Most Transformer Modifications Still Do Not Transfer at 1–3B: A 2020–2026 Update to Narang et al. (2021) with Downstream Evaluation and a Noise Floor

Anonymous ACL submission

Abstract

Narang et al. (2021) evaluated 40+ Transformer modifications at T5-base scale and concluded that most did not transfer. Five years later, the typical working regime has moved to 1–3B parameters, downstream evaluation has replaced pretraining perplexity, and a substantially different catalogue of modifications has emerged. We revisit their question by testing 19 post-2021 Transformer modifications against a controlled baseline (20 configurations total) at 1.2B and 3B under strict iso-data, iso-compute, iso-recipe control, with a multi-seed baseline noise floor and CLIMB-12 downstream evaluation as the primary metric. The central finding reproduces theirs at this curated set: most modifications do not transfer. Of the 19 modifications, only two clear Bonferroni correction at 1.2B; one of those two further fails to train stably at 3B under the shared recipe. We also find that the loss–downstream gap reported by Tay et al. (2023) enlarges several-fold for attention-output modifications: two significant failures converge to within 2–3% of baseline validation loss yet drop 6–16 CLIMB-points. We conclude that noise-floor reporting, downstream evaluation, and cross-scale stability testing are now prerequisites for architecture comparisons at 1–3B.

1 Introduction

Narang et al. (2021) tested 40-plus Transformer modifications on T5-base (∼220M parameters) under a shared codebase and training recipe, and reported that most did not transfer: most reported gains either disappeared or fell within seed variance under controlled comparison. The study became the standard reference for whether a proposed architecture change is likely to survive re-implementation.

Five years later, three conditions of that comparison have changed. *First, the scale has moved.* 1–3B parameters is now the typical working regime for open small-LM releases and controlled architecture comparisons, ten times the T5-base size. *Second, the metric has moved.* Pretraining perplexity, Narang et al.'s primary metric, has since been shown to be an unreliable predictor of downstream task accuracy across architectures and scales (Tay et al., 2023; Liu et al., 2023; Lourie et al., 2025). *Third, the catalogue has moved.* A new generation of modifications—softmax replacements (Ramapuram et al., 2024; Nakanishi, 2025; Zuhri et al., 2025), attention regularizers (Henry et al., 2020; Leviathan et al., 2024; Ye et al., 2024; Zhou et al., 2025), residual variants (Pagliardini et al., 2024; Zhu et al., 2024; Touvron et al., 2021; Kimi Team et al., 2026), and FFN activations (Shazeer, 2020)—has been proposed since, none of which appear in Narang et al.'s original 40+ set.

We revisit Narang et al.'s question under these three changes. We build a controlled benchmark of 19 intra-Transformer modifications proposed since their study, paired with a shared baseline (20 configurations total), and train each under a fixed 1.2B (and separately 3B) Llama-style baseline configuration, sharing the same mixed-domain English corpus, optimizer recipe, tokenizer, and data shuffle (§3). Two further protocol updates: we establish a baseline noise floor by running three independent seeds each of the baseline and the strongest modification, and adopt the seed-level distribution as the significance threshold (§3.3); and we evaluate on CLIMB-12 downstream benchmarks rather than on pretraining loss.[1]

Main finding. Narang et al.'s central conclusion reproduces at 1.2B. Of the 19 modifications, seven fall within the baseline bootstrap 95% interval ($|z| < 2$); six are nominal degraders ($|z| > 2$); six are nominal improvers. Under Bonferroni at

[1] We adopt the 12-task downstream evaluation protocol of Diao et al. (2025), which we refer to as CLIMB-12. The corpus we pretrain on, CLIMB-Mix-400B, is released alongside that protocol (Appendix A); specific tasks are listed in §3.4.

m=19 (one test per non-baseline method), two improvers survive; five degraders remain significant. A 3B robustness check (13 attempted; ten complete, three diverge) adds a second filter. The ten completing runs (1 baseline, 7 improver candidates with positive CLIMB-avg delta, 2 failure references) all preserve the sign of their 1.2B effect; ordering within the improver band reshuffles (Spearman $\rho = -0.27$). One of the two 1.2B Bonferroni survivors fails to train stably at 3B across all three noise-floor seeds (42, 43, 44): each diverges to NaN in a narrow 2800–3500 step window with a reproducible Post-LN-on-FFN variance growth signature (Xiong et al., 2020) rather than seed noise. A 1.2B-significant gain therefore does not automatically imply 1–3B deployability.

