```
--- Starting Experiment: some_candidate_tinystories ---
Results will be saved in:
/content/drive/MyDrive/SoME_Experiments/some_candidate_tinystories

--- Part 2: Data Preparation & Configuration ---
Training universal BPE tokenizer...

README.md:
  1.06k/?  [00:00<00:00,  122kB/s]
Tokenizer loaded with vocab size: 8192

Tokenizing dataset: roneneldan/TinyStories...

data/train-00000-of-00004-2d5a1467fff108(…):  100%
  249M/249M  [00:03<00:00,  3.72MB/s]
data/train-00001-of-00004-5852b56a2bd28f(…):  100%
  248M/248M  [00:01<00:00,  316MB/s]
data/train-00002-of-00004-a26307300439e9(…):  100%
  246M/246M  [00:01<00:00,  299MB/s]
data/train-00003-of-00004-d243063613e5a0(…):  100%
  248M/248M  [00:01<00:00,  297MB/s]
data/validation-00000-of-00001-869c898b5(…):  100%
  9.99M/9.99M  [00:00<00:00,  14.0MB/s]
Generating  train  split:  100%
  2119719/2119719  [00:06<00:00,  333547.56  examples/s]
Generating  validation  split:  100%
  21990/21990  [00:00<00:00,  293475.96  examples/s]
Map  (num_proc=12):  100%
  20000/20000  [00:03<00:00,  9488.03  examples/s]
Map  (num_proc=12):  100%
  5000/5000  [00:01<00:00,  4543.58  examples/s]
Train dataset size (subset): 20000
Validation dataset size (subset): 5000
Using 6 workers for DataLoader.

--- Part 3: Model Definition ---

--- Part 4: Training, Evaluation, and Metrics ---
Total Parameters: 1631.63M
Trainable Parameters: 18.92M (1.16%)

/tmp/ipython-input-881065511.py:102: FutureWarning: `torch.cuda.amp.GradScaler(args...)` is
deprecated. Please use `torch.amp.GradScaler('cuda', args...)` instead.
  scaler = torch.cuda.amp.GradScaler()
```

```
--- Epoch 1/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:110: FutureWarning: `torch.cuda.amp.autocast(args...)` is
deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast():
/usr/local/lib/python3.12/dist-packages/torch/optim/lr_scheduler.py:192: UserWarning: Detected
call of `lr_scheduler.step()` before `optimizer.step()`. In PyTorch 1.1.0 and later, you should call
them in the opposite order: `optimizer.step()` before `lr_scheduler.step()`.  Failure to do this will
result in PyTorch skipping the first value of the learning rate schedule. See more details at
https://pytorch.org/docs/stable/optim.html#how-to-adjust-learning-rate
  warnings.warn(
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
```

