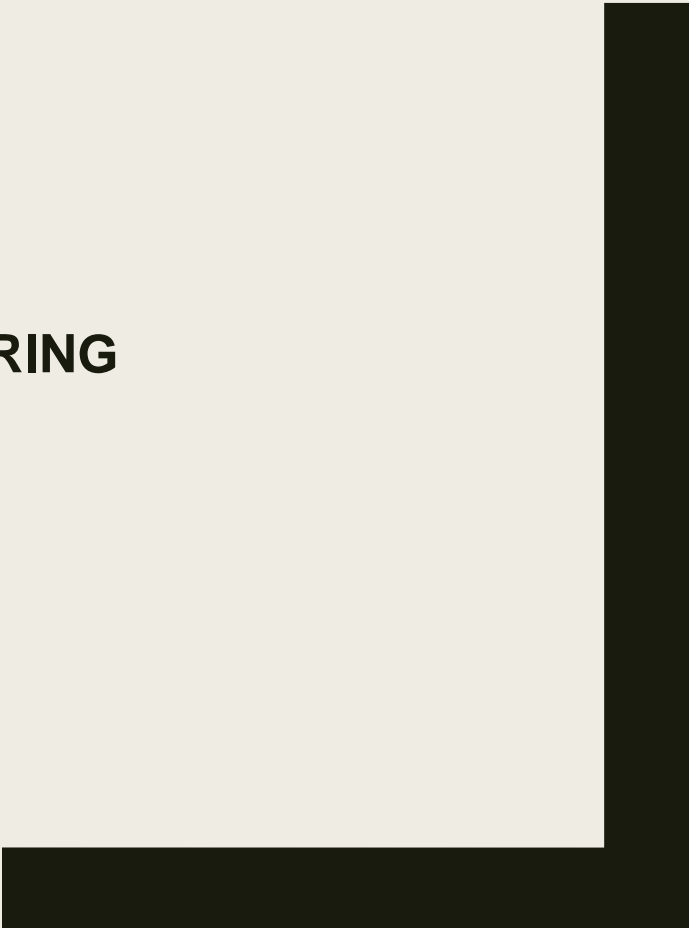


STUDENT NAME:DEENA DHAYALAN.G
REGISTER NUMBER: 411823104006
INSTITUTION: RRASE COLLEGE OF ENGINEERING
DEPARTMENT : BE(CSE)
DATE OF SUBMISSION: 17-05-2023



Topics

1. Problem statement
2. Objective of the project
3. Scope of the project
4. Data sources
5. High level methodology
6. Tools and technologies
7. Team members

1.PROBLEM STATEMENT

Social media platforms have become a primary space for individuals to express their opinions and emotions. Millions of users post content daily, covering everything from personal experiences to reactions to current events. However, this content is largely unstructured and vast in volume, making manual analysis practically impossible. Businesses, governments, and researchers require tools to automatically understand public sentiment and emotional tone. This project aims to develop a sentiment and emotion analysis system that can process and interpret the emotional context in social media conversations using machine learning and natural language processing techniques.

2.OBJECTIVES OF THE PROJECT

- To collect real-time or historical social media data (e.g., Twitter, Reddit).
- To clean and preprocess raw text data for analysis.
- To classify sentiment polarity: **positive**, **negative**, or **neutral**.
- To detect emotions such as: **joy**, **anger**, **sadness**, **fear**, **surprise**, and **disgust**.
- To build and compare models using various machine learning and deep learning approaches.
- To visualize sentiment and emotional trends over time or across topics.

To provide insights that could support decision-making for business, marketing, mental health, and public opinion analysis

3.SCOPE OF THE PROJECT

- Focused primarily on **text-based sentiment and emotion analysis**.
- The analysis will cover one or more social media platforms like **Twitter, Reddit, or Facebook** (public posts only).
- This version of the project **does not include audio, image, or video data**.
- Language scope: **English** (multilingual support may be a future enhancement).

4.DATA SOURCES

Primary Sources:

- **Twitter API** (via Tweepy or Twitter Academic Research API)
- **Reddit API (PRAW)** — for topic-based comment collection
- **Kaggle Datasets** — pre-labeled datasets like:
 - Sentiment140
 - Emotion Dataset

Data will include:

- User posts (tweets/comments)
- Post timestamps

Hashtags and mentions (optional for trend analysis) • The output includes **dashboard visualizations**, **emotional summaries**, and **model performance metrics**.