Downstream matters. The loss-downstream decoupling identified by Tay et al. (2023) at architecture-family scale enlarges several times over in the attention-modification setting at 1.2B. Sigmoid Attention's validation loss is only 2.4% above baseline and SSMax's only 3.0%, yet their CLIMB accuracies drop by 16 and 6 percentage points. A loss-only ranking would place both near baseline. The residual-side failure AttnRes, whose loss rises by 13%, has a CLIMB-avg drop consistent with that loss gap (Appendix E); the decoupling is concentrated in attention-output modifications, not a general "loss is unreliable" claim.

Contributions. The contribution is the protocol update, not a new architecture:

(i) A controlled benchmark of 19 post-2021 modifications against a shared baseline (20 configurations total) under one shared recipe, with a 10-configuration 3B robustness check.

(ii) A noise-floor protocol calibrated on three three-seed references (baseline, QK-Norm, Softpick), with σ_{base} used as the conservative common threshold under Bonferroni / Holm / BH correction.

(iii) A cross-hardware baseline reproduction (NVIDIA A100 vs. primary accelerator) matching CLIMB-avg to 1×10^{-4}.

(iv) Three recommendations (R5–R7, §6) extending Narang et al.'s original four: multi-seed noise floors, downstream evaluation for attention-output modifications, and cross-scale stability checks.

2 Related Work

2.1 Controlled comparative studies

The direct ancestor of the present study is Narang et al. (2021), who evaluated 40-plus modifications of the T5 encoder-decoder architecture under a controlled recipe at T5-Base scale (∼220M parameters) and concluded that most proposed modifications did not transfer. That paper established the standard methodological frame for post-hoc audits of Transformer architecture claims—iso-codebase, iso-recipe, single-seed pretraining-perplexity comparison—and is the reference against which individual architecture improvements are judged. We adopt the same spirit and update the protocol along three axes that have since become binding: scale (1–3B rather than T5-Base), metric (downstream rather than perplexity), and statistical threshold (multi-seed noise floor with family-wise correction rather than single-seed point estimates). The modification set is disjoint from theirs by construction. Tay et al. (2023) compare ten architecture *families* from 15M to 40B parameters and document a systematic gap between upstream pretraining perplexity and downstream task performance; their observation motivates our metric change and is the closest prior statement of the upstream–downstream decoupling Appendix E analyzes. Their comparison is at the level of architecture families; ours is at the level of within-Transformer modifications, where the decoupling turns out to be several times larger in magnitude for post-2024 attention-output changes. Liu et al. (2023) and Lourie et al. (2025) provide additional evidence that pretraining loss is an unreliable downstream predictor in adjacent settings.

2.2 Modifications evaluated in this study

Soft modifications of softmax attention. QK-Norm (Henry et al., 2020; Dehghani et al., 2023) applies an RMSNorm to query and key projections before the dot product. Selective Attention (Leviathan et al., 2024) adds a learnable per-position mask that down-weights distractor keys. Differential Transformer (Ye et al., 2024) computes two softmax distributions per head and returns their difference. Value Residual (Zhou et al., 2025) threads a shortcut from value projections to subsequent layers. Softpick (Zuhri et al., 2025) rectifies softmax, $\text{softpick}(z) = \text{ReLU}(\text{softmax}(z) - 1/n)$, producing exact zeros for below-uniform weights.

Hard replacements of softmax. Ramapuram et al. (2024) replace softmax with element-wise sigmoid, arguing the absence of cross-position normalization can ease long-context scaling. SSMax (Nakanishi, 2025) is a scalable-softmax variant with a learned per-head temperature. Our results (§4, Appendix E) indicate that at 1.2B, removing softmax's normalization across positions is not a productive modification on CLIMB-12.

Residual and inter-layer connectivity. DenseFormer (Pagliardini et al., 2024) replaces the standard residual with a depth-weighted average of all previous layers' outputs. HyperConnections (Zhu et al., 2024) introduce multi-channel inter-layer connectivity and have been used in DeepSeek-V3 (DeepSeek-AI, 2024) ($\sim$671B parameters). LayerScale (Touvron et al., 2021) adds a learnable per-channel gate to the residual branch. AttnRes (Kimi Team et al., 2026) replaces the identity residual after attention with softmax attention over preceding layer outputs; we adopt the pseudo-query variant (Appendix B.1).

