

<b>Started on</b>	Thursday, 7 November 2024, 4:04 PM
<b>State</b>	Finished
<b>Completed on</b>	Thursday, 7 November 2024, 4:08 PM
<b>Time taken</b>	4 mins 4 secs
<b>Marks</b>	9.00/10.00
<b>Grade</b>	<b>90.00</b> out of 100.00

**Question 1**

Complete

Mark 1.00 out of 1.00

What is the primary purpose of Distributed Data Parallel (DDP) in PyTorch?

- ☐ a. To ensure that only one GPU is used during training.
- ☒ b. To parallelize model training across multiple GPUs, improving efficiency and speed.
- ☐ c. To reduce the number of parameters in the model.
- ☐ d. To run models sequentially on multiple GPUs.

**Question 2**

Complete

Mark 1.00 out of 1.00

In PyTorch's Distributed Data Parallel, what does the "all-reduce" operation achieve?

- ☒ a. Aggregates gradients across all GPUs and synchronizes them.
- ☐ b. Reduces the learning rate dynamically.
- ☐ c. Increases the batch size automatically.
- ☐ d. Divides the dataset among all GPUs.

**Question 3**

Complete

Mark 1.00 out of 1.00

In the Adam optimizer, what is the role of the first moment estimate?

- ☒ a. It tracks the moving average of gradients to help with momentum.
- ☐ b. It adjusts the learning rate based on the gradient's variance.
- ☐ c. It decreases the learning rate over time.
- ☐ d. It reinitializes weights after each iteration.

**Question 4**

Complete

Mark 0.00 out of 1.00

What does the rendezvous process facilitate in distributed training?

- ☒ a. Synchronizing model weights across GPUs.
- ☐ b. Saving checkpoints at regular intervals.
- ☐ c. Setting up the initial connection between multiple nodes.
- ☐ d. Changing the optimizer parameters dynamically

**Question 5**

Complete

Mark 1.00 out of 1.00

Fine-tuning a pre-trained model typically involves:

- ☒ a. Adjusting specific layers or weights for a new task or dataset.
- ☐ b. Removing the original layers and adding new ones.
- ☐ c. Training the model from scratch on a new dataset.
- ☐ d. Only using the original model without any changes.

**Question 6**

Complete

Mark 1.00 out of 1.00

The purpose of LoRA (Low-Rank Adaptation) in model fine-tuning is to:

- ☒ a. Allow fine-tuning by adding low-rank matrices without modifying original weights.
- ☐ b. Replace the model's entire architecture with a smaller one.
- ☐ c. Only train the last layer of the model.
- ☐ d. Increase the number of parameters in the model.

**Question 7**

Complete

Mark 1.00 out of 1.00

Adaptive Fine-Tuning is useful because it:

- ☒ a. Adapts the learning rate and fine-tuning parameters based on the specific task and dataset.
- ☐ b. Increases the model's size by adding new layers.
- ☐ c. Trains the model from scratch without using pre-trained weights.
- ☐ d. Reduces the need for checkpoints.

**Question 8**

Complete

Mark 1.00 out of 1.00

In a distributed training setup, what does the "node rank" parameter represent?

- ☐ a. The number of GPUs available in each machine.
- ☐ b. The optimizer's initial learning rate.
- ☒ c. The specific identifier for each machine in the distributed system.
- ☐ d. The type of model being used.

**Question 9**

Complete

Mark 1.00 out of 1.00

Which of the following is an advantage of using LoRA in fine-tuning large models like GPT-2?

- ☐ a. It modifies only the bias parameters in each layer.
- ☐ b. It increases the overall model size, improving accuracy.
- ☐ c. It trains the model from scratch on a large dataset.
- ☒ d. It allows for efficient fine-tuning by only adapting low-rank parameters, saving memory and computation.

**Question 10**

Complete

Mark 1.00 out of 1.00

What is the purpose of saving model snapshots or checkpoints in distributed training?

- ☐ a. To reduce memory usage.
- ☐ b. To speed up training on each GPU.
- ☒ c. To allow resuming training if interrupted.
- ☐ d. To increase the model's accuracy.