Training Report

Run ID: b648aef2144645d2958c32257b01f4e5
Status: FINISHED
Start: 2026-02-05 05:18:11.233000
End: 2026-02-05 05:48:47.537000
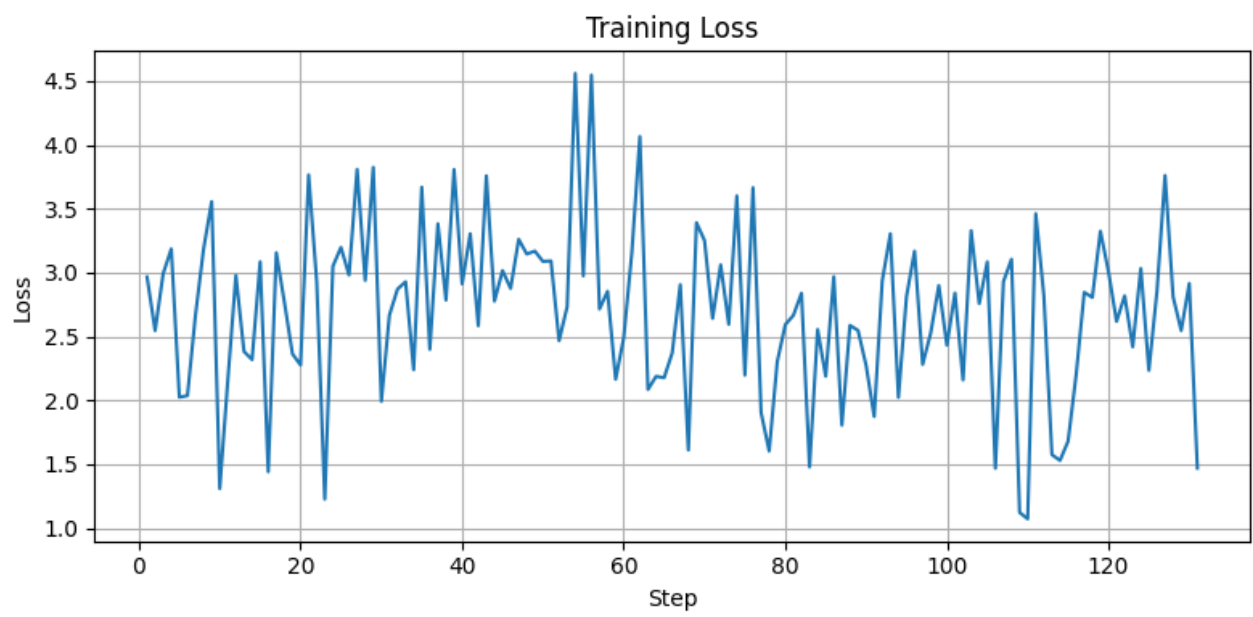**Duration (sec):** 1836.304

Parameters

Parameter	Value	Description
base_model_path	/mnt/c/Users/operator/ emtac/models/llm/ mistral-7b-instruct	
dataset_hash	c0b17d026fd5ff6d63d4 b754a464862dca5ba09 27cf9e0e9f382f1d582f 5891f	
dataset_path	/mnt/c/Users/operator/ PycharmProjects/ gpu_train/ dataset_output/ qna_training__complete /v001/ qna_training__complete .jsonl	
dataset_size	527	
dataset_version	v001	
device_type	cuda	
effective_batch_size	4	
enable_lora	True	Whether LoRA (Low-Rank Adaptation) fine-tuning is enabled
enable_mlflow	False	
epochs	1	Total number of training epochs
extra_args	[]	
extra_env	{}	
fsdp_auto_wrap_policy	transformer	
fsdp_min_num_params	100000000	
fsdp_sharding_strategy	FULL_SHARD	How model parameters are split across GPUs (FULL_SHARD = maximum memory savings)
gpu_device_index	0	
gpu_name	NVIDIA GeForce RTX 5090	
gpu_total_memory_mb	32607.0	
gradient_accumulation_step s	4	Number of steps to accumulate gradients before updating (simulates larger batch)
job_name	mistral7b_mini_lora	