Normalization and FFN. Sandwich Norm (Ding et al., 2021) wraps both Pre- and Post-LN around each sublayer. HybridNorm (Zhuo et al., 2025) mixes attention-side Pre-LN with FFN-side Post-LN. Gated Attention (Qiu et al., 2025) adds a per-head sigmoid gate on the attention output. Shazeer (2020) proposes GLU variants including SwiGLU (the baseline used here) and GeGLU. So et al. (2022) introduce Primer's non-gated ReLU^2 activation, since adopted by PaLM-540B.

2.3 FLOPs accounting

The $6N$ approximation from Chinchilla scaling laws (Hoffmann et al., 2022) does not distinguish attention's $O(s^2d)$ cost from the FFN's $O(s{\cdot}d{\cdot}d_{\text{inter}})$ cost. We compute operation-level per-step FLOPs (Appendix C) so that method-to-method compute differences can be read directly from Table 2.

3 Experimental Methodology

3.1 Methods and taxonomy

We select 19 Transformer modifications proposed since Narang et al. (2021)'s 2021 study, each claiming a gain over a standard Llama-style baseline (or, for one combination variant QK-Norm + GeGLU, representing a production stack of standard components), and pair them with the baseline as the 20th configuration. The primary grouping (Table 1) is by the *module* the modification targets: attention, FFN, normalization, residual.

The 19-modification set is not a census of the post-2021 catalogue, nor a random sample. It is curated to cover the categories that Narang et al. (2021)'s original 40+ set spanned (attention, FFN, normalization, residual, softmax). Sandwich Normalization (Ding et al., 2021), Hybrid Normalization (Zhuo et al., 2025), ReLU^2 (So et al., 2022), and Gated Attention (Qiu et al., 2025) were added to correct under-representation relative to Narang et al.'s set, where activation and normalization changes accounted for $\sim$30% of the tested modifications. Inclusion required (a) targeting a vanilla decoder Transformer rather than a non-Transformer alternative (state-space, linear attention), (b) public implementation or sufficient detail to reproduce, and (c) at least one published evaluation at $\geq$0.5B parameters. Claims of the form "most modifications do not transfer" are scoped to this curated set.

Within the attention column, a secondary label (*soft*/*hard*) records whether softmax remains the primary mixing operation: *soft* if the attention block uses softmax of a (possibly transformed) $QK^\top$ as its mixing weights, with at most element-wise scalar reweighting of the result (QK-Norm, Selective Attention, SSMax, Value-Residual, Softmax-Cap); *hard* if it replaces softmax with a different nonlinearity (Sigmoid Attention), rectifies the output to break row-stochasticity (Softpick), or combines multiple softmaxes in a way that admits negative weights (Differential Transformer).

3.2 Training setup

The 1.2B baseline is a decoder-only Llama-2-style Transformer (Touvron et al., 2023): 24 layers, hidden 2048, 32 attention heads (8 KV heads, GQA), SwiGLU FFN intermediate 5632, RoPE, 65,664-token BPE vocabulary, context 1024. The 3B variant (§5) scales to 28 layers and hidden 3072 with all other details identical. Method-specific hyperparameters (e.g. softmax cap threshold) follow original papers; all other architectural hyperparameters are held fixed. We pre-tokenize a mixed-domain English corpus into 1024-token sequences; the same pack and shuffle seed are used across every 1.2B run; 3B runs consume a superset of the 1.2B shards (Appendix A reports the exact 23.28B / 60.04B subset/superset splits). All 1.2B runs train for 44,000 steps at 2^{19} tokens/step (23.07B train-