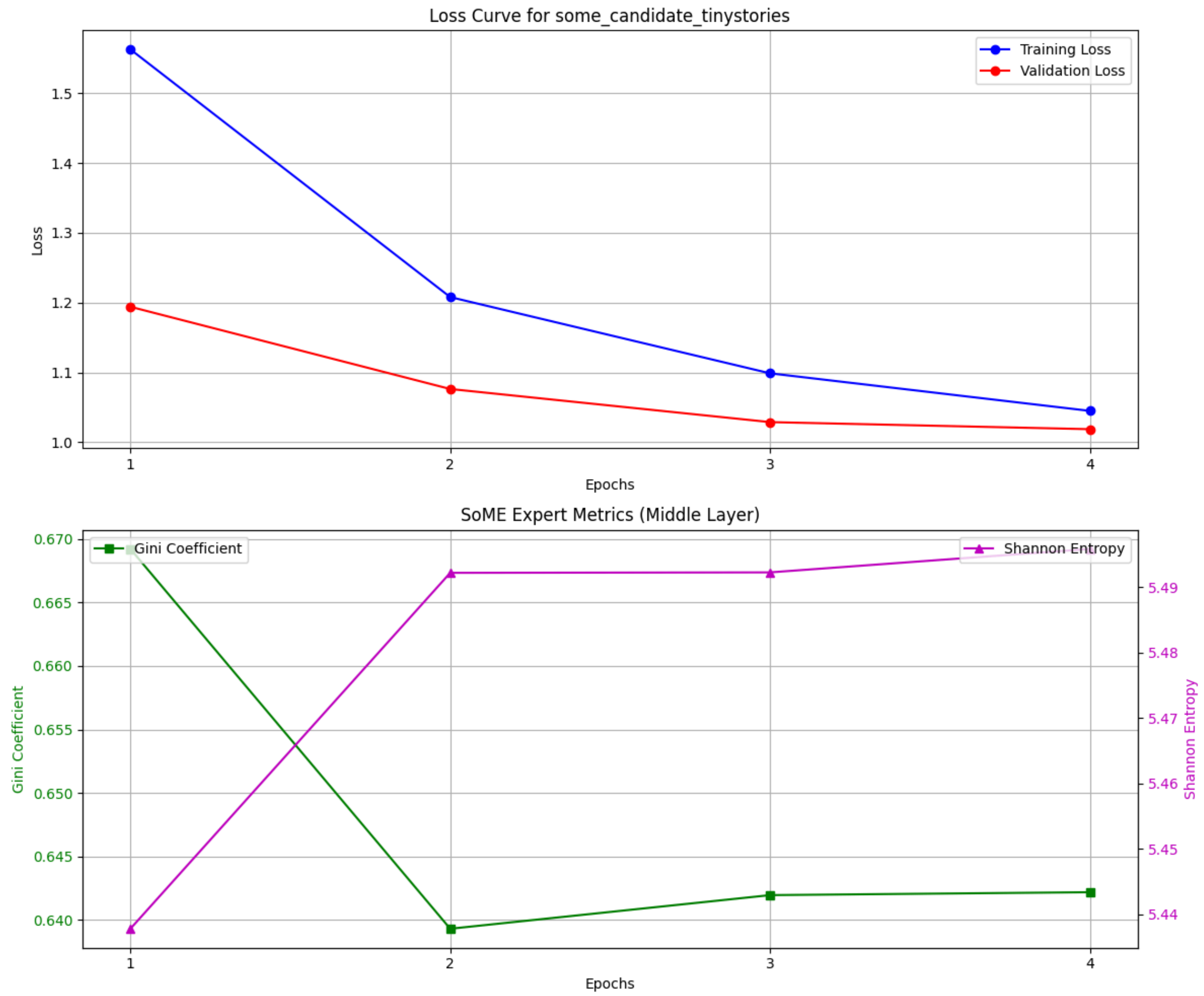
```
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:137: FutureWarning: `torch.cuda.amp.autocast(args...)` is
deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast():

Epoch 1: Train Loss = 1.5628, Val Loss = 1.1941, Val Perplexity = 3.30
  Middle Layer Expert Metrics: Gini = 0.669, Entropy = 5.438
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tinystories/best_model_some_can
didate_tinystories.pth

--- Epoch 2/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
```

```
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 2: Train Loss = 1.2080, Val Loss = 1.0763, Val Perplexity = 2.93
  Middle Layer Expert Metrics: Gini = 0.639, Entropy = 5.492
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tinystories/best_model_some_can
didate_tinystories.pth

--- Epoch 3/4 ---
```

```
Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
```

```
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 3: Train Loss = 1.0988, Val Loss = 1.0289, Val Perplexity = 2.80
  Middle Layer Expert Metrics: Gini = 0.642, Entropy = 5.492
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tinystories/best_model_some_can
didate_tinystories.pth

--- Epoch 4/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 4: Train Loss = 1.0450, Val Loss = 1.0187, Val Perplexity = 2.77
  Middle Layer Expert Metrics: Gini = 0.642, Entropy = 5.496
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tinystories/best_model_some_candidate_tinystories.pth

--- Training Complete for some_candidate_tinystories ---
Best Validation Loss: 1.0187
Final Validation Perplexity: 2.77

Metrics plot saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tinystories/metrics_some_candidate_tinystories.png
```


