1. **Text Preprocessing:**
    - Converted text to **lowercase**.
    - Removed **numbers, special characters, and punctuation**.
    - Eliminated **stopwords** using NLTK's predefined stopword list.
2. **Feature Engineering:**
    - Applied **TF-IDF vectorization** (max_features=10,000) to transform text into numerical features.
3. **Model Training & Evaluation:**
    - Split the dataset into **80% training and 20% testing**.
    - Trained a **Naïve Bayes classifier** on the processed data.
    - Evaluated using **accuracy, F1-score, and confusion matrix**.

Challenges Faced

- **Handling Text Variability:** The model struggled with **misspellings, sarcasm, and nuanced language**, which Naïve Bayes does not inherently handle well.
- **Feature Independence Assumption:** Naïve Bayes assumes that all features are independent, which is often unrealistic in natural language processing.
- **Imbalanced Classes (If Any):** If one sentiment was more frequent than others, it could bias predictions.

Model Performance & Improvements

1. **Performance Metrics:**
    - **Accuracy: 85.72%**
    - **F1-Score: 85.81%**
2. **Observations:**
    - The model performed **reasonably well**, capturing overall sentiment effectively.
    - However, **TF-IDF with Naïve Bayes may struggle with complex sentence structures and context-dependent words**.
3. **Possible Improvements:**
    - **Use Word Embeddings:** Implementing **Word2Vec** or **pretrained embeddings like GloVe** could improve semantic understanding.
    - **Try Advanced Models:** Exploring **Logistic Regression, LSTMs, or BERT** may enhance accuracy.
    - **Hyperparameter Tuning:** Adjusting **smoothing parameters (alpha)** in Naïve Bayes could refine predictions.