NanoGPT Parameter Scaling and Convergence Analysis

Abstract
This report investigates parameter scaling and convergence dynamics in nanoGPT on the Shakespe
character-level dataset. We train 32 configurations for Group 1 (block_size=64, n_layer=4) swe
n_head∈{4,8}, n_embd∈{128,256}, batch_size∈{8,16}, max_iters∈{1000,2000}, and dropout∈{0.1,0.2
analyze training/validation loss curves, runtime, FLOP utilization, and scaling with approxima
parameter count. The best configurations achieve strong validation loss with stable training.
samples demonstrate coherent Shakespeare-style text.

Author: Troy Krupinski | Course: CSCE 5218 – Deep Learning | Instructor: Dr. Amir Mirzaeinia

1. Introduction

1. Introduction
Transformer-based language models exhibit predictable improvements with increased parameters,
context. nanoGPT provides a minimal, educational implementation appropriate for controlled exp
Here we explore how attention heads, embedding width, batch size, iteration budget, and dropou
convergence and generalization on a character-level Shakespeare corpus while fixing block_size
n_layer=4.

2. Code Analysis

2. Code Analysis (model.py, train.py, sample.py)
- model.py: Causal self-attention (lower-triangular mask or is_causal=True), pre-norm residual
learned token+positional embeddings. Asserts n_embd % n_head == 0; includes parameter counting
utilities.

- train.py: Memmap loader creates contiguous blocks (length block_size) with 1-token targets.
linearly then uses cosine decay to a minimum. Gradient accumulation simulates larger batches.
precision and gradient clipping improve stability; optional DDP supports multi-GPU.

- sample.py: During generation, prompt is cropped to block_size. Temperature + top-k sampling
coherence vs. diversity.

