Mainrun assessment — Harvey Houlahan

*Final validation loss of **1.1754 / 33.0%** improvement from baseline 1.7533 **(Δ** 0.57786)*

- *≈40.0M params, 7 epochs, seed 1337, fixed evaluate()*

ABSTRACT

With theoretical Machine Learning (ML) knowledge but limited hands-on experience training transformers, I adopted an iterative approach to improving the baseline training run; observing loss behaviour, reasoning about its cause, adjusting a variable, and reproducing. This paradigm represents Feynman's formulation of the scientific method, where each experiment becomes a hypothesis for the model's behavior, confirmed or refuted by the validation curve. Furthermore, minimal prior exposure to this problem space allowed me to identify and adopt techniques directly relevant to the task without established assumptions; muon optimisation, warmup-stable-decay scheduling, rotary position embeddings, ReLU² activations, and value-residual connections, each adapted iteratively to the particular constraints of the Mainrun Assessment. The result was a 33.0% reduction in validation loss from 1.7533 to 1.1754 within the fixed 7-epoch ceiling.

APPROACH AND METHOD

The baseline trains a 27.17M-parameter GPT-2 style transformer with Stochastic Gradient Descent (SGD) architecture reaching a final validation loss of 1.7533 at step 938. The fundamental behavior demonstrates slow but stable convergence: validation loss drops from 2.098 (step 1) to 1.774 (step 352), then improves by only ~0.021 over the remaining ~586 steps, indicating a long plateau and opposing the idea of model divergence or overfitting. Furthermore, training loss remains high throughout (9.7741 → 8.1794), reinforcing the notion that the model is underfitting i.e. failing to extract sufficient signal from the training data rather than memorising it. This is consistent with Goodfellow et al.'s characterisation of underfitting as a model's failure to achieve sufficiently low error on the training set (Goodfellow, Bengio & Courville, *Deep Learning*, MIT Press, 2016, p. 111).

Early ablations reinforce the hypothesis; warmup and weight-decay tuning (*Exp2 vs Exp3: 1.31947 → 1.31941*) and dropout tuning (*Exp4 vs Exp5: 1.27465 → 1.27442*) produced negligible gains, invalidating regularization as the operative constraint and shifting subsequent optimization toward increasing effective learning per step. SGD's uniform parameters are fundamentally limited in non-convex loss landscapes with heterogeneous gradient curvature and AdamW's per-parameter adaptive learning rates address this directly across the varying gradient scales of attention and MLP weights.

All experiments operated within the fixed framework constraints: epochs (7), seed (1337), dataset, validation fraction (0.10), and evaluate() unchanged throughout. Within this container, a single variable protocol was applied against the current best configuration, with full validation sweeps logged per run. Optimums were empirically highlighted, and regressions were retained in the ledger as experiments locating the boundaries.

RESULTS

Figure 1:

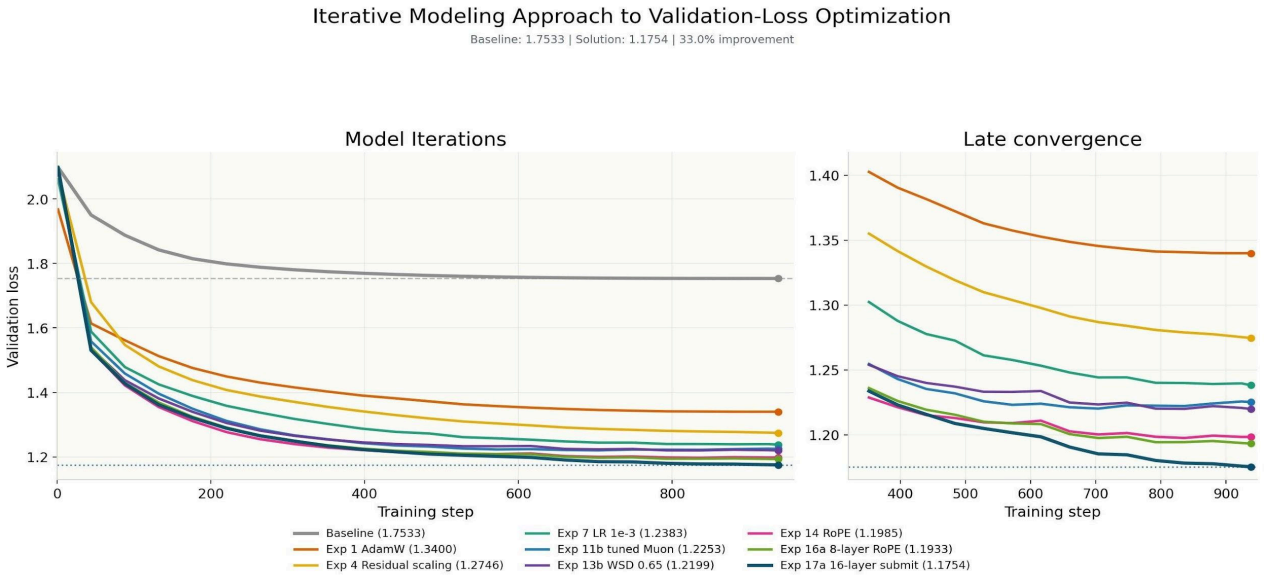

Figure 1. Validation loss across milestone configurations (left) with the late-convergence region magnified (right). Baseline plateaus near 1.75; AdamW alone collapses to 1.34; each subsequent technique stacks a smaller gain to the 1.1754 submission.

The complete experiment ledger is presented below; regression and overshoot experiments are highlighted in red (served to empirically bracket the performance boundaries of each technique) and the final submission configuration, Exp 17a, is denoted in green.

Table 1:

#	Experiment	Key change	Val Loss	Δ base
0	Baseline	SGD lr=6e-3, CosineAnnealingLR, 27.2M param	1.7533	—
1	AdamW	SGD → AdamW lr=3e-4, betas (0.9, 0.95)	1.3400	−0.4133
2	Warmup+cosine	+ 10% linear warmup, manual cosine decay	1.3195	−0.4338
3	Weight decay	+ wd=0.1 on 2D params only	1.3194	−0.4339
4	Residual scaling	+ GPT-2 init 1/√(2·n_layer) on projections	1.2746	−0.4787
5	Dropout 0.05	Dropout 0.1 → 0.05	1.2744	−0.4789
6	Grad accum × 2	+ SDPA + gradient accumulation (regression)	1.3156	−0.4377
6b	SDPA only	Grad accum back to 1 → isolated cause	1.2744	−0.4789
7	LR 1e-3	Pushed LR above 3e-4 (underfitting)	1.2383	−0.5150
7b	LR 1.5e-3	Overshoot → bracket optimum at 1e-3	1.2457	−0.5076
8	Deeper/narrower	8 layers, d_model 416 (23.4M)	1.2352	−0.5181
9	WD ablation	wd 0.1 → 0.0 at optimal LR (control)	1.2388	−0.5145
11	Muon (CPU)	Muon lr=0.02, AdamW lr=3e-3 → plateaued	1.3285	−0.4248
11b	Muon tuned	Muon lr=0.005, AdamW lr=1e-3	1.2253	−0.5280

#	Experiment	Key change	Val Loss	Δ base
11c	Muon lr=0.004	Confirms 0.005 is optimum (6-layer)	1.2350	−0.5183
12	Muon + RoPE (lr 0.002)	Muon lr=0.02 + RoPE → LR too high, RoPE effect masked	1.3326	−0.4207
13	WSD schedule	Cosine → warmup-stable-decay, cooldown 0.70	1.2209	−0.5324
13b	WSD cooldown .65	Earlier cooldown	1.2199	−0.5334
14	RoPE	Replaced learned positions with rotary embeddings	1.1985	−0.5548
15	QK-norm	Parameter-free RMS-norm on q/k (regression, dropped)	1.2178	−0.5355
16a	Depth re-test	8L/416 on the full stack	1.1933	−0.5600
16b	Muon lr=0.006	Re-tuned Muon LR for 8-layer depth	1.1879	−0.5654
16c	Muon lr=0.008	Overshoot → bracket 0.007 (8-layer)	1.1875	−0.5658
16d	Muon lr=0.007	8-layer optimum	1.1866	−0.5667
16e	ReLU² MLP	Squared-ReLU activation (Primer)	1.1835	−0.5698
16f	Value-residual	Learnable v-mix to first layer; n_head 8→4	1.1823	−0.5710
17a	16 layers	n_layer 8 → 16 (submission)	**1.1754**	**−0.5779**
17b	20 layers	n_layer 16 → 20 (longer run, not submitted)	1.1745	−0.5788