Category	Subcategory	Methods
Baseline	—	Baseline
Attention	QK-norm	QK-Norm
	Learnable gate	Selective Attention, Selective + QK-Norm
	Softmax cap	Softmax-Cap
	Scaled softmax	SSMax
	Value shortcut	Value-Residual
	Sigmoid replacement	Sigmoid Attention
	Rectified softmax	Softpick
	Head subtraction	Diff-Attn
	Output gate	Gated + QK-Norm
FFN	Gated (GeGLU)	GeGLU
	Non-gated ReLU^2	ReLU^2
	GeGLU + QK-Norm stack	QK-Norm + GeGLU
Normalization	Sandwich	Sandwich Norm
	Attn Pre + FFN Post	HybridNorm
Residual	Depth average	DenseFormer
	Learnable gate	LayerScale
	Multi-channel	HyperConnections
	Cross-layer pseudo-Q	AttnRes

Table 1: The 20 configurations studied (1 baseline + 19 modifications: 10 attention + 3 FFN + 2 normalization + 4 residual), grouped by the module modified. The set covers the Attention, FFN, Normalization, and Residual categories present in Narang et al. (2021)'s original 40+ comparison, expanded with methods proposed between 2020 and 2026. Within the Attention column a soft/hard label indicates whether softmax remains the primary mixing operation: QK-Norm, Selective Attention, SSMax, Value-Residual and Softmax-Cap apply softmax to a (possibly transformed) $QK^\top$ with at most an element-wise gate on the result and are *soft*; Sigmoid replaces softmax with a different nonlinearity, Softpick rectifies softmax in a way that breaks the row-sum, and Diff-Attn combines two softmaxes via subtraction that can produce negative weights—these three are *hard*.

ing tokens). Optimization: AdamW, lr 3×10^{-4}, β_1=0.9, β_2=0.95, weight decay 0.1, gradient clipping 1.0, 2,000 linear warm-up, cosine decay to 10%, bf16 with FP32 master weights. No per-method tuning. Each 1.2B run uses one node of eight AI accelerators (∼16h); each 3B run uses two RDMA-connected nodes (∼3 days). Hardware vendor anonymized for review.

Numerical-stability disclosure. At 3B, three methods (HybridNorm, SSMax, HyperConnections) diverge under the shared recipe; the baseline and 9 other 3B-attempted modifications train to completion. Each divergence has documented theoretical fragility (Post-LN deep-layer variance growth (Xiong et al., 2020), softmax-replacement logit blowup (Ramapuram et al., 2024; Nakanishi, 2025), multi-channel EMA residual drift (Zhu et al., 2024)) independent of accelerator; observed grad-norm signatures are mechanism-aligned (§5.3). The possibility that a tuned recipe could stabilize any of the three is discussed in Limitations.

3.3 Noise-floor protocol

We establish a noise floor by running three independent seeds (42, 43, 44) each of three reference configurations: the baseline, the strongest *soft* modification (QK-Norm), and the strongest *hard* modification (Softpick). Each seed varies both initialization and data-shuffle RNG. Three-seed empirical standard deviations of CLIMB-avg (unbiased): σ_{baseline}=0.00208, $\sigma_{\text{QK-Norm}}$=0.00146, σ_{Softpick}=0.00133. The baseline calibration is the most conservative; we use σ_{baseline} operationally so improver z-scores are if anything understated. The primary significance test is a seed-only bootstrap with N=10,000 resamples drawing three baseline seeds with replacement: the QK-Norm and Softpick 95% intervals are disjoint from baseline's ($\Pr(\text{method} \leq \text{base})$=0 in the paired bootstrap). For a single-seed method with CLIMB-avg x, we report $z = (x - \hat{\mu}_{\text{base}})/\hat{\sigma}_{\text{base}}$ and flag $|z| > 2$ as significant. Per-method cross-checks not inflated by catalogue size: 3-seed Welch's t gives t=+4.39, p=0.015 for QK-Norm and t=+6.36, p=0.0053

for Softpick. A per-task Welch + Stouffer combination is reported in Appendix E.4.

3.4 Downstream evaluation suite

We evaluate every model on CLIMB-12: PIQA (Bisk et al., 2020), ARC-Challenge and ARC-Easy (Clark et al., 2018), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2020), SocialIQA (Sap et al., 2019), MMLU (Hendrycks et al., 2021), OpenBookQA (Mihaylov et al., 2018), BoolQ (Clark et al., 2019), RACE (Lai et al., 2017), LAMBADA (Paperno et al., 2016), and TruthfulQA-MC2 (Lin et al., 2022), using the default few-shot configuration in lm-evaluation-harness (Gao et al., 2023); CLIMB-avg is the unweighted macro-average. Downstream rather than pretraining loss is the primary metric because nearly-identical validation losses can correspond to 6–16-point CLIMB differences (Appendix E). Optimizer, schedule, tokenizer, sequence length, batch size, RoPE base, initialization, and data shuffle are fixed across all 20 configurations. Per-step FLOPs are computed by operation-level counting (Appendix C); all 19 modifications lie within $\pm 0.13\%$ of baseline.