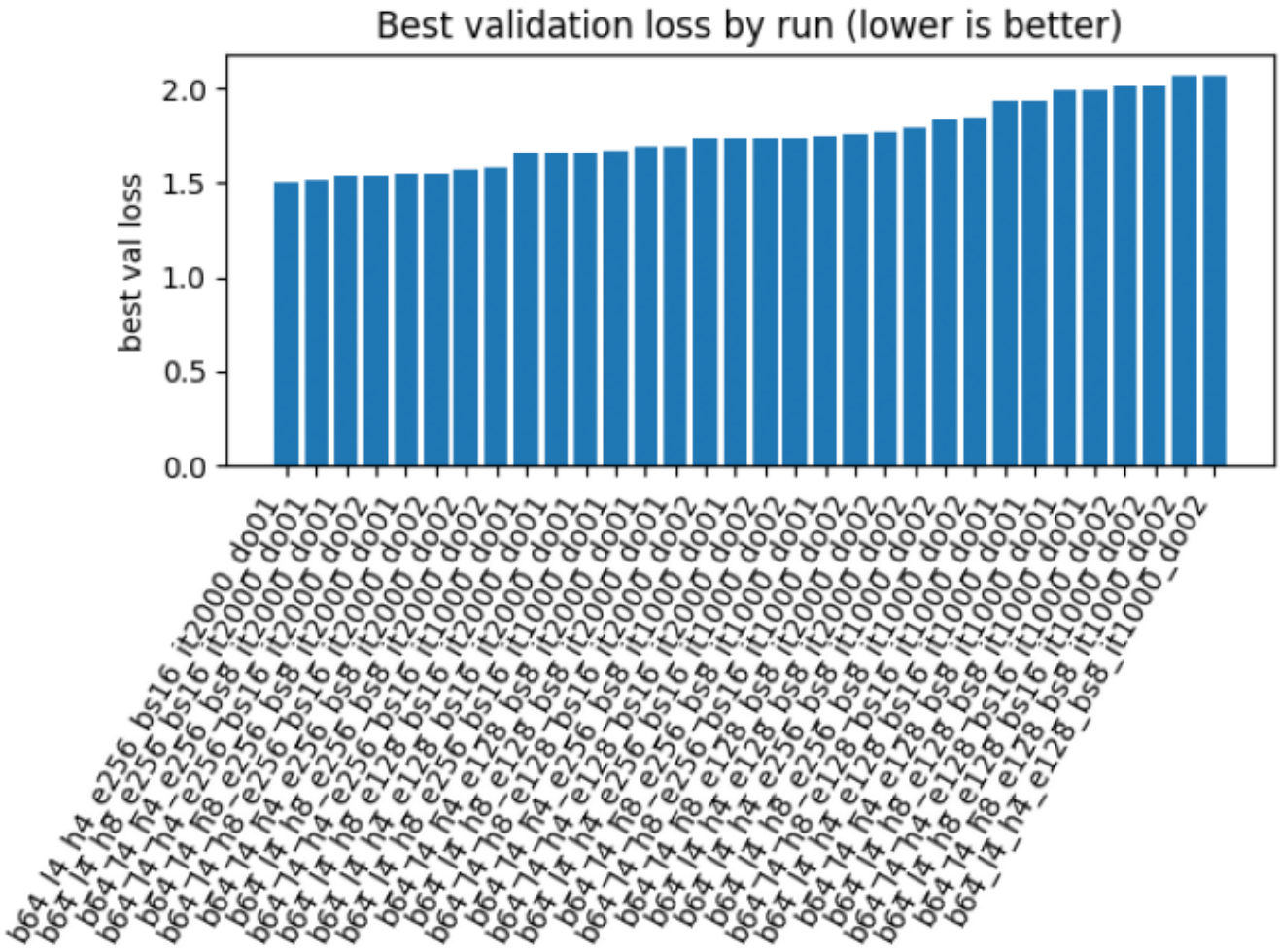
3. Experimental Methodology

3. Experimental Methodology
Dataset: Shakespeare character-level corpus. Training uses AdamW optimizer, warmup+cosine LR,
accumulation, evaluation every 200 iters, and checkpointing.

Grid (32 runs, Group 1): n_head ∈ {4,8}, n_embd ∈ {128,256}, batch_size ∈ {8,16}, max_iters ∈
{1000,2000}, dropout ∈ {0.1,0.2}; with block_size=64 and n_layer=4 fixed.