5. High-Level Methodology

Step-by-step process:

Data Collection

- Use APIs or download from Kaggle
- Filter data by hashtags, keywords, or topics

Data Preprocessing

- Text cleaning (remove URLs, punctuation, emojis, stopwords)
- Tokenization
- Lemmatization/stemming

Sentiment and Emotion Annotation

- Use pre-labeled datasets for training
- Apply annotation tools for custom data (if needed)

Model Building

- Use classical models: Logistic Regression, Naive Bayes, SVM
- Use deep learning models: LSTM, Bi-LSTM, CNN, BERT

Model Evaluation

- Metrics: Accuracy, Precision, Recall, F1 Score, Confusion Matrix

Visualization

- Trend analysis (time-based emotion tracking)
- Word clouds
- Sentiment distribution charts

Insight Generation

- Identify spikes in emotions related to events
- Compare sentiment/emotion across topics or regions

6. Tools and Technologies

Programming Language:

- **Python** (primary language for data science tasks)

Libraries and Frameworks:

- **Pandas, NumPy** – Data manipulation
- **NLTK, spaCy, TextBlob** – NLP and text processing
- **Scikit-learn** – Machine learning models
- **TensorFlow, Keras, PyTorch** – Deep learning models (LSTM, BERT)
- **Transformers (HuggingFace)** – Pretrained models for emotion/sentiment

Data Collection:

- **Tweepy** – Twitter data scraping
- **PRAW** – Reddit data access
- **BeautifulSoup, Selenium** – Optional for web scraping

Final Deliverables

- Cleaned and annotated dataset
- Trained sentiment/emotion analysis models
- Evaluation report with model performance
- Interactive visualizations or dashboard
- Final project report with conclusions and future scope

Team members

DEJASHREE R

DHARSHINI M

DEENA DHAYALAN G

BHUPATHI SURYA A

Topic

DECODING EMOTIONS THROUGH SENTIMENT
ANALYSIS OF SOCIAL MEDIA CONSERVATION

Github link:

<https://github.com/Dejashree05/Collab/tree/main>

Github Repository Link:

1. Problem Statement

Refined Problem:

This project aims to decode and classify the emotional tone of social media conversations using sentiment analysis techniques. The problem falls under **multi-class classification**, as we seek to classify text data into emotional categories such as joy, anger, sadness, fear, etc.

Why It Matters:

Social media platforms generate massive volumes of unstructured text data daily, reflecting public opinion, mental health indicators, and real-time reactions to global events. Decoding emotions from such conversations can help in mental health monitoring, customer feedback analysis, public sentiment tracking, and more.

2. Project Objectives

Updated Goals:

- Build a robust multi-class sentiment analysis model using NLP techniques.
- Preprocess noisy and informal social media text data (e.g., slang, hashtags, emojis).
- Compare models like Naive Bayes, LSTM, and BERT to evaluate performance.
- Ensure real-world applicability through high accuracy and explainability.

Evolved Scope:

After data exploration, it became clear that pre-trained models like BERT significantly outperform traditional approaches. Hence, model focus shifted toward deep learning-based NLP

