STA5007 Assignment 2

Li Xuanran
12312110

December 5, 2025

1 Q1

Explain and answer the following questions in your own words.

1.1

(2 pts) Can LoRA match the performance of full fine-tuning, and if so, under which conditions?

Solution.

Yes, LoRA can match the performance of full fine-tuning in many tasks under the following conditions:

1. **Appropriate task type**: LoRA performs well only when the downstream tasks are not extremely different from the pretraining domain. Also, LoRA is more effective for tasks like RLHF and simple classification.
2. **Good hyperparameter selection**: We need to choose high rank and learning rate.
3. **Appropriate dataset size**: LoRA performs well when the dataset size is relatively small compared to the size of the pre-training data.
4. **Stable and optimized base model**: LoRA relies on fixed base model weights. If the pre-trained model is already optimized and stable, LoRA fine-tuning can reach parity.

□

1.2

(1 pt) How does learning rate and batch size effect the performance of LoRA-based PEFT methods?

Solution.

1. **Learning rate (LR)**: LoRA is more sensitive to the choice of LR, and needs a slightly higher LR than full fine-tuning. If LR is too low, it leads to slower convergence and causes underfitting; if LR is too high, GD becomes unstable and it causes the loss to diverge.

2. **Batch size**: LoRA enables larger batch size, and it leads to smoother gradient estimate and more stable convergence. If batch size is too small, it introduces more noise to the gradient, and leads to better generalization but unstable convergence, even becomes diverge; if batch size is too large, it leads to worse generalization.

□

1.3

(2 pts) Implement the TODOs in the attached python file `lora_training.py`. Then run the training script with Qwen/Qwen2.5-0.5B and show the printed results. Please refer to the notes in the code file for more instructions.

```
Downloading/Locating model: Qwen/Qwen2.5-0.5B...
Downloading Model from https://www.modelscope.cn to directory: /Users/lxr/.cache/modelscope/hub/models/Qwen/Qwen2.5-0.5B
Model local path: /Users/lxr/.cache/modelscope/hub/models/Qwen/Qwen2___5-0___5B

=== LoRA INJECTION ===
[LoRA] Replaced linear layers: 168

=== MARK TRAINABLE ===
[Params] trainable=4,399,104, total=498,431,872, ratio=0.8826%

--- Starting LoRA Training ---
Loaded 4 dummy samples.
Epoch 1 finished. Average Loss: 2.5103
Epoch 2 finished. Average Loss: 1.0608
Epoch 3 finished. Average Loss: 0.2968
Epoch 4 finished. Average Loss: 0.1268
Epoch 5 finished. Average Loss: 0.1319
--- Training Complete ---

=== LoRA MERGE ===
[LoRA] Merged modules: 168
Prompt:
Human: What is the capital of China? Assistant:

Generated Output:
Human: What is the capital of China? Assistant: The capital of China is Beijing. The capital of France is Paris. The capital of the United States is Washington, D.C. The capital of Germany
```

2 Q2

Explain and answer the following questions in your own words.

2.1

(2 pts) The principles behind different sampling methods: Greedy, Top-K, Top-P, and Min-P decoding.

Solution.

1. **Greedy**: Directly choose the next token which has the highest probability in every step.
2. **Top-K**: Select K tokens which have the top-K highest probabilities, then normalize the token distribution and sample the next token accordingly.
3. **Top-P**: Select the smallest number of top tokens such that their cumulative probability is at least P.
4. **Min-P**: Set a threshold $p_{\min}$, and in every step we filter all the tokens whose probability is less than $p_{\min}$.

□

2.2

(1 pt) How changing parameters such as `temperature`, `k`, `p`, and `p_min` influences the generated results for each decoding algorithm (if applied).

Solution.

1. `temperature` (T): It applies in Top-K, Top-P, and Min-P decoding, and does not apply in Greedy decoding. From the softmax

$$P_i^{\text{temp}} = \frac{\exp(\frac{\log P_i}{T})}{\sum_j \exp(\frac{\log P_j}{T})},$$

   we know that when $T < 1$, the smaller T is, the more likely to choose the highest probabilities; when $T > 1$, the greater T is, the flatter the distribution is, which becomes more stochastic.

2. `k`: In Top-K decoding, when k is small, let's say less than 10, the choice is more predictable and monotonous; when k is large, let's say more than 100, the choice is more diverse and increases noise.

3. `p`: Similar to k. In Top-P decoding, when $p \leq 0.5$, the choice is more predictable and monotonous; when $p \to 1$, the choice is more diverse and increases noise.

4. `p_min`: In Min-P decoding, when $p_{\min} = 0$, it means do not filter; when $p_{\min}$ is small, let's say 10^{-5}, it can filter extreme uncommon noise and leads it more stable; when $p_{\min}$ is large, let's say more than 10^{-2}, it excludes many optional tokens, and the output is conservative.

□

2.3

(2 pts) Complete the missing parts of the provided Python file `decoding.py`. Then run the training script with Qwen/Qwen2.5-0.5B-instruct and show the printed results.

```
Downloading/Locating model: Qwen/Qwen2.5-0.5B-Instruct...
Downloading Model from https://www.modelscope.cn to directory: /Users/lxr/.cache/modelscope/hub/models/Qwen/Qwen2.5-0.5B-Instruct
Model local path: /Users/lxr/.cache/modelscope/hub/models/Qwen/Qwen2___5-0___5B-Instruct

--- Starting Full Fine-Tuning Training ---
Prepared 2 training samples.
Epoch 1/5, Step 1, Loss: 4.0591
Epoch 1/5, Step 2, Loss: 1.9455
Epoch 1 finished. Average Loss: 3.0023
Epoch 2/5, Step 1, Loss: 0.9978
Epoch 2/5, Step 2, Loss: 1.1440
Epoch 2 finished. Average Loss: 1.0709
Epoch 3/5, Step 1, Loss: 0.2938
Epoch 3/5, Step 2, Loss: 0.3006
Epoch 3 finished. Average Loss: 0.2972
Epoch 4/5, Step 1, Loss: 0.1458
Epoch 4/5, Step 2, Loss: 0.1287
Epoch 4 finished. Average Loss: 0.1372
Epoch 5/5, Step 1, Loss: 0.0874
Epoch 5/5, Step 2, Loss: 0.0647
Epoch 5 finished. Average Loss: 0.0761
--- Training Complete ---

=== INFERENCE TEST AFTER TRAINING ===
Prompt:
<|im_start|>system
You are a helpful assistant.<|im_end|>
<|im_start|>user
What is the capital of China?<|im_end|>
<|im_start|>assistant


Generated Output:
system
You are a helpful assistant.
user
What is the capital of China?
assistant
The capital of China is Beijing.
```

Note: *All the answers are summarized by myself.*