Table 1. Chronological experiment ledger; Red: regressions and boundary-bracketing overshoots. Green: final submission (Exp 17a).

OBSERVATIONS

1. Optimizer selection as the primary performance lever

- Exp 1: SGD → AdamW (lr=3e-4, betas 0.9/0.95) reduced validation loss from 1.7533 to 1.3400, accounting for the approx. 73% reduction in final validation loss submission.
- Transformers exhibit heterogeneous gradient scales across attention and MLP weights; AdamW's per-parameter adaptive rates address this limitation where SGD's uniform updates cannot.
- Every subsequent experiment extends on this notion, which serves as the foundation for all further optimisation.

2. Underfitting as the operative constraint

- Exp 2→3 (warmup and weight decay): 1.31947 → 1.31941; Exp 4→5 (dropout): 1.27465 → 1.27442 demonstrate negligible improvement, invalidating regularisation as the operative constraint.
- A model unresponsive to regularisation adjustments is underfitting rather than overfitting; the correct intervention is increased learning capacity per step, not further constraint.
- Exp 7 (AdamW LR 3e-4 → 1e-3): loss fell to 1.2383 with no instability; Exp 7b (LR 1.5e-3): overshot to 1.2457, bracketing the optimum empirically rather than assuming it from defaults.

3. **Controlled isolation of self-introduced regression**

- Exp 6 introduced PyTorch SDPA and gradient accumulation = 2 simultaneously, resulting in the regression of loss by 0.0412.
- Gradient accumulation halved the number of optimiser updates from 938 to 469 within the fixed 7-epoch budget, a reduction the larger effective batch size could not offset.
- Exp 6b reverted accumulation to 1 with SDPA retained: validation loss returned to exactly 1.2744, recovering Exp 5 output and confirming the regression was attributable to update frequency, not the attention kernel.

4. **Frontier optimizer implementation and constraint specific adaptations**

- Muon (Keller Jordan, nanoGPT speedrun literature) orthogonalises 2D weight-matrix updates through a 5 step Newton–Schulz iteration, focusing on per-step learning efficiency rather than indirectly through LR scaling.
- Exp. 11 (Muon lr=0.02) plateaus at 1.3285; the hyperparameters are calibrated for 124M-parameter GPU runs at larger batch sizes; the curve stalled rather leading to an LR mismatch diagnosis rather than architectural incompatibility.
- Exp. 11b (lr=0.005) achieved 1.2253 and Exp. 11c (lr=0.004) regressed to 1.2350; the result at the lower learning rate confirmed the residual degradation as a schedule problem rather than an LR magnitude issue leading the transition to warmup-stable-decay
- Exp. 13 (WSD, cooldown 0.65) eliminated the late-training rise; Exp. 14 (RoPE), 16e (ReLU²), and 16f (value-residual) contributed reductions of 0.0214, 0.0031, and 0.0012 respectively, each drawing from current pretraining literature and adapted to the assessment conditions.

