Assignment #1: Sentiment Analysis with Various Approaches
Deadline: 5th January (Sunday)
Marks: 10

Note: *This is a group assignment. On the first page of the report, clearly mention the contribution of each member. It is not necessary that each member has worked on all the tasks, but whatever has been done should be highlighted without any ambiguity or vague wording, as I may opt for a follow-up viva of certain groups.*

A) Objective:
The objective of this assignment is to explore and compare classical machine learning methods with modern fine-tuned pre-trained language models (PLMs) for sentiment analysis.

B) Dataset:
You will work with the IMDb movie reviews dataset, containing a large collection of movie reviews labelled as positive or negative sentiment. There are two files: training and testing. Use the training dataset (or portion of it) to train your machine learning model, or finetune PLMs (RoBERTa, DistillBERT). The test data should be used for evaluation purposes.

C) Tasks:
You need to perform sentiment analysis using the following approaches and benchmark the scores against [distilbert-base-uncased-finetuned-sst-2-English]:

1. Classical Machine Learning Approaches:

- Preprocess the text data using tokenization, stemming/lemmatization, POS Taggers and stopword removal.
- Vectorize the text data using different variations of TF-IDF or CountVectorizer.
- Train at least four different machine learning models (e.g., Naive Bayes, Random Forest, k-NN, Gradient Boosting, etc) on the processed training dataset.
- Evaluate the accuracy of these models on the test data.

2. Fine-tuning Pre-trained Language Models (PLMs):

- Fine-tune three different pre-trained language models (PLMs):
    a) DistilBERT
    b) RoBERTa
- Add LoRA (Low-Rank Adaptation) layers for parameter-efficient fine-tuning.
- Compare their performance with the classical ML models.
- Experiment with different LoRA configurations
    - rank
    - matrices to modify

    - batch size
    - learning rate
    - number of epochs
- Evaluate the fine-tuned PLMs on the test data using accuracy, precision, recall, and F1 score.

3. Experimentation, Comparison and Insight:

- Discuss the impact of different configurations on performance (e.g., increasing rank in LoRA, larger batch size, number of epochs, choice of matrices to modify, etc.).
- Discuss training time, resource requirements, and performance metrics when compared against the ML models.
- Highlight cases where classical ML models performed well and where PLMs excelled.

D) Deliverables:

a) **Code:**

- Submit well-documented code for each task.
- Ensure separate Python Notebooks for classical ML and PLMs, with clear naming.

b) **Report:** Write a comprehensive report covering:

- Preprocessing steps for classical ML and fine-tuning configurations for PLMs.
- Parameters of winning models (e.g., TF-IDF settings, hyperparameters of ML models, LoRA configurations).
- Performance comparison across models.
- Training time and resources required for each approach.
- Insights and conclusions about the strengths/weaknesses of each method.
- Benchmark scores against distilbert-base-uncased-finetuned-sst-2-English