EE 599: Systems for Machine Learning
Final Year Project - Phase 3

Efficient Fine-tuning of LLM on a Single GPU

Sheik Dhawood Ashfaq Asick Ali
Sam Devavaram Jebaraj

Implementation:

- Mixed Precision Training:
  Utilized torch.cuda.amp for MP training. This technique takes advantage of Tensor cores on GPUs to accelerate training by using half-precision floating points (ie. FP16 instead of FP32) for certain parts of the network, reducing memory usage significantly.

- LoRA Integration:
  Used LoRA (Low-Rank Adaptation) into the model architecture. It works by inserting a smaller number of new weights into the model and only these are trained, this can help improve performance and efficiency.

  Integrated LoRA into the model architecture by replacing the linear transformations (wq, wk, wv) with custom Linear layers that include LoRA functionality. This linear function was used from that of Huggingface. Modified the fine-tuning script to freeze all model parameters initially and then selectively unfreeze LoRA parameters for training.

- Gradient Accumulation:
  Implemented gradient accumulation, which allows for accumulating gradients over multiple batches before performing a parameter update. Here the batch size is fixed as eight.

Testing Results:
After training Llama2 7B model with 200 samples from the Alpaca dataset. All possible combinations were tested as shown in Table 2. Found that when LoRA was turned off, all test cases failed due to GPU memory issues, even with high-end GPU in CARC like NVIDIA A100 80 GB. Recorded metrics such as loss, peak memory usage, and runtime for each test case.

GA	**OFF**				**ON**			
MP	**OFF**		**ON**		**OFF**		**ON**	
LoRA	**OFF**	**ON**	**OFF**	**ON**	**OFF**	**ON**	**OFF**	**ON**
Peak Mem (MB)	X	30151.6	X	41590.64	X	30199.60	X	41638.64
Runtime (s)	X	279.13	X	107.21	X	278.7	X	109.38

Table 2: System performance measurement

Prompt 1: Best Hiking place in Los Angeles for sunset is"
Prompt 2: A brief message congratulating the team on the launch:
Hi everyone!
I just

GA off LoRa On Amp off
Trainable params: 12,582,912 || all params: 6,751,391,744 || trainable%: 0.19
Average loss: 0.9242087
Peak GPU memory usage: 30151.60 MB
Total change in Memory: 111450036.19 MB
Computation: 279.13 secs

Prompt 1:
Best Hiking place in Los Angeles for sunset is
> Echo Mountain.
Echo Mountain is a mountain located in the San Gabriel Mountains in Los Angeles County, California. It is located on the north side of Echo Canyon, west of Glendora, and east of Azusa.
Echo Mountain is a mountain located in the San Gabriel Mountains in
Prompt 2:
A brief message congratulating the team on the launch:
Hi everyone,
I just
> want to say how proud I am of all of you.
The launch of the new website has been a success. I'm so excited about the future of this project. Congratulations on a job well done!
Best

Inference Time: 227.60134 secs

GA off LoRa On Amp On
Trainable params: 12,582,912 || all params: 6,751,391,744 || trainable%: 0.19
Average loss: 0.9250231
Peak GPU memory usage: 41590.64 MB
Computation: 107.21 secs

Prompt 1:
Best Hiking place in Los Angeles for sunset is
> Echo Mountain.
Echo Mountain is a 5,850-foot (1,780 m) summit in the San Gabriel Mountains, in Los Angeles County, California. It is located within the Angeles National Forest, about 15 miles (24 km) northwest of downtown Pasadena. The mountain offers stunning panoramic views of the surrounding landscape, making it an ideal spot to watch the sunset.

Prompt 2:
A brief message congratulating the team on the launch:
Hi everyone,
I just want to say how proud I am of all of you. The launch of the new website has been a success. I'm so excited about the future of this project. Congratulations on a job well done!
Best
Inference Time: 39.9389 secs

GA On Lora On Amp off
Trainable params: 12,582,912 || all params: 6,751,391,744 || trainable%: 0.19
Average loss: 0.1601637
Peak GPU memory usage: 30199.60 MB
Total change in Memory: 110646029.95 MB
Computation: 278.70 secs

Prompt 1:
Best Hiking place in Los Angeles for sunset is
> Echo Park Lake.
Echo Park Lake offers a serene setting to enjoy the sunset in Los Angeles. Surrounded by lush greenery and dotted with picturesque lotus flowers, the lake provides a tranquil escape from the bustle of the city. It's a perfect spot for a leisurely stroll or a relaxing picnic as you watch the sun dip below the horizon.

Prompt 2:
A brief message congratulating the team on the launch:
Hi everyone,
I just want to say how proud I am of all of you. The launch of the website was a great success. I'm looking forward to the next milestone and thank you for all your hard work.
Best wishes,
The Team

Inference Time: 231.5229 secs

GA On LoRa On Amp On
Trainable params: 12,582,912 || all params: 6,751,391,744 || trainable%: 0.19
Average loss: 0.1603995
Peak GPU memory usage: 41638.64 MB
Computation: 109.38 secs

Best Hiking place in Los Angeles for sunset is
> Echo Park Lake.

I love the view of Echo Park Lake. Itâ ™s a beautiful place to go for a walk and take a break from the city. There are also plenty of places to sit and relax. I recommend taking a picnic lunch or dinner and spending some time enjoying the

A brief message congratulating the team on the launch:

Hi everyone,
I just want to say how proud I am of all of you.
The launch of the web site was a great success.
I'm looking forward to the next milestone and thank you for all your hard work.
Best wishes,
The

Inference time: 33.045 secs

Table 1 - System Performance Analysis :

- Grad. Accumulation: Doesn't change the memory footprint of parameters, activations, gradients, or the optimizer state. It simply accumulates gradients over multiple mini-batches before performing a weight update, so the computation also remains the same. As we can infer from Table 2.
- Grad. Checkpoint: Reduces the memory usage of activations at the cost of increased computation. As it is recomputing intermediate activation during the backward pass it does extra computation.
- Mixed Precision: Reduces the memory footprint of parameters, activations, gradients, and the optimizer state by using a mix of float16 and float32 data types. Doing so it also cuts down in wall-clock time for computation.
- LoRA: It's a parameter-efficient fine-tuning technique that adds trainable low-rank decomposition matrices to different layers of a neural network, then freezes the network's remaining parameters. Hence respective trends are observed.

		Grad. Accumulation	Grad. Checkpoint	Mixed Precision	LoRA
Memory	parameter	-	-	↓	↑
	gradient	-	↓	↓	↓
	activation	-	-	↓	↓
	optimizer state	-	-	↓	↓
Computation		-	↑	↓	↓