5. **Depth, width, and the compute trade off**

- Increasing width (8L/512, ~10M additional params) reduced performance; additional parameters per layer exceeded what 938 optimizer steps could train, consistent with the step limitation interpretation of underfitting.
- These findings are not contradictory: width increases the per-layer parameter cost without adding representational depth, while additional layers compound sequential transformation quality at a lower per-step learning cost.
- Increasing depth from 8 to 16 layers improved performance; contingent on Muon, residual scaling, ReLU² (Exp. 16e, −0.0031) and value-residual (Exp 16f, −0.0012) first establishing stable signal propagation through the deeper stack as without this infrastructure, the additional depth would not have been trainable.
- Exp. 17a (16 layers) achieved 1.1754 loss at approximately 40M parameters and 9.1 minutes runtime, selected as the optimal configuration.
- Exp. 17b (20 layers) achieved 1.1745; minimal gain at substantially greater computational cost and therefore was not submitted (Table 1).
- Time complexity is unconstrained by the assessment rules; the depth increase is within scope, and the 16-layer configuration represents the practical ceiling given the diminishing returns observed beyond that point.

Figure 2:

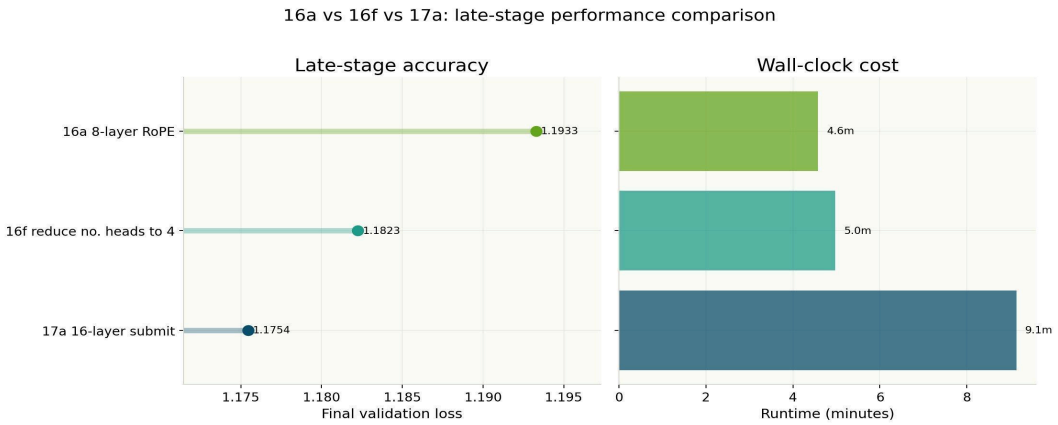

Figure 2. Late-stage accuracy versus wall-clock cost for the 8-layer RoPE (16a), 4-head value-residual (16f), and 16-layer submission (17a) configurations; the 16-layer model achieves its accuracy at roughly double the runtime.

CONSIDERATIONS

The ablations and regressions are kept as evidence for the design decisions.

Gradient accumulation (Exp 6)	+0.0412 regression; halved optimiser updates from 938 to 469 within the fixed budget.
Weight decay > 0 (Exp 3, 9)	Negligible movement; confirmed underfitting diagnosis and regularisation as not the operative constraint.
Lower dropout (Exp 5)	0.1 → 0.05 was −0.0002; further reinforces the underfitting hypothesis.
Muon at default lr=0.02 (Exp 11, 12)	Plateaued at 1.3285; hyperparameters mismatched at this batch size; re-tuned 4× lower to recover.
QK-norm, no learnable scale (Exp 15)	+0.0193 regression; parameter-free RMS-norm capped attention dynamic range without the learnable per-head gain required to recover it.
8L/512 width (scaling test)	Worse than 8L/416 at the same step budget; confirmed capacity was not bottleneck, rather optimizer steps
Overshoot brackets (7b, 11c, 16c)	Deliberate runs past each optimum confirming minima was located empirically.