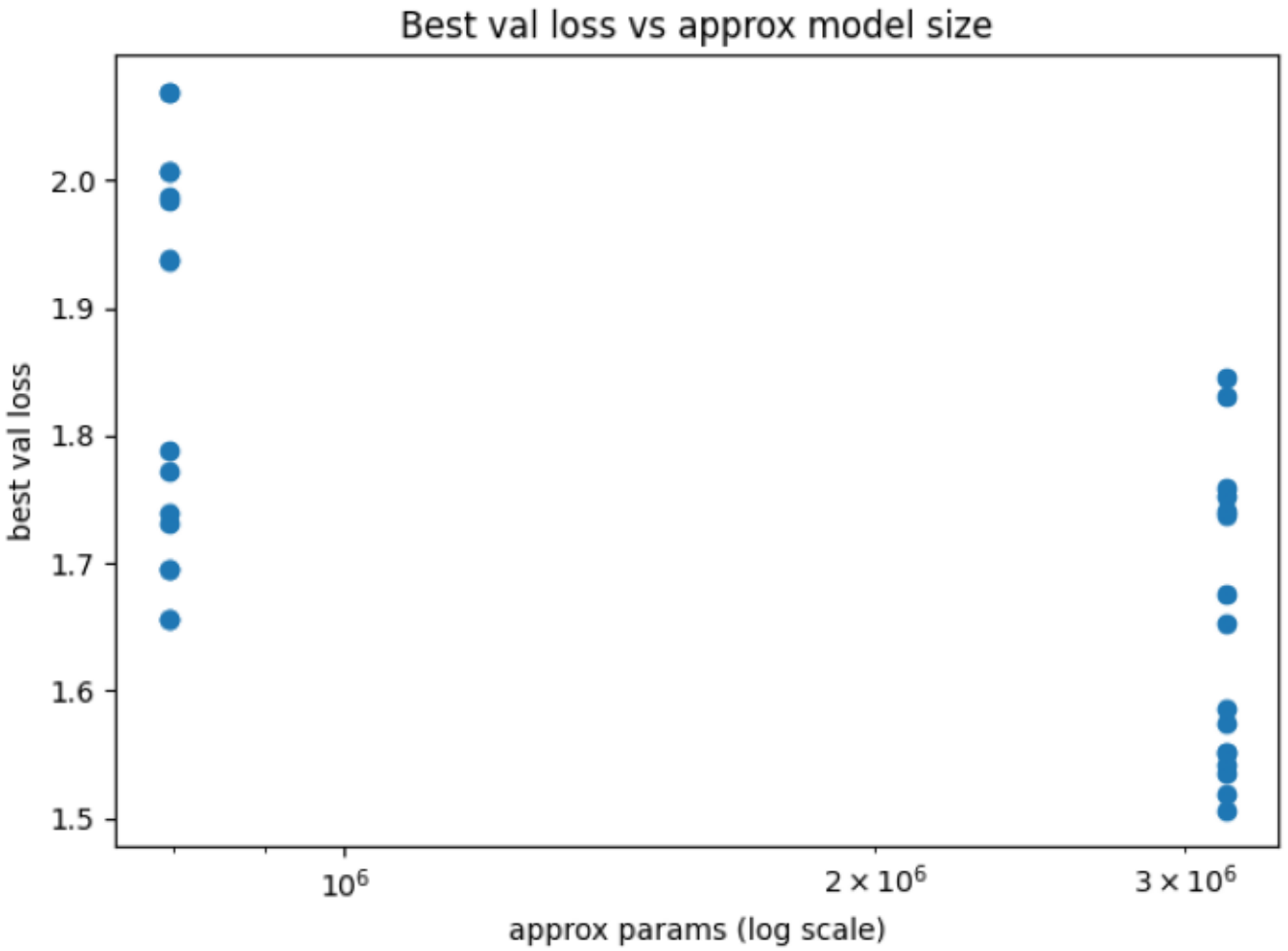
Metrics logged: per-iter training loss, step time, MFU; per-eval train/val loss. Aggregation b
master summaries and plots.