4 Results at 1.2B

4.1 Overview

Table 2 reports each method's CLIMB-avg, delta relative to baseline, bootstrap z-score, and per-step FLOPs. Most modifications in this curated set do not transfer, consistent with Narang et al. (2021) at T5-base. Of the 19 modifications, seven land within the baseline bootstrap 95% interval ($|z| < 2$); six significantly degrade baseline; six nominally improve at $|z| > 2$, $p < 0.05$—Softpick (Zuhri et al., 2025) ($z{=}{+}4.47$), HybridNorm (Zhuo et al., 2025) ($z{=}{+}3.19$), QK-Norm (Henry et al., 2020) ($z{=}{+}2.51$), Sandwich Norm (Ding et al., 2021) ($z{=}{+}2.45$), ReLU^2 (So et al., 2022) ($z{=}{+}2.36$), and Selective Attention (Leviathan et al., 2024) ($z{=}{+}2.15$). Under Bonferroni at $\alpha{=}0.05/19$ (one test per non-baseline method), Softpick ($p_{\text{Bonf}} \approx 1.5{\times}10^{-4}$) and HybridNorm ($\approx 0.027$) survive; the other four reduce to suggestive evidence. The five large-effect degradations remain significant after correction; Diff-Attn does not. Per-method 3-seed Welch's t gives QK-Norm $t{=}{+}4.39$, $p{=}0.015$ and Softpick $t{=}{+}6.36$, $p{=}0.0053$. Holm–Bonferroni preserves the same 7 methods; Benjamini–Hochberg at FDR$= 0.05$ adds QK-Norm, Sandwich Norm, ReLU^2, and Diff-Attn. The qualitative conclusion is the same under all three procedures. Of the two 1.2B Bonferroni survivors, only one trains to completion at 3B under the shared recipe; the other diverges at step 3500 with a Post-LN-compatible signature (§5.3).

4.2 Attention modifications span the largest effect range

The ten attention modifications (Table 1) span CLIMB-avg from $+0.009$ (Softpick) to -0.161 (Sigmoid Attention) —roughly eight times the baseline bootstrap width. Six of the eight non-combination attention modifications are statistically distinguishable from baseline, three improving and five degrading; only Value-Residual and Softmax-Cap fall inside the noise band. Modifying the attention normalization is therefore a high-variance intervention: small perturbations such as QK-Norm shift the result by a few σ_{base} in the positive direction, while softmax-replacement variants (Sigmoid, SSMax) overshoot the negative direction by two orders of magnitude over the noise floor. The soft/hard label of Table 1 does not predict outcome at the level of individual methods: the single Bonferroni-surviving improvement (Softpick) is hard, the largest improvers besides Softpick are soft, and the largest failure (Sigmoid) is hard. The label is a structural taxonomy, not a predictor.

4.3 Residual and FFN modifications

None of the four residual-connection modifications improves over baseline; three are significantly worse ($z{=}{-}4.4$, -9.8, -29.5); DenseFormer is statistically indistinguishable ($z{=}{-}0.97$). On FFN: GeGLU ($z{=}{+}0.20$) is indistinguishable from the SwiGLU baseline; ReLU^2 (So et al., 2022), evaluated at iso-parameter count, reaches $z{=}{+}2.36$ but not Bonferroni; QK-Norm + GeGLU (the production stack of Llama-3, Gemma-2) reaches the top of the 3B rank at iso-token (§5.2) despite an uninformative 1.2B $z{=}{+}1.60$.