Figure 3:

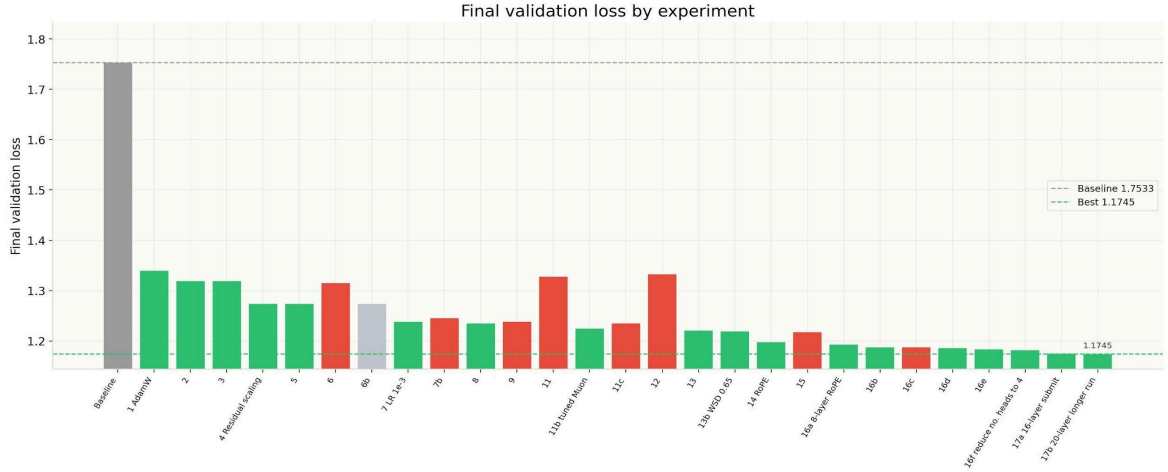

Figure 3. Final validation loss per run; green improved on the prior best, red bracketed a boundary, dashed line marks the best result.

FINAL CONFIGURATION

Table 2:

Architecture	16 layers, d_model 416, 4 heads (head_dim 104), block_size 128
Parameters	≈ 40.0M total → ~47% MORE than the 27.2M baseline
Vocabulary	16k byte-level BPE tokenizer
Optimizer	Muon (lr=0.007, momentum=0.95, wd=0.01, 5-step Newton–Schulz) on 2D hidden matrices
	AdamW (lr=1e-3, betas=0.9/0.95, wd=0) on embeddings, head, norms, mixing scalars
Schedule	WSD: 6% linear warmup → stable peak → (1−√) cooldown from cooldown_frac=0.65
Positions	RoPE (rotary), applied to q,k inside attention
MLP	Squared-ReLU (ReLU²) activation, 4× expansion
Attention	PyTorch SDPA, causal mask, dropout 0.05; value-residual (learnable per-layer mix to first-layer values)
Other	GPT-2 residual init: 1/√(2·n_layer), weight tying, gradient clip 1.0
Final val	1.1754 (33.0% improvement over 1.7533 baseline)

On the same dataset, seed 1337, 7-epoch budget, and fixed evaluate() function, this configuration achieves a final validation loss of 1.1754 or a 33.0% reduction from the 1.7533 baseline. Dissimilar to earlier configurations that reduced parameter count, the 16-layer submission is approximately 40M parameters which is approximately 47% larger than the baseline implementation. It can be seen that depth was intentionally swapped for performance, with wall-clock increasing from 4.6 to 9.1 minutes. As illustrated in Figure 2, 16 layers represent the accuracy/cost inflection point. This is as 20 layers produces only a marginal further reduction of 0.0009 at substantially greater runtime and was therefore not submitted.

Figure 4:

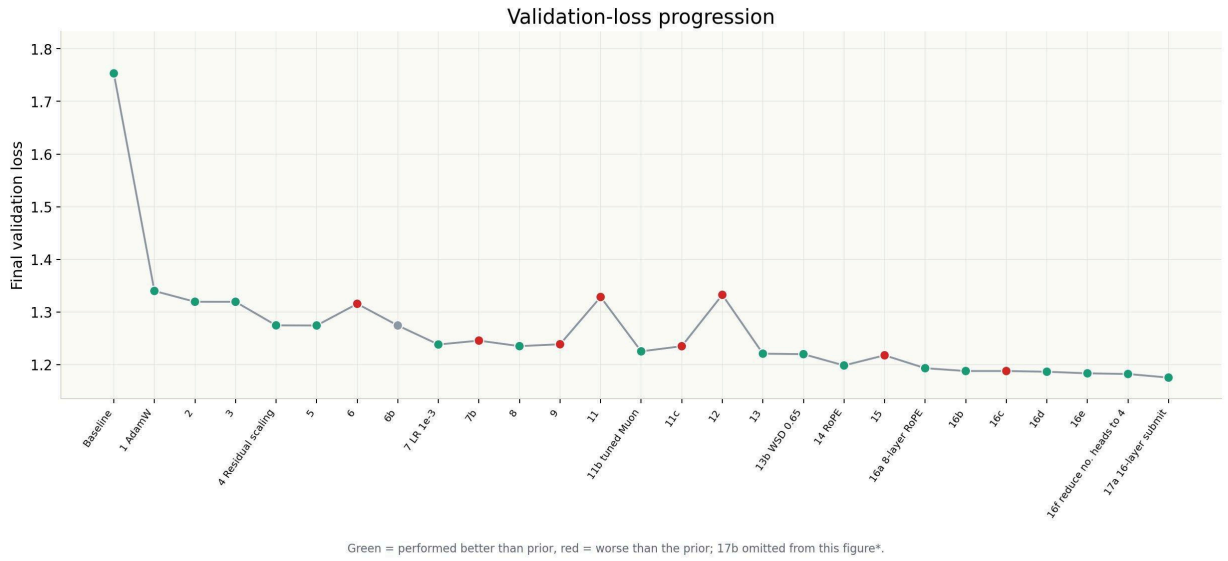

Figure 4. Best-so-far validation loss across the full experiment sequence. Green markers improved on the prior best, red regressed.

EXTENSIBILITY AND LIMITATIONS

The central observations scale beyond this task; on a 27M model, replacing SGD architecture with AdamW recovered 24% of baseline loss. As aforementioned, I've learnt that this systematic approach when applied across iterations (selecting the optimizer for the gradient geometry, locating learning rate optima empirically, distinguishing underfitting from overfitting before regularisation, and separating schedule problems from learning rate problems) accumulates as model and dataset size grow. When applying this notion to other models, any nuance regarding optimisation scales accordingly where for a model like Matilda the cost is days of compute.

The techniques applied like Muon, WSD, RoPE, and value-residual are current methods utilized in advancing efficient pretraining globally. In implementing these methods, I learnt that the difficulty presided in correctly adapting the different techniques to assessment conditions they were not designed for:

- Muon's default learning rate plateaued at this batch size
- QK-norm without a learnable scale degraded rather than improved performance

Identifying that frontier methods transfer in direction as opposed to magnitude became the distinction at the core of this paper.

Limitations

- **Time:** This submission was completed alongside full-time professional commitments; the response surface had not fully flattened at submission, and further iteration would likely yield additional gains, particularly in the untested directions below.
- **Single seed (1337):** Differences below approximately 0.005 fall within run-to-run noise and were not treated as significant; the seed cannot be averaged over under the assessment rules.
- **Compute not parameter efficiency:** The submission is ~40M parameters at ~9 minutes runtime which is larger and slower than the 8-layer configurations for a 0.012 gain (Figure 2). A parameter constrained variant would prioritise the 8-layer configuration.
- **Untested levers:** QK-norm with a learnable per-head scale, BPE vocabulary sweeps under the per-character metric, and mixed-precision (bf16) training were deprioritised once reductions fell to the fourth and fifth decimal.

CONCLUSION

The hypothesis that the model was underfitting served as the basis for every subsequent decision regarding optimization in this framework. The final result of 1.1754, a 33.0% improvement over baseline was the product of various decisions; adapting a frontier optimizer to the constraints of the assessment, diagnosing and correcting a self-introduced regression, counterintuitive learning rate call validated by the data, and deliberate compromisation of accuracy versus compute.

Approaching this with limited experience proved as much an advantage as a constraint; it forced me to reason from first principles and test every assumption against the validation curve. It was a genuinely humbling exercise, and reinforced that this is the problem space I want to be involved in, especially the pioneering work Maincode is doing with Matilda.