4. Results: Best Validation Loss by Run



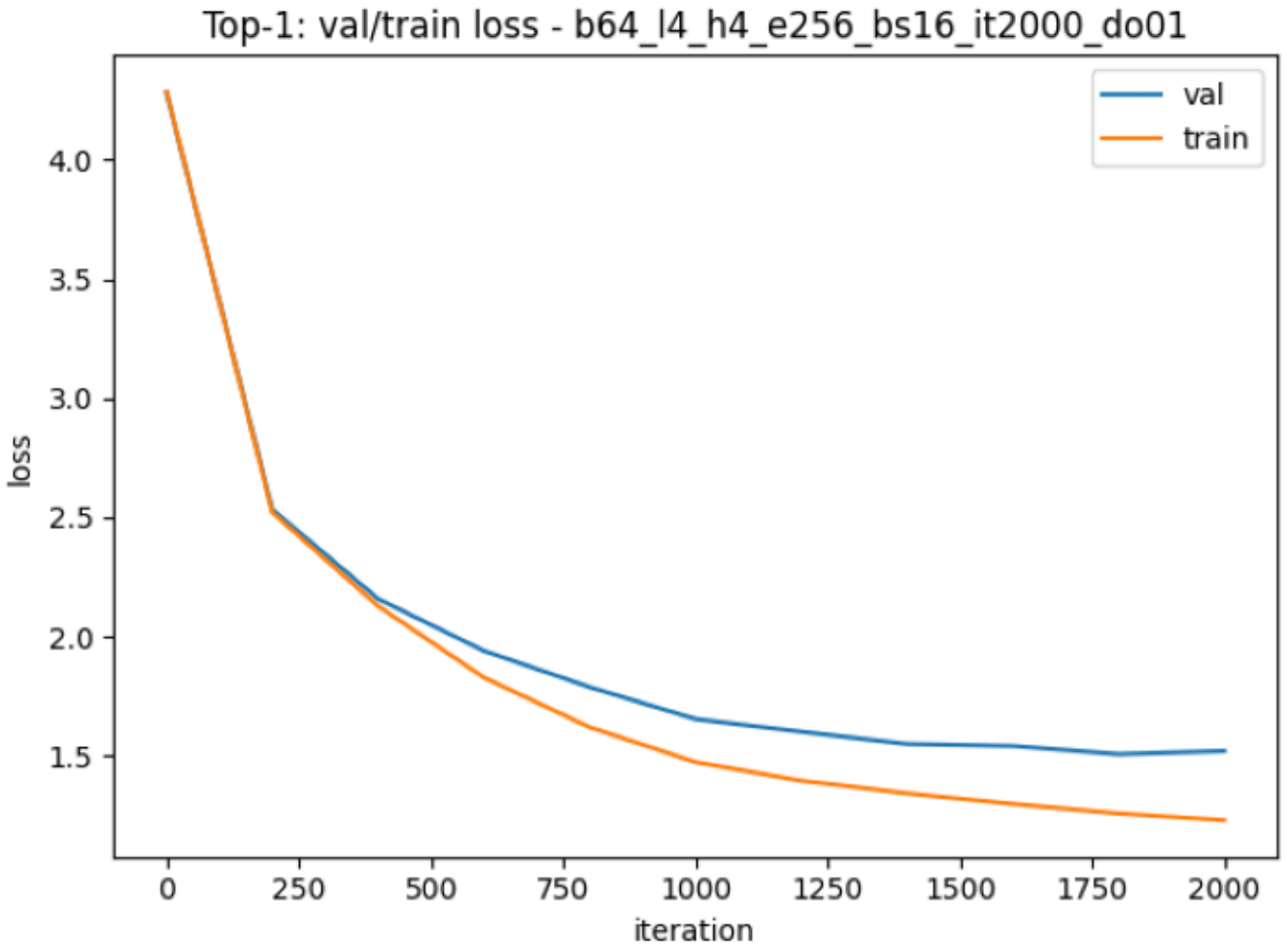
Lower is better across configurations.