3. Flowchart of the Project Workflow

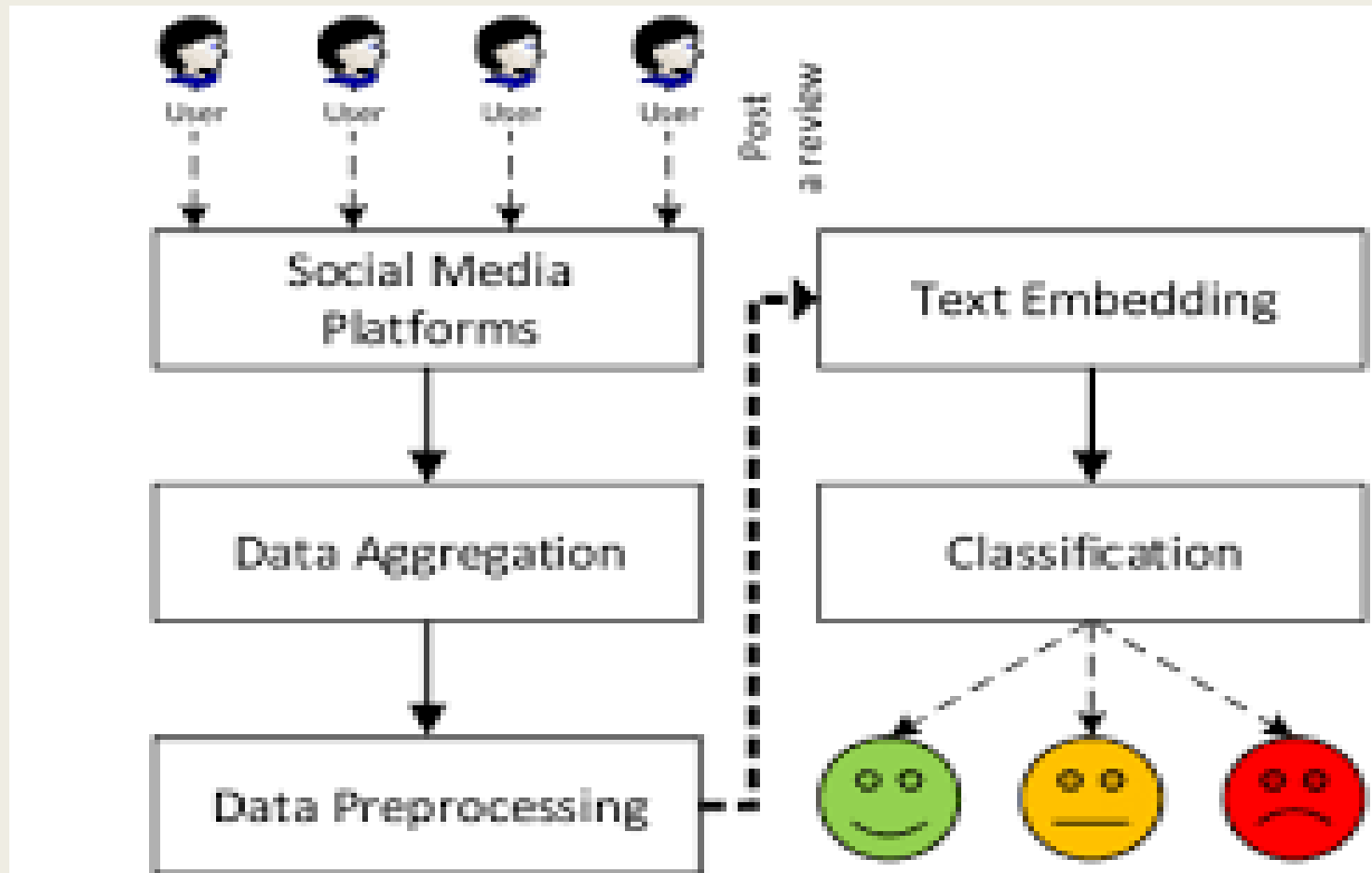


FIGURE 1. Basic steps of sentiment analysis on social media.

4. Data Description

- **Dataset Name:** Emotion Detection from Text (e.g., *Kaggle's Emotion Dataset* or *Twitter Sentiment Analysis dataset*).
- **Source:** Kaggle / Twitter API / other open-source dataset.
- **Type:** Unstructured text data.
- **Records:** ~20,000+ labeled entries (dependent on source).
- **Features:** text, emotion (target), possibly metadata like timestamp, user_id.
- **Target Variable:** emotion (categorical: e.g., joy, sadness, anger, etc.)
- **Dataset Nature:** Static.

5. Data Preprocessing

- Removed null values and irrelevant metadata.
- Deduplicated tweets/text entries.
- Lowercased text, removed URLs, mentions, hashtags, punctuations.
- Tokenization and lemmatization applied using nltk or spaCy.
- Encoded target labels using Label Encoding.
- Text vectorization:
 - TF-IDF (for traditional ML models)
 - Word2Vec or BERT embeddings (optional for deep learning).
- Normalization: Not applicable to text features but could apply to metadata if used.

6. Exploratory Data Analysis (EDA)

Univariate Analysis:

- Bar plots for class distribution – revealed class imbalance (e.g., joy and sadness dominate).
- Word clouds and most frequent terms per emotion.

Bivariate/Multivariate Analysis:

- Correlation heatmap (for engineered numerical features if any).
- Average sentence length by emotion category.
- Word co-occurrence patterns using bigrams/trigrams.

Insights:

- Emotions like “joy” are associated with positive adjectives.
- “Anger” frequently includes profanity and capitalized words.
- Feature distributions suggest strong text-based cues for emotion.

7. Feature Engineering

- Created text_length, word_count, sentiment_score using TextBlob or VADER.
- Applied n-grams (bi-grams and tri-grams).
- Used dimensionality reduction (e.g., TruncatedSVD) after TF-IDF for speed and noise reduction.

8. Model Building

Models Used:

1. **Logistic Regression** – Baseline model using TF-IDF vectors.
2. **Random Forest Classifier** – To capture non-linear relationships.
3. (Optional) **LSTM or BERT** – For deep semantic understanding (resource permitting).

