**FINAL: Sentimental analysis**

This notebook demostrates on **Sentiment Analysis**, a natural language processing (NLP) task that determines the emotional tone behind a body of text. The primary goal is to classify textual data (such as reviews, comments, or posts) into categories like **positive, negative, or neutral.** The dataset containing labeled text samples was used for training and testing the model. The dataset was first downloaded and cleaned. It was then split into **training and testing sets** to evaluate model performance effectively. Text data was converted into numerical form using **TF-IDF (Term Frequency – Inverse Document Frequency).** This step ensures that important words contributing to sentiment carry higher weights than common stopwords.

***Classical Machine Learning Pipeline***

The pipeline included data preprocessing, vectorization, training, and evaluation. The model was trained using the **training set.** Evaluation was performed on the **test set** with metrics such as **accuracy, precision, recall, and F1-score.** A **confusion matrix** was also generated to visualize correct vs. incorrect predictions.The model successfully classified sentiments with good performance.The **confusion matrix** showed that most predictions aligned correctly with ground-truth labels.Any misclassifications mainly occurred between **neutral and slightly positive/negative** cases, which is common in sentiment analysis due to subjectivity in text.The trained sentiment analysis model was **saved using the Joblib library,** allowing it to be reused without retraining.A test stage verified that the saved model produced consistent results when reloaded.

***API(gemini) using Machine learning***

As an extension, the project explored using **Gemini API** for text processing and predictions. This integration shows potential for combining **custom ML models** with **external AI services** to enhance sentiment analysis capabilities. This project successfully demonstrates how to build a **complete sentiment analysis pipeline** from **data preprocessing** to **feature extraction (TF-IDF)** to **training ML models** to **evaluating with confusion matrix** to **saving the model** to **API integration.** The approach is scalable and can be applied to larger datasets or integrated into real-world applications such as **chatbots, recommendation engines, and social media analytics tools.**