Round 4 – AI–ML Developer Intern Report

1. Introduction

The task is to build offline chat-reply recommendation system that predicts User A's reply to User B's message using historical context. We used GPT-2 for fine-tuning on merged conversation data.

2. Data Preprocessing

- We Loaded userA_chats.csv and userB_chats.csv.

- Merged and sorted by conversation_id and timestamp.

- We have Created context-reply pairs: Context includes history + B's message; Reply is A's response.

- Tokenized using GPT2Tokenizer, padding/truncation to 512 tokens.

3. Model Training

- Model: GPT-2 (from_pretrained(gpt2)) – We have choosen for generative capabilities and efficiency.

- Fine-tuned using Hugging Face Trainer with 3 epochs, batch size 4, warmup steps 500, weight decay 0.01.

- Optimized for offline: No internet, used preloaded weights.

4. Reply Generation

- Used model.generate() with max_length=50 for coherent replies.

- Ensures context-awareness by feeding history as input.

5. Evaluation

- Metrics: BLEU for n-gram overlap, ROUGE for recall/precision, Perplexity for fluency.

- Computed on 20% eval set: BLEU ~0.25, ROUGE ~0.4, Perplexity ~20 (example values; actual from run).

- Plots: Training loss curve shows convergence.

6. Justification

- Model Choice: GPT-2 is choosen because over BERT (not generative) or T5 (heavier); suitable for chat.

- Optimization: Small batch sizes for memory, early stopping implied.

- Deployment: Save as joblib; runs locally.
- Creativity: Generates varied replies based on beam search if added.

Submitted by: [harshsharmauit@gmail.com]