Summer Training TR-103 Prompt Engineering

Day 9 Report

Fine-Tuning Process

Understood how to fine-tune the OpenAI model using custom prompt-response pairs.

• Two JSON files were generated:

- Training File – for fine-tuning the model.
- Test File – for evaluating the model after training

Important Note from Session

• Code generation, optimization, and debugging are independent of the finetuning process.

• These tasks should be performed and saved in separate files.

Practical Task

• Created a CSV file containing prompts (like “Write a C++ function to add

two numbers”) and their completions.

• Split this file into training (80%) and test (20%) datasets.

• Converted both datasets into JSONL format.

Model Testing

• Used the trained model to generate C++ code from prompts like:

- “Write a C++ function to reverse a string”
- “Write a C++ program to check if a number is prime”

• The model generated correct and relevant C++ code.

Code Optimization

• Provided basic C++ functions to the model and asked for optimization. The model returned more readable and efficient versions.

• The model provided more readable and efficient versions of the code.

Outcome

• Gained hands-on experience in fine-tuning and evaluating AI models.

• Learned how to use fine-tuned models to generate and improve real code.

• Understood the importance of organizing tasks into separate files.