Loss Curve for some_candidate_tinystories

SoME Expert Metrics (Middle Layer)

```
--- Starting Experiment: transformer_baseline_tinystories ---
Results will be saved in:
/content/drive/MyDrive/SoME_Experiments/transformer_baseline_tinystories

--- Part 2: Data Preparation & Configuration ---
Training universal BPE tokenizer...

README.md:
  1.06k/?  [00:00<00:00,  102kB/s]
Tokenizer loaded with vocab size: 8192

Tokenizing dataset: roneneldan/TinyStories...

data/train-00000-of-00004-2d5a1467fff108(…):  100%
  249M/249M  [00:01<00:00,  103MB/s]
data/train-00001-of-00004-5852b56a2bd28f(…):  100%
```

```
  248M/248M  [00:00<00:00,  323MB/s]
data/train-00002-of-00004-a26307300439e9(…):  100%
  246M/246M  [00:01<00:00,  315MB/s]
data/train-00003-of-00004-d243063613e5a0(…):  100%
  248M/248M  [00:00<00:00,  329MB/s]
data/validation-00000-of-00001-869c898b5(…):  100%
  9.99M/9.99M  [00:00<00:00,  22.2MB/s]
Generating  train  split:  100%
  2119719/2119719  [00:06<00:00,  327225.66  examples/s]
Generating  validation  split:  100%
  21990/21990  [00:00<00:00,  302487.73  examples/s]
Map  (num_proc=12):  100%
  20000/20000  [00:03<00:00,  6683.09  examples/s]
Map  (num_proc=12):  100%
  5000/5000  [00:01<00:00,  418.88  examples/s]
Train dataset size (subset): 20000
Validation dataset size (subset): 5000
Using 6 workers for DataLoader.

--- Part 3: Model Definition ---

--- Part 4: Training, Evaluation, and Metrics ---
Total Parameters: 33.62M
Trainable Parameters: 33.62M (100.00%)

/tmp/ipython-input-881065511.py:102: FutureWarning: `torch.cuda.amp.GradScaler(args...)` is
deprecated. Please use `torch.amp.GradScaler('cuda', args...)` instead.
  scaler = torch.cuda.amp.GradScaler()

--- Epoch 1/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:110: FutureWarning: `torch.cuda.amp.autocast(args...)` is
deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast():
/usr/local/lib/python3.12/dist-packages/torch/optim/lr_scheduler.py:192: UserWarning: Detected
call of `lr_scheduler.step()` before `optimizer.step()`. In PyTorch 1.1.0 and later, you should call
them in the opposite order: `optimizer.step()` before `lr_scheduler.step()`.  Failure to do this will
result in PyTorch skipping the first value of the learning rate schedule. See more details at
https://pytorch.org/docs/stable/optim.html#how-to-adjust-learning-rate
  warnings.warn(
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

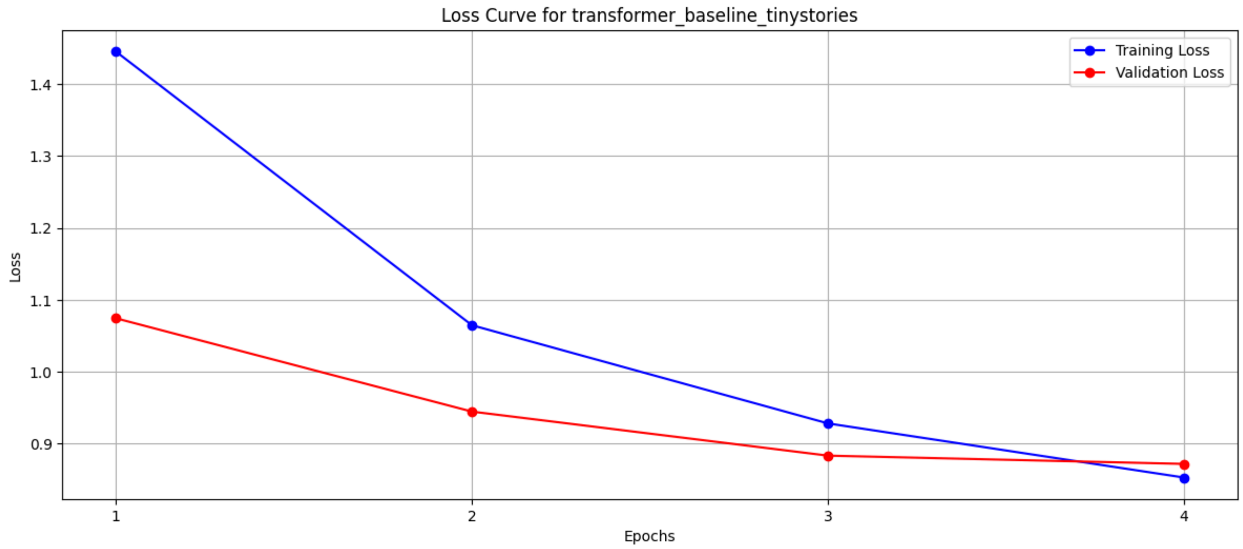
```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:137: FutureWarning: `torch.cuda.amp.autocast(args...)` is
deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast():

