LLM/NLP Engineer Project Task

Portfolio News Monitoring & Summarization

Develop an automated agent that:

- Monitors and fetches financial news related to a stock portfolio (e.g., Tesla, Apple or S&P 500).
- Summarizes key articles using both a baseline (e.g. zero-shot/few-shot) and a fine-tuned model.
- Presents concise, high-quality summaries.

Scope and Requirements

- Use **Open-source** technologies.
- **Lightweight solution** due to computational resource limits (e.g., fine-tuning can happen on Colab Free Tier GPUs or identical).
- Either **scrape financial news** yourself (e.g., Yahoo Finance), use an API or use a relevant **dataset**.
- Push all code to a public **GitHub repository** with:
    - Clear documentation.
    - Easy setup instructions (environment, dependencies).
    - Strongly recommended**:** provide a Dockerfile with a docker-compose setup (bonus points).
- **Compare**:
    - Zero-shot or few-shot summarization baseline.
    - Fine-tuned model (small-scale experiments).
- **Evaluate performance**:
    - Go beyond standard metrics like ROUGE or BLEU.
    - Assess the factual accuracy of summaries and identify hallucinations in a meaningful way.
    - Also asses and report the improvement of the fine-tuning effect.

Resourceful Data

You are encouraged to creatively build or source your dataset. **Some** Options include:

- Using open datasets (whether originally annotated with summaries or not).
- Scraping (either programmatically or agentically) financial news articles (whether they originally include summaries or not).
- Generating artificial articles (with or without summaries).
- Using a more sophisticated and heavy LLM to generate summaries (and/or annotations) for training data and apply distillation techniques to fine-tune a smaller model.

Note:
The **less synthetic** the original data and annotations are, and the **more original** your dataset is, the more credit you will earn.

Efficiency Matters

- Work within free-tier resources (e.g., Colab Free GPU or identical).
- Choose **small models** or **parameter-efficient fine-tuning methods** that fit the resource constraints.
- The **size of the pre-trained model does not matter** as long as:
    - It fits the available resources.
    - Fine-tuning demonstrably improves the results compared to the zero-shot/few-shot baseline.

Deliverables

1. Shared Git repository containing:
    - Full source code.
    - Clear environment setup instructions (Dockerfile highly recommended).
    - Example scripts or instructions for running inference.
2. A detailed report including:
    - Technical choices made (models, fine-tuning strategies, data sources, etc.).
    - Reasoning behind these choices (trade-offs considered, resource constraints handled).
    - Description of challenges faced and how they were addressed.
    - Evaluation results and discussion (quantitative and qualitative).
    - Some examples of the summarized news results.

3. (Optional but encouraged)
   A simple UI or dashboard to display the summarized news results after an inference.

Notes & Evaluation Criteria

- The project aims to assess how fine-tuning alters summarization quality (for at least one model) over baseline methods, with a particular focus on the evaluation process (including factuality and hallucinations).
- Creativity and problem solving is highly valued. Showcase how you adapt when facing challenges such as limited data, limited compute, or noisy evaluation.
- It is acceptable to create synthetic datasets if necessary, but higher value is placed on the use of original and high-quality, real-world data when possible.
- Clarity and reproducibility of the codebase is highly recommended (good documentation, report, clear environment setup).