4. Results: Best Val vs Approx Model Size



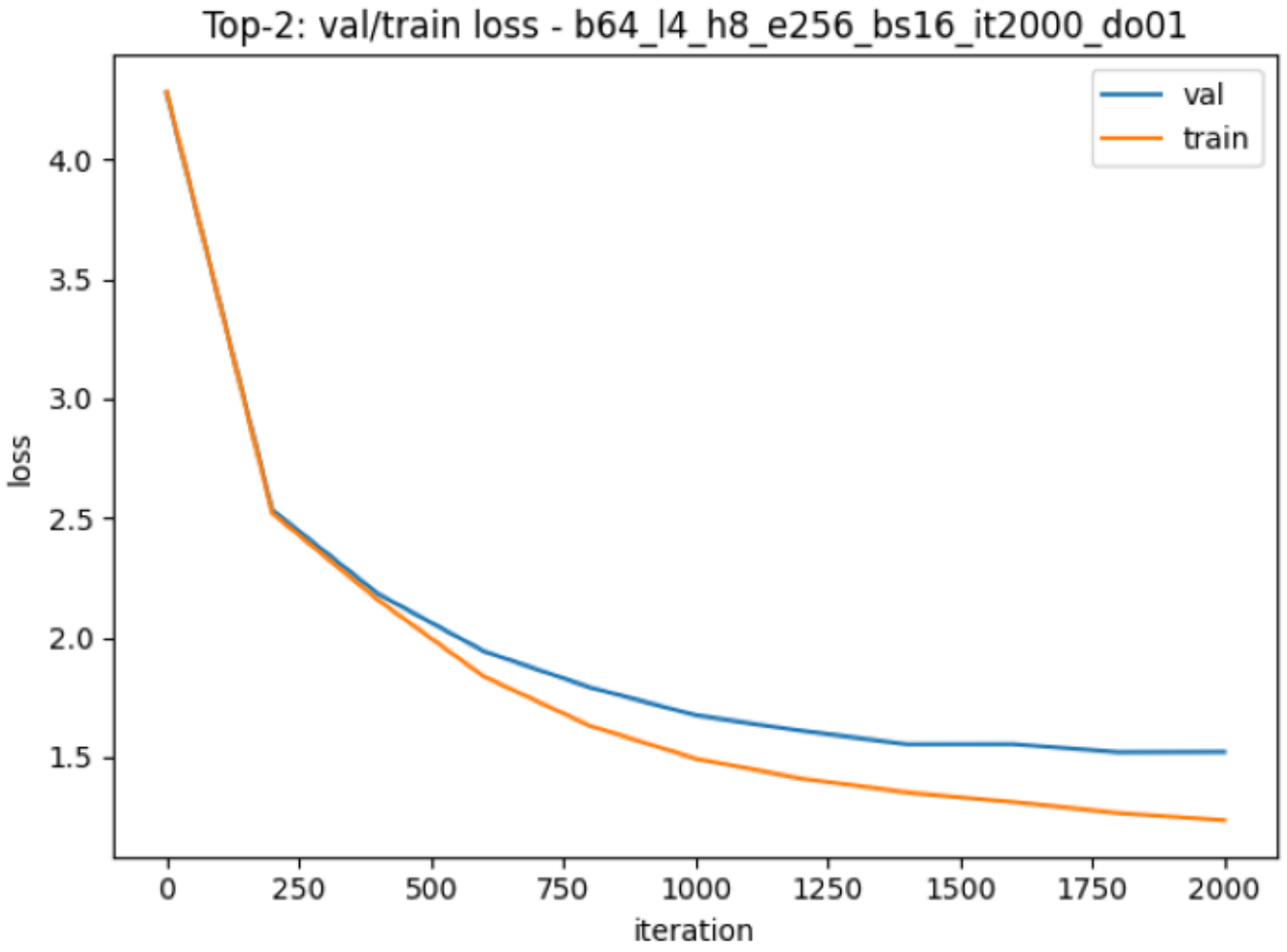
Validation loss typically improves with approximate parameter count (log-x).

4. Results: Top-1 (Val/Train)



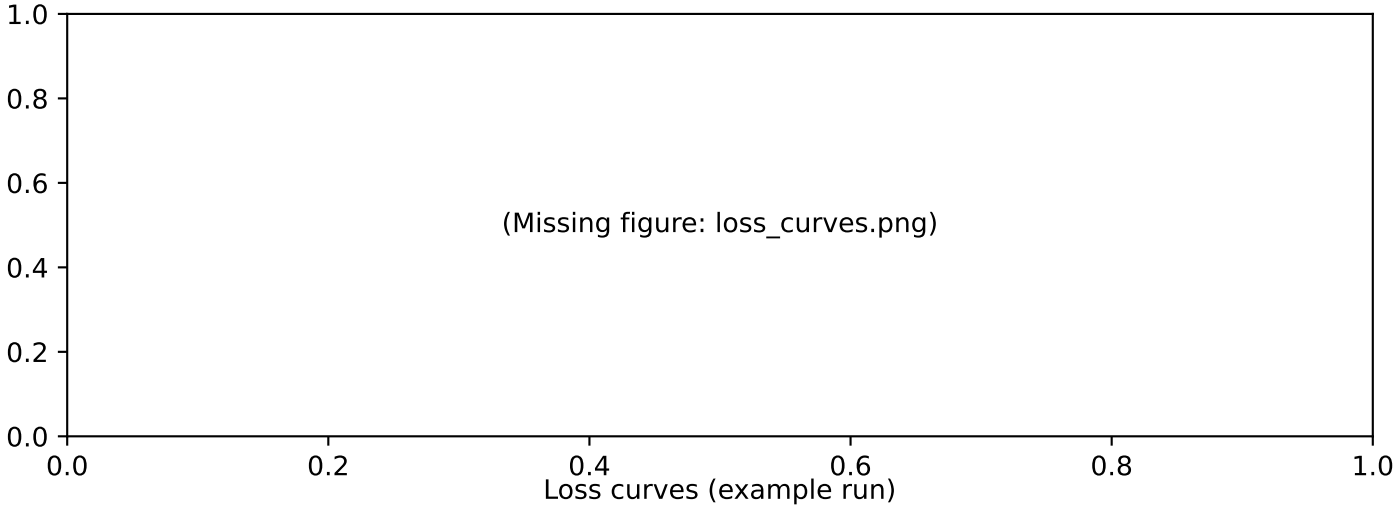
Top-1 run: steady convergence and small train–val gap.