Epoch 1: Train Loss = 1.4453, Val Loss = 1.0744, Val Perplexity = 2.93
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/transformer_baseline_tinystories/best_model_transf
ormer_baseline_tinystories.pth

--- Epoch 2/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 2: Train Loss = 1.0648, Val Loss = 0.9447, Val Perplexity = 2.57
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/transformer_baseline_tinystories/best_model_transf
ormer_baseline_tinystories.pth

--- Epoch 3/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 3: Train Loss = 0.9283, Val Loss = 0.8835, Val Perplexity = 2.42
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/transformer_baseline_tinystories/best_model_transf
ormer_baseline_tinystories.pth

--- Epoch 4/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
```

```
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 4: Train Loss = 0.8530, Val Loss = 0.8720, Val Perplexity = 2.39
  -> New best model saved to /content/drive/MyDrive/SoME_Experiments/transformer_baseline_tinystories/best_model_transformer_baseline_tinystories.pth

--- Training Complete for transformer_baseline_tinystories ---
Best Validation Loss: 0.8720
Final Validation Perplexity: 2.39

Metrics plot saved to /content/drive/MyDrive/SoME_Experiments/transformer_baseline_tinystories/metrics_transformer_baseline_tinystories.png
```



```
--- Starting Experiment: some_candidate_tiny_textbooks ---
Results will be saved in:
/content/drive/MyDrive/SoME_Experiments/some_candidate_tiny_textbooks

--- Part 2: Data Preparation & Configuration ---
Training universal BPE tokenizer...

README.md:
  1.06k/?  [00:00<00:00,  99.8kB/s]
Tokenizer loaded with vocab size: 8192
```

```
Tokenizing dataset: nampdn-ai/tiny-textbooks...

README.md:  100%
  6.55k/6.55k  [00:00<00:00,  664kB/s]
train-00000-of-00001.parquet:  100%
  857M/857M  [00:34<00:00,  23.1MB/s]
test-00000-of-00001.parquet:  100%
  44.9M/44.9M  [00:02<00:00,  21.0MB/s]
Generating  train  split:  100%
  399000/399000  [00:04<00:00,  94576.94  examples/s]
Generating  test  split:  100%
  21000/21000  [00:00<00:00,  85298.97  examples/s]
Map  (num_proc=12):  100%
  20000/20000  [00:07<00:00,  4399.34  examples/s]
Map  (num_proc=12):  100%
  5000/5000  [00:01<00:00,  4630.88  examples/s]
Train dataset size (subset): 20000
Validation dataset size (subset): 5000
Using 6 workers for DataLoader.

--- Part 3: Model Definition ---

--- Part 4: Training, Evaluation, and Metrics ---
Total Parameters: 1631.63M
Trainable Parameters: 18.92M (1.16%)

/tmp/ipython-input-881065511.py:102: FutureWarning: `torch.cuda.amp.GradScaler(args...)` is
deprecated. Please use `torch.amp.GradScaler('cuda', args...)` instead.
  scaler = torch.cuda.amp.GradScaler()

--- Epoch 1/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
```

```
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:110: FutureWarning: `torch.cuda.amp.autocast(args...)` is
deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast():
/usr/local/lib/python3.12/dist-packages/torch/optim/lr_scheduler.py:192: UserWarning: Detected
call of `lr_scheduler.step()` before `optimizer.step()`. In PyTorch 1.1.0 and later, you should call
them in the opposite order: `optimizer.step()` before `lr_scheduler.step()`.  Failure to do this will
result in PyTorch skipping the first value of the learning rate schedule. See more details at
https://pytorch.org/docs/stable/optim.html#how-to-adjust-learning-rate
  warnings.warn(
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