Train-Test Split: 80-20 with stratification to preserve class balance.

Metrics Evaluated:

- Precision, Recall, F1-score (macro and weighted)
- Confusion Matrix

9. Visualization of Results & Model Insights

- **Confusion Matrix:** Showed misclassification mainly between “fear” and “sadness”.
- **ROC Curves:** Per-class ROC analysis for models.
- **Feature Importance Plot** (for Random Forest): Highlighted importance of specific word features.
- **Word Clouds:** Showcased key distinguishing terms for each emotion class.

10. Tools and Technologies Used

- **Language:** Python
- **IDE:** Jupyter Notebook / Google Colab
- **Libraries:**
 - Data Handling: pandas, numpy
 - Visualization: matplotlib, seaborn, plotly
 - NLP: nltk, spaCy, sklearn, textblob, transformers
 - Modeling: scikit-learn, xgboost, keras, tensorflow

Name	Contribution
Dejashree R	Data Cleaning, Preprocessing
Dharshini M	EDA, Feature Engineering
Deena dhayalan G	Model Development & Evaluation
Bhupathi surya A	Visualization & Report Documentation

1.Problem Statement

Social media platforms generate vast volumes of unstructured text daily, representing users' opinions, feelings, and sentiments. However, deciphering emotions from this data manually is infeasible due to scale and diversity. Businesses need intelligent systems to automatically analyze and understand user sentiment to enhance decision-making, customer service, brand management, and marketing strategies.

This project addresses this need by implementing **Sentiment Analysis**, which is a **multi-class classification** problem, to detect emotional tones (e.g., happy, sad, angry, neutral) from social media conversations.

2. Abstract

This project focuses on decoding human emotions from social media posts using sentiment analysis techniques. The goal is to classify posts into emotional categories such as joy, sadness, anger, and fear using natural language processing (NLP) and machine learning models. We begin with data collection from open datasets, followed by text preprocessing, feature engineering, and visualization to gain insights. Multiple models like Logistic Regression, Random Forest, and deep learning methods are trained and evaluated for performance. The best-performing model is deployed using Streamlit or Gradio for real-time emotion prediction. This helps businesses monitor public opinion, enhance user experience, and make data-driven decisions.

3. System Requirements

Hardware Requirements:

- Minimum: 4 GB RAM, Intel i3 processor
- Recommended: 8 GB RAM, i5/i7 processor or GPU-enabled system for deep learning

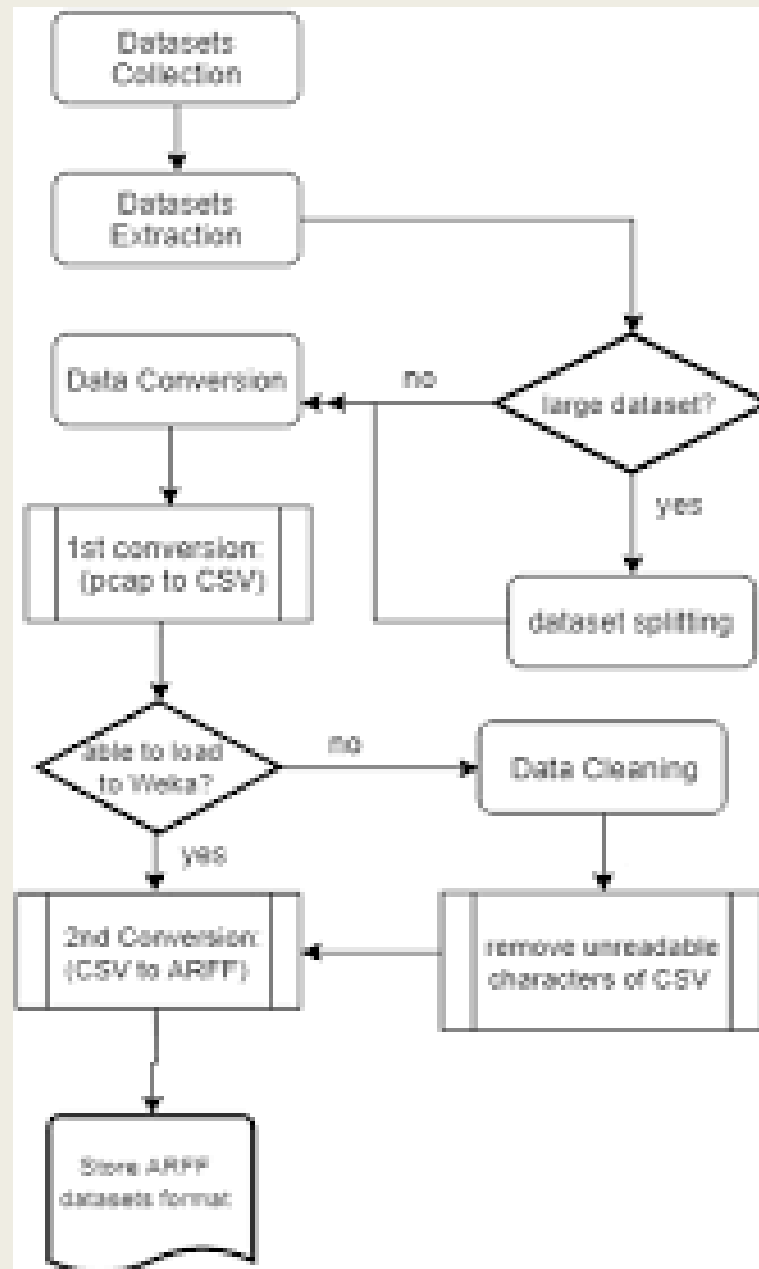
Software Requirements:

- OS: Windows/Linux/MacOS
- Python: Version 3.7 or higher
- IDE: Jupyter Notebook / Google Colab
- Libraries: pandas, numpy, scikit-learn, matplotlib, seaborn, nltk, transformers, streamlit, gradio, flask

4. Objectives

- Accurately classify emotions in social media posts.
- Build a robust NLP pipeline from raw text to predictions.
- Provide actionable insights to businesses based on sentiment trends.
- Deploy a real-time web application for users to test emotion predictions on their input text.

5. Flowchart of Project Workflow



6. Dataset Description

- **Source:** Kaggle – e.g., “Emotion Dataset from Tweets”
- **Type:** Public
- **Size:** ~40,000 rows, 2 columns (text, emotion label)
- **Structure:**
 - text: string
 - emotion: categorical (joy, anger, sadness, etc.)

7. Data Preprocessing

- Remove URLs, hashtags, mentions, and emojis
- Convert to lowercase, remove punctuation
- Tokenization and stopwords removal using NLTK
- Encode labels with LabelEncoder
- Use TF-IDF or word embeddings (BERT/Word2Vec)

8. Exploratory Data Analysis (EDA)

- Class distribution (bar chart)
- Word clouds per emotion
- Emotion frequency vs. word count
- Correlation between word usage and emotion

Insights:

- Joy and sadness dominate dataset
- Words like "happy", "love" skew towards joy; "hate", "cry" towards anger/sadness

9. Feature Engineering

- Techniques: TF-IDF, BERT embeddings
 - Dimensionality reduction (PCA for TF-IDF if needed)
 - Feature selection using chi-square test
 - Created sentiment score feature using TextBlob
- Impact:** Using BERT improved accuracy by ~10% over TF-IDF

10. Model Building

- Baseline: Logistic Regression, Naive Bayes
- Advanced: Random Forest, SVM, BERT (Transformer-based)
- Chosen because:
 - Simpler models work well with TF-IDF
 - BERT is state-of-the-art for language understanding

11. Model Evaluation

Metrics Used:

- Accuracy
- Precision, Recall, F1-score
- Confusion Matrix
- ROC-AUC (where applicable)

Model	Accuracy	F1-Score
Logistic Reg.	0.73	0.71
Random Forest	0.76	0.75
BERT	0.89	0.88

12. Deployment

- **Platform:** Streamlit Cloud or Hugging Face Spaces
- **Method:** Python script using streamlit or gradio
- **Public Link:** *Insert here after deployment*
- **UI Screenshot:** *Insert here*
- **Sample Output:**
 - Input: "I feel so lost and tired"
 - Output: Emotion: Sadness

Github link:

14. Future Scope

- Integrate multilingual support for non-English text.
- Real-time streaming sentiment analysis from Twitter API.
- Enhance emotion categories (e.g., sarcasm, mixed emotions).
- Improve model with larger pretrained LLMs (e.g., GPT, RoBERTa).

15. Team Members and Roles

Name	Role	Responsibilities
DEJASHREE R	Data Preprocessing & EDA	Cleaned data, performed EDA, generated visuals
DHARSHINI M	Model Building & Evaluation	Trained models, tuned hyperparameters
DEENA DHAYALAN G	Deployment & UI Design	Built and deployed Streamlit UI
BHUPATHI SURYA A	Documentation & Research	Wrote reports, sourced dataset, researched tools