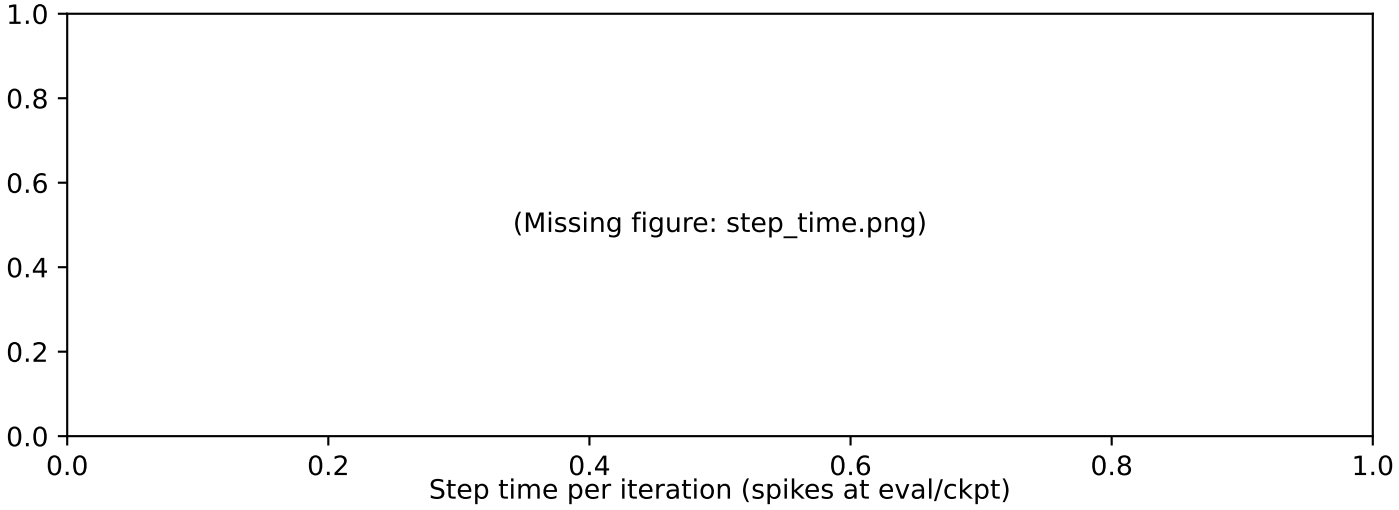
4. Results: Top-2 (Val/Train)