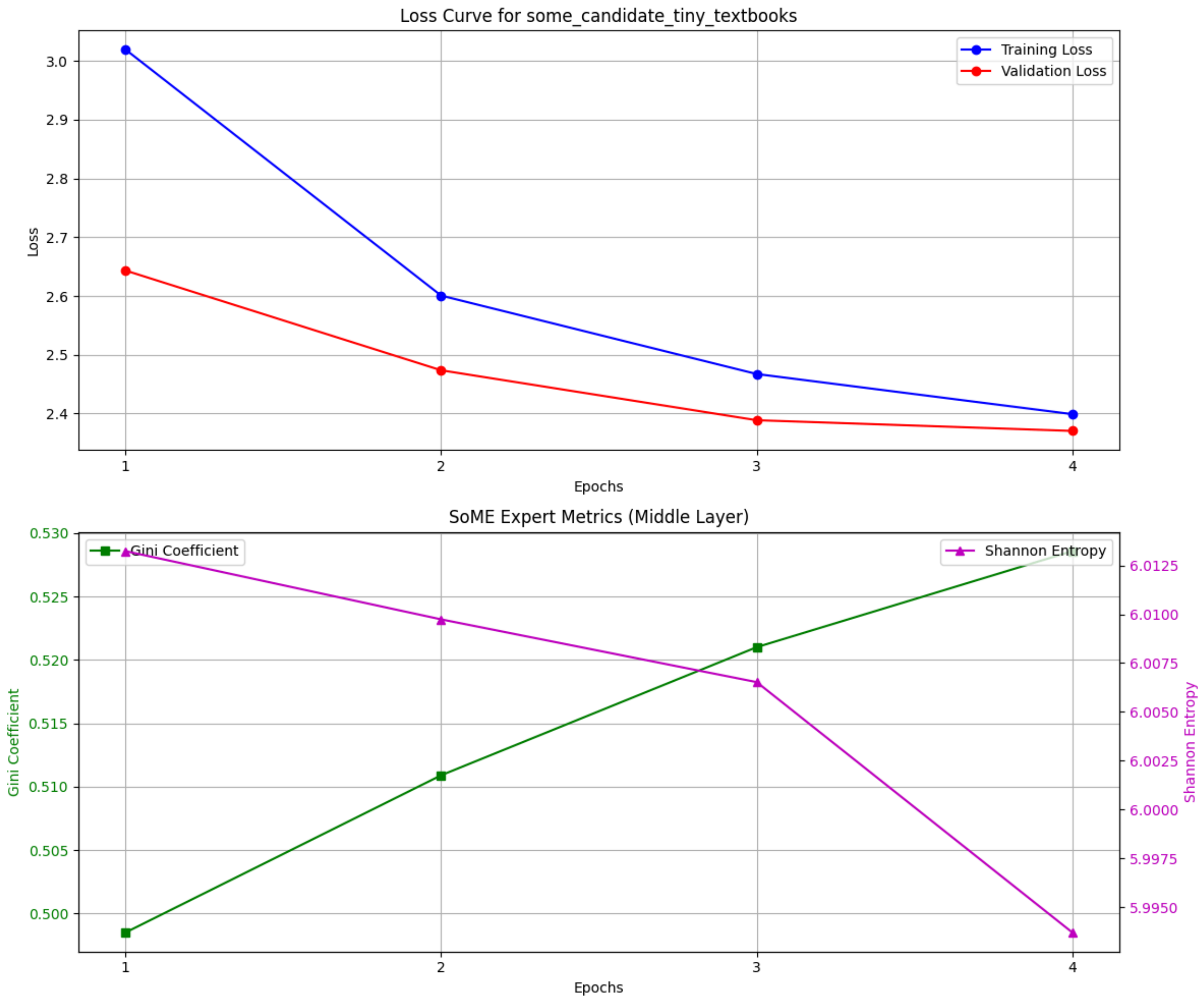
```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:137: FutureWarning: `torch.cuda.amp.autocast(args...)` is
deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast():

Epoch 1: Train Loss = 3.0192, Val Loss = 2.6433, Val Perplexity = 14.06
  Middle Layer Expert Metrics: Gini = 0.499, Entropy = 6.013
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tiny_textbooks/best_model_some
_candidate_tiny_textbooks.pth

--- Epoch 2/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
```

```
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 2: Train Loss = 2.6006, Val Loss = 2.4736, Val Perplexity = 11.87
  Middle Layer Expert Metrics: Gini = 0.511, Entropy = 6.010
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tiny_textbooks/best_model_some
_candidate_tiny_textbooks.pth

--- Epoch 3/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 3: Train Loss = 2.4673, Val Loss = 2.3887, Val Perplexity = 10.90
  Middle Layer Expert Metrics: Gini = 0.521, Entropy = 6.007
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tiny_textbooks/best_model_some
_candidate_tiny_textbooks.pth
```

```
--- Epoch 4/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 4: Train Loss = 2.3990, Val Loss = 2.3705, Val Perplexity = 10.70
  Middle Layer Expert Metrics: Gini = 0.529, Entropy = 5.994
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tiny_textbooks/best_model_some
_candidate_tiny_textbooks.pth

--- Training Complete for some_candidate_tiny_textbooks ---
Best Validation Loss: 2.3705
Final Validation Perplexity: 10.70

Metrics plot saved to
/content/drive/MyDrive/SoME_Experiments/some_candidate_tiny_textbooks/metrics_some_can
didate_tiny_textbooks.png
```



```
--- Starting Experiment: transformer_baseline_tiny_textbooks ---
Results will be saved in:
/content/drive/MyDrive/SoME_Experiments/transformer_baseline_tiny_textbooks

--- Part 2: Data Preparation & Configuration ---
Training universal BPE tokenizer...

