**Classification Fine tuning**

**Model : Gpt2-small (124M)**

Data: Standard spam ham data set of 6000K

Pruned 800 Spam vs 800 Ham

**A100 Gpu 40 Gb**

**Accuracy (Before Training)**

Training accuracy: 46.25%

Validation accuracy: 45.00%

Test accuracy: 48.75%

**Train 1**

Parameters Trained : 7 million (**Last transformer block and layer norm)**

Training

………….

Ep 5 (Step 000550): Train loss 0.207, Val loss 0.143

Ep 5 (Step 000600): Train loss 0.083, Val loss 0.074

Training accuracy: 100.00% | Validation accuracy: 97.50%

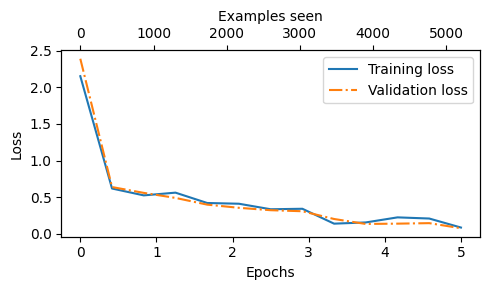
Training completed in 0.28 minutes.

After Training

Training accuracy: 98.75%

Validation accuracy: 98.75%

**Test accuracy: 95.00%**



**Train 2**

Parameters Trained : 14 Million (**Last two transformers)**

………….

Ep 5 (Step 000550): Train loss 0.190, Val loss 0.046

Ep 5 (Step 000600): Train loss 0.028, Val loss 0.034

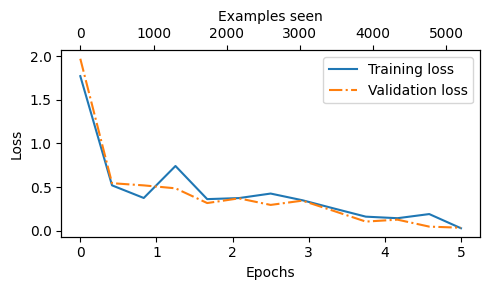
Training accuracy: 100.00% | Validation accuracy: 100.00%

Training completed in 0.29 minutes.

Training accuracy: 98.75%

Validation accuracy: 100.00%

Test accuracy: 96.25%



**Train 3**

Parameters Trained : 124Million **(Entire model)**

………………..

Ep 5 (Step 000550): Train loss 0.058, Val loss 0.024

Ep 5 (Step 000600): Train loss 0.098, Val loss 0.021

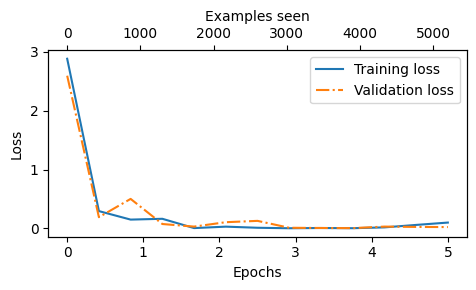
Training accuracy: 100.00% | Validation accuracy: 97.50%

Training completed in 0.67 minutes.

Training accuracy: 100.00%

Validation accuracy: 98.75%

Test accuracy: 96.25%

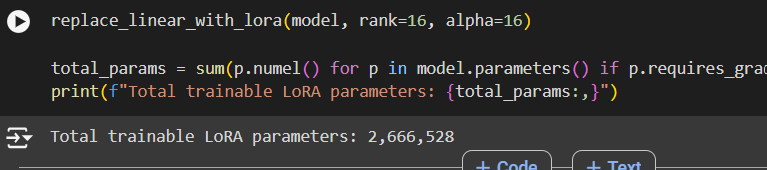


**Train 4**

**Parameter efficient training with LoRA for spam ham classification**

Total trainable parameters for GPT Medium : 124Million

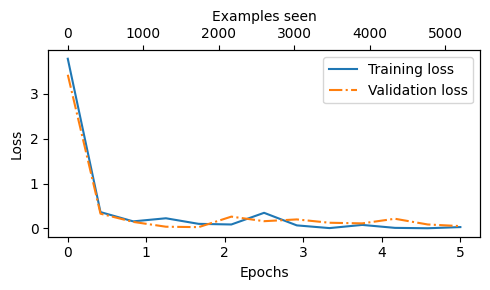
After Introducing Lora Layers : 2.6Million Parameters

****

Training accuracy: 99.90%

Validation accuracy: 97.32%

Test accuracy: 97.00%

****

**Instruction Fine Tuning**

**Train 1**

**Model : GPT MEDIUM 340 Million**

Data : Alpaca Dataset (50k)

Epoch : 1

Training 44k

A100 40GB Max Usage is around 28GB

Training: 44201

Validation: 2601

Test: 5200

**Response Before Instruction Finetuning**

Below is an instruction that describes a task. Write a response that appropriately completes the request.

### Instruction:

Explain how using transitional words help in writing

### Input:

"<noinput>"

### Output:

"<nooutput>"

### Result:

"<success>"

### Example:

"<noinput>”

Training loss: 3.2639075756072997

Validation loss: 3.41228928565979

**Training**

……………………..

Ep 1 (Step 011040): Train loss 1.133, Val loss 1.516

Ep 1 (Step 011045): Train loss 1.186, Val loss 1.518

Training completed in 45.71 minutes.

**Response After Training**

Below is an instruction that describes a task. Write a response that appropriately completes the request.

### Instruction:

Explain how using transitional words help in writing

### Input:

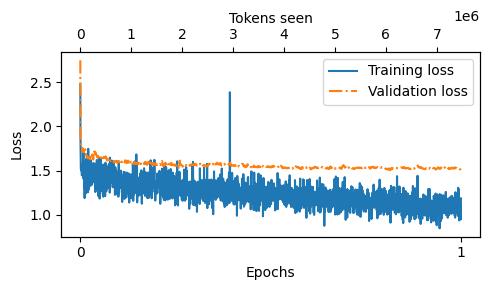
"<noinput>"

### Response:

Transitional words help to create a more structured and organized writing style. They help to create a more consistent structure and make it easier to remember.

Responses

instruction-data-with-response\_Final.json



**Train 2**

**Model : LLama 3 Instruction Fine Tuning 8 Billion Parameters**

Alpacca Dataset 1000 data

Google - TPU

Approx Time : 300Minutes

**Response**

**Before Training**

Output text:

Below is an instruction that describes a task. Write a response that appropriately completes the request.

Training

### Instruction:

Summarize the given poem in 3 lines.

### Input:

The poem reads:

The trees in my yard are singing a song,

The breeze is blowing and gently moving along,

The crickets chirp and the birds join in song.

### Output:

The trees in my yard are singing a song, the breeze is blowing and gently moving along, the cr

**After Training**

Below is an instruction that describes a task. Write a response that appropriately completes the request.

### Instruction:

Explain how using transitional words help in writing

### Input:

"<noinput>"

### Response:

Transitional words help to create a more structured and organized writing style. They help to create a more consistent structure and make it easier to remember.

