Technical Assignment: Sentiment Analysis System with Custom Fine-Tuned Model and Llama 3 Models

---

Objective

Students will:

1. Fine-tune a sentiment analysis model on the IMDB dataset.
2. Save and upload the model to Hugging Face.
3. Test their fine-tuned model alongside the pre-trained **Llama 3 model** via the Groq Cloud API.
4. Explain how to design systems and test models.
5. Submit a GitHub repository and a YouTube video link in the notebook.
6. You can use Kaggle, which provides 30 hours of free GPU usage each week. Simply verify your phone number on Kaggle to access it.
7.

Total Points: 15

Steps and Tasks

Part 1: Dataset Preparation and Fine-Tuning (7 points)

Step 1: Download the IMDB Dataset (1 point)

1. Use the IMDB dataset from Kaggle: `/kaggle/input/imdb-dataset/IMDB Dataset.csv`.
2. Load the dataset using Pandas and verify it in your notebook.

Step 2: Data Preprocessing (1 point)

1. Clean and preprocess the dataset:
    - Encode the `sentiment` column (`positive -> 1`, `negative -> 0`).
    - Retain only the `review` and `label` columns.

2. Split the data into **training** and **validation, testing**

Step 3: Model Selection and Tokenization (1 point)

1. Select a pre-trained Hugging Face transformer model for fine-tuning (e.g., `distilbert-base-uncased`).
2. Tokenize the dataset with (see if required)
    - Truncation.
    - Padding.
    - Maximum sequence length of 256.

Step 4: Fine-Tune the Model (2 points)

1. Fine-tune the model on the IMDB dataset for **2 epochs** using the Hugging Face `Trainer`.
2. Set training parameters:
    - Learning rate: `5e-5  or your own`
    - Batch size: `16 or 32`
    - Evaluation at the end of each epoch.
3. Ensure that metrics like `accuracy`, `precision`, `recall`, and `F1-score` are logged during training.

Step 5: Save and Upload the Model to Hugging Face (2 points)

1. Save the fine-tuned model and tokenizer locally using `save_pretrained()`.
2. Log in to Hugging Face using `notebook_login`.
3. Upload the model to Hugging Face using `push_to_hub`.
4. Verify the model on Hugging Face Hub and include the link in your notebook.

Part 2: API Development and Testing (5 points)

Step 6: Set Up the Backend API (1 point)

1. Use **FastAPI** or **Flask, Express, Nest Nodejs** to create an API.
2. Define a POST endpoint (`/analyze/`) that:

- Accepts:
  - `text`: The input text for sentiment analysis.
  - `model`: A parameter specifying the model to use (`custom` or `llama`).
- Returns:
  - Sentiment (`positive` or `negative`).
  - Confidence score.

Step 7: Load Models (1 point)

1. Load the fine-tuned model from Hugging Face.
2. Access the Llama 3 model using the Groq Cloud API.

Step 8: Test the API Locally (1 point)

1. Test the `/analyze/` endpoint with both models (`custom` and `llama`) using:
   - Postman.
   - `curl`.
   - Python requests.

Step 9: Define the Llama 3 Prompt (1 point)

1. Write a clear and reusable prompt for the Llama 3 model in Groq Cloud.
   Example: can be improved more

```
"Classify the sentiment of this text as positive or negative:
'This movie was fantastic"
```

Step 10: Test with Both Models (1 point)

1. Verify that the API works for both the fine-tuned model and the Llama 3 model.
2. Ensure the results return the sentiment score too.
3. For Groq you can add into prompt,
4. ---

Part 3: UI Design and Explanation (3 points)

Step 11: React UI Design (1 point)

- A text input field for user input.
- A dropdown menu for model selection:
  - **Custom Model**.
  - **Llama 3**.
- A button labeled **"Analyze Sentiment"** to send input and selected model to the backend API.
- A result display section showing:
  - Sentiment (`positive` or `negative`).
  - Confidence score(optional)

Step 12: Submit GitHub Repository (1 point)

1. Upload all code (notebook, backend, and UI explanation) to a public GitHub repository.
2. Include a `README.md` file that explains how to:
   - Install dependencies.
   - Run the notebook and API locally.
   - Use the endpoints.

Step 13: Record a YouTube Demo Video (1 point)

1. Record a demo video (2-3 minutes) showing:
   - Testing the system with both models (`custom` and `llama`).
   - One question with custom fine and one with llam3 any llama 3 will be fine.
2. Upload the video to YouTube and include the link in your notebook.

Grading Points

Task	Points
Downloading and Cleaning Dataset	1
Data Preprocessing	1
Model Selection and Tokenization	1
Fine-Tuning the Model	2
Save and Upload Model to Hugging Face	2
Backend API Setup	1
Loading Models	1
Local API Testing	1
Llama 3 Prompt Definition	1
Testing Both Models via API	1
React UI Design Explanation	1
GitHub Repository Submission	1
YouTube Demo	1
Total	**15**

Submission Requirements

1. **Notebook .ipynb**:
    - Jupyter Notebook, either Google Colab, Kaggle, etc, also Provide the link to hugging face mode.
2. **GitHub Repository**:
    - Share the GitHub repo link in a notebook
3. **YouTube Video Link**:
    - Include the link to the demo video in the notebook.