README.md:
  1.06k/?  [00:00<00:00,  116kB/s]
Tokenizer loaded with vocab size: 8192

Tokenizing dataset: nampdn-ai/tiny-textbooks...

README.md:  100%
  6.55k/6.55k  [00:00<00:00,  764kB/s]
train-00000-of-00001.parquet:  100%
```

```
  857M/857M  [00:02<00:00,  331MB/s]
test-00000-of-00001.parquet:  100%
  44.9M/44.9M  [00:00<00:00,  168MB/s]
Generating  train  split:  100%
  399000/399000  [00:03<00:00,  101526.25  examples/s]
Generating  test  split:  100%
  21000/21000  [00:00<00:00,  87415.94  examples/s]
Map  (num_proc=12):  100%
  20000/20000  [00:03<00:00,  8190.23  examples/s]
Map  (num_proc=12):  100%
  5000/5000  [00:01<00:00,  4593.21  examples/s]
Train dataset size (subset): 20000
Validation dataset size (subset): 5000
Using 6 workers for DataLoader.

--- Part 3: Model Definition ---

--- Part 4: Training, Evaluation, and Metrics ---
Total Parameters: 33.62M
Trainable Parameters: 33.62M (100.00%)

/tmp/ipython-input-881065511.py:102: FutureWarning: `torch.cuda.amp.GradScaler(args...)` is
deprecated. Please use `torch.amp.GradScaler('cuda', args...)` instead.
  scaler = torch.cuda.amp.GradScaler()

--- Epoch 1/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:110: FutureWarning: `torch.cuda.amp.autocast(args...)` is
deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with torch.cuda.amp.autocast():
/usr/local/lib/python3.12/dist-packages/torch/optim/lr_scheduler.py:192: UserWarning: Detected
call of `lr_scheduler.step()` before `optimizer.step()`. In PyTorch 1.1.0 and later, you should call
them in the opposite order: `optimizer.step()` before `lr_scheduler.step()`.  Failure to do this will
result in PyTorch skipping the first value of the learning rate schedule. See more details at
https://pytorch.org/docs/stable/optim.html#how-to-adjust-learning-rate
  warnings.warn(
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:137: FutureWarning: `torch.cuda.amp.autocast(args...)` is
deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
```