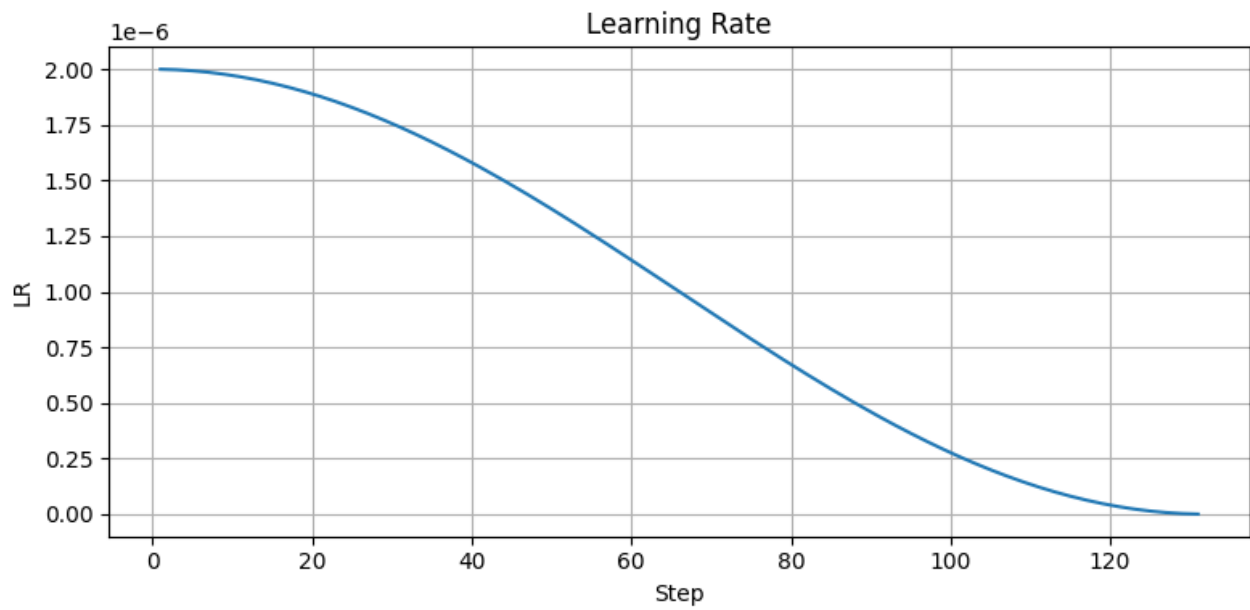|		
learning_rate	2e-06	Step size for weight updates during training (lower = more cautious learning)
max_seq_length	1024	Maximum input sequence length in tokens (longer = more context but more memory)
mixed_precision	bf16	Precision format for calculations (bf16 = faster training with lower memory usage)
nnodes	1	Number of machines/nodes used for distributed training
node_rank	0	Identifier for this node in multi-node training (0 = primary node)
num_train_epochs	1.0	Number of complete passes through the training dataset
output_dir	/mnt/c/Users/operator/ PycharmProjects/ gpu_train/prod_out/ mistral7b_mini	
per_device_eval_batch_size	1	
per_device_train_batch_size	1	Number of training samples processed together per device
rdzv_backend	c10d	
resume_strict	True	
seed	42	Random seed for reproducible training runs
trainer	sft_fsdp	
training_policy	{'register_model': True, 'registry_name': 'emtac_mistral_sft', 'dataset_hash': 'c0b17d026fd5ff6d63d 4b754a464862dca5ba0 927cf9e0e9f382f1d582 f5891f', 'dataset_name': 'qna_training__complet e', 'dataset_version': 'v001'}	
train_data_path	/mnt/c/Users/operator/ PycharmProjects/ gpu_train/ dataset_output/ qna_training__complete /v001/ qna_training__complete .jsonl	
warmup_ratio	0.03	Fraction of training for learning rate warmup (stabilizes early training)
weight_decay	0.0	Regularization to prevent overfitting (penalizes large weights)
world_size	1	

Metrics

Metric	Value	Description
gpu_memory_total_mb	32607.0	GPU utilization / memory telemetry collected during training
gpu_memory_used_mb	32277.84375	GPU utilization / memory telemetry collected during training
gpu_memory_utilizatio n_pct	98.990535007820 4	GPU utilization / memory telemetry collected during training
gpu_peak_memory_allo cated_mb	55406.950683593 75	GPU utilization / memory telemetry collected during training
gpu_power_watts	153.47	GPU utilization / memory telemetry collected during training
gpu_temperature_c	41.0	GPU utilization / memory telemetry collected during training
gpu_torch_max_memor y_allocated_mb	55406.950683593 75	GPU utilization / memory telemetry collected during training
gpu_torch_memory_allo cated_mb	27718.255859375	GPU utilization / memory telemetry collected during training
gpu_torch_memory_res erved_mb	56412.0	GPU utilization / memory telemetry collected during training
gpu_utilization_pct	100.0	GPU utilization / memory telemetry collected during training
grad_norm	8.625	Gradient norm magnitude (stability indicator)
lr	2.8320530687087 055e-10	Learning rate at final training step
train_loss	1.4687834978103 638	Final training loss value (lower is better)

Loss Curve

How to read this chart: The loss curve shows how well the model is learning over time. The y-axis shows the loss value (lower is better), and the x-axis shows training steps. A good training run shows a general downward trend, indicating the model is improving. Some fluctuation is normal, but large spikes or plateaus may indicate learning rate issues, data quality problems, or the need for more training steps.

Learning Rate

How to read this chart: The learning rate controls how much the model adjusts its weights during training. This chart shows how the learning rate changes over training steps. Most schedules start with a warmup phase (gradual increase), then decay (gradual decrease) toward the end. The warmup helps stabilize early training, while the decay allows the model to fine-tune and converge. A schedule that decays too quickly may prevent the model from learning effectively.