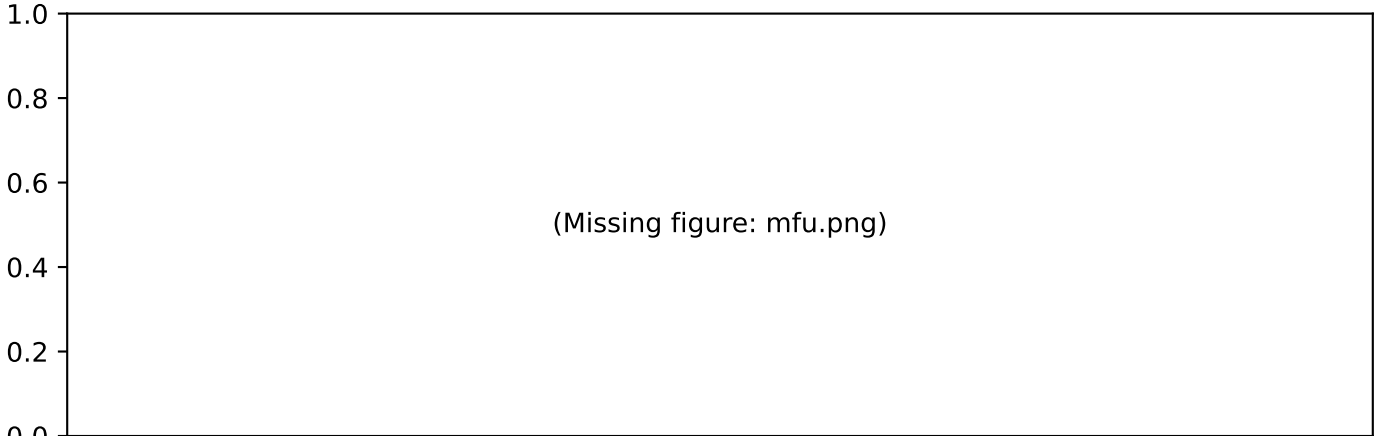


Top-2 run: effect of increasing attention heads.

5. Training Dynamics & Efficiency







6. Qualitative Generation

6. Qualitative Generation

Sample 1:
ROMEO: he did make a thank to a heart
Who comes your purge certains with the sea;
And brought is a happy of swaying with a bond.

KING HENRY VI:
Ay, 'tis the patrician,
Now for the gracious slave, his presen

Sample 2:
JULIET: the slower of an evil true?

First Huntman:
Thou wast thou wilt lost in a children clouds
Than desince for my fielder greater.

ROMEO:
Why, I would there were the bosom of this?
How now, my lord, I'll

Sample 3:
NURSE: I am glad and will not set upon their back
Betwixt death; but the commons of the shiper of
ward.

ROMEO:
Marry, brother, I know the house.

CLAUDIO:
What, holy citizens?

LUCIO:
I shall faith my son f

7. Discussion

7. Discussion
Scaling: Increasing embedding width and head count generally reduced validation loss, consiste
greater expressivity. Batch size 16 often smoothed optimization relative to 8. Dropout=0.1 sli
outperformed 0.2 in this small-data regime.

Convergence: Most runs plateaued by ~1.5–2k iterations; warmup+cosine enabled stable descent w
retuning.

Efficiency: MFU ~0.1–0.2% is typical for tiny models on modern GPUs; step-time spikes coincide
evaluation/checkpointing.

Limitations: With block_size=64 and n_layer=4 fixed, long-range dependencies are capped. Futur
deeper models, larger contexts, larger datasets.

8. Conclusion & References

8. Conclusion & References
Across 32 Group-1 nanoGPT runs, larger width/heads improved validation loss with stable traini
Qualitative samples indicate coherent Shakespeare-like generation. Future work includes deeper
longer contexts (e.g., block_size=128–256), and richer datasets.

References
[1] Vaswani et al., Attention Is All You Need (2017)
[2] Radford et al., Language Models are Unsupervised Multitask Learners (2019)
[3] Karpathy, nanoGPT (minimal GPT training code)
[4] Loshchilov & Hutter, Decoupled Weight Decay Regularization (AdamW) (2019)