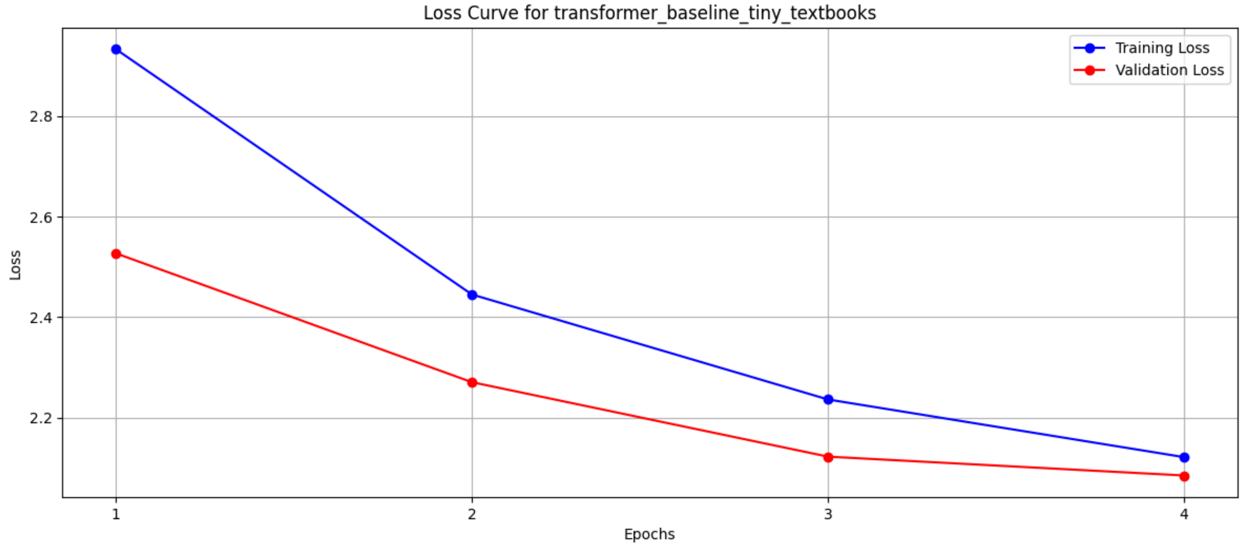
```
  with torch.cuda.amp.autocast():

Epoch 1: Train Loss = 2.9327, Val Loss = 2.5267, Val Perplexity = 12.51
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/transformer_baseline_tiny_textbooks/best_model_tr
ansformer_baseline_tiny_textbooks.pth

--- Epoch 2/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 2: Train Loss = 2.4448, Val Loss = 2.2704, Val Perplexity = 9.68
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/transformer_baseline_tiny_textbooks/best_model_tr
ansformer_baseline_tiny_textbooks.pth

--- Epoch 3/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
```

```
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 3: Train Loss = 2.2360, Val Loss = 2.1223, Val Perplexity = 8.35
  -> New best model saved to
/content/drive/MyDrive/SoME_Experiments/transformer_baseline_tiny_textbooks/best_model_tr
ansformer_baseline_tiny_textbooks.pth

--- Epoch 4/4 ---

Training:   0%|          | 0/833 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14: UserWarning:
To copy construct from a tensor, it is recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
```

```
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
Evaluating:   0%|          | 0/208 [00:00<?, ?it/s]/tmp/ipython-input-881065511.py:14:
UserWarning: To copy construct from a tensor, it is recommended to use
sourceTensor.detach().clone() or sourceTensor.detach().clone().requires_grad_(True), rather
than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])
/tmp/ipython-input-881065511.py:14: UserWarning: To copy construct from a tensor, it is
recommended to use sourceTensor.detach().clone() or
sourceTensor.detach().clone().requires_grad_(True), rather than torch.tensor(sourceTensor).
  inputs = torch.tensor(item['input_ids'])

Epoch 4: Train Loss = 2.1213, Val Loss = 2.0849, Val Perplexity = 8.04
```

```
  -> New best model saved to /content/drive/MyDrive/SoME_Experiments/transformer_baseline_tiny_textbooks/best_model_transformer_baseline_tiny_textbooks.pth

--- Training Complete for transformer_baseline_tiny_textbooks ---
Best Validation Loss: 2.0849
Final Validation Perplexity: 8.04

Metrics plot saved to /content/drive/MyDrive/SoME_Experiments/transformer_baseline_tiny_textbooks/metrics_transformer_baseline_tiny_textbooks.png
```


Loss Curve for transformer_baseline_tiny_textbooks
Training Loss
Validation Loss
Loss
Epochs