4.4 Per-method evidence for Softpick

Of the two 1.2B Bonferroni survivors, Softpick (Zuhri et al., 2025) also trains stably at 3B (§5). Softpick is a one-line modification of softmax, $\text{softpick}(z) = \text{ReLU}(\text{softmax}(z) - 1/n)$, producing exact zeros for below-uniform attention weights. The original paper reports it eliminates attention sinks and reduces massive activations at

Method	CLIMB-avg	Δ	z	p_{Bonf}
Softpick	0.4922	$+0.009$	$+4.47$	1.5×10^{-4}
HybridNorm‡	0.4896	$+0.007$	$+3.19$	0.027
QK-Norm	0.4882	$+0.005$	$+2.51$	0.23
Sandwich Norm	0.4880	$+0.005$	$+2.45$	0.27
ReLU^2	0.4878	$+0.005$	$+2.36$	0.35
Selective Attention	0.4874	$+0.004$	$+2.15$	0.61
QK-Norm + GeGLU	0.4863	$+0.003$	$+1.60$	1.00
Gated + QK-Norm	0.4855	$+0.003$	$+1.23$	1.00
Softmax-Cap	0.4854	$+0.002$	$+1.18$	1.00
Selective + QK-Norm	0.4852	$+0.002$	$+1.11$	1.00
Value-Residual	0.4846	$+0.002$	$+0.81$	1.00
GeGLU	0.4834	$+0.001$	$+0.20$	1.00
Baseline	0.4829	0.000	0.00	—
DenseFormer	0.4809	-0.002	-0.97	1.00
Diff-Attn	0.4779	-0.005	-2.45	0.27
LayerScale	0.4738	-0.009	-4.42	1.8×10^{-4}
HyperConnections	0.4626	-0.020	-9.79	2.6×10^{-21}
AttnRes	0.4218	-0.061	-29.47	$<10^{-7}$
SSMax	0.4208	-0.062	-29.94	$<10^{-7}$
Sigmoid Attention	0.3217	-0.161	-77.68	$<10^{-7}$

Table 2: Main results at 1.2B, sorted by CLIMB-avg. The Baseline row reports the three-seed mean $\bar{x}_{\text{base}}$=0.4829 used as the reference for Δ and z throughout: $\Delta = x - \bar{x}_{\text{base}}$, $z = \Delta/\sigma_{\text{base}}$, with σ_{base}=0.00208 (unbiased 3-seed empirical std; §3.3). p_{Bonf}: Bonferroni at m=19 (one test per non-baseline method; the Baseline row is the z=0 reference, not a tested hypothesis). $|z|$>20 should be read as "substantially below baseline noise." ‡ HybridNorm passes Bonferroni at 1.2B but diverges at 3B.

scales up to 340M, evaluated primarily on pretraining loss. The 1.2B and 3B controlled replication here uses the same protocol as the other 19 methods, with full CLIMB-12 downstream evaluation and per-method multi-seed evidence. Three Softpick seeds yield CLIMB-avgs 0.4922, 0.4905, 0.4931 (mean 0.4919, σ=0.00133); three baseline seeds yield 0.4820, 0.4815, 0.4853 (mean 0.4829, σ=0.00208). The 3-seed Welch's t-test (t=+6.36, p=0.0053) is a per-method test that does not inflate with catalogue size. Under the operational criterion of §3.1, Softpick is *hard*: ReLU rectification zeroes weights below $1/n$, so the row no longer sums to one. The narrative "replacing softmax hurts" has a counter-example introduced by Zuhri et al. and replicated here. What separates this from the two significant hard failures (Sigmoid, Diff-Attn) is that rectified softmax preserves *peaked* attention: weights surviving rectification are exactly the softmax weights above uniform, so a single-key lookup pattern (e.g. LAMBADA) is structurally supported (Appendix E).

4.5 Most modifications are indistinguishable from baseline

Seven of the 19 modifications land inside the baseline bootstrap 95% interval ($|z| < 2$): QK-Norm + GeGLU, Gated + QK-Norm, Softmax-Cap, Selective + QK-Norm, Value-Residual, GeGLU, and DenseFormer (the baseline reference itself sits at z=0 by construction). Narang et al. report approximately five of 40+ modifications yielded a meaningful T5-base improvement; our 19-modification 1.2B sample (6 nominal improvers, 2 Bonferroni survivors, 1 surviving the 3B filter) is consistent in order of magnitude. We do not claim our 19-modification set is random or exhaustive; the headline "most do not transfer" is scoped to this curated set (§3.1).

5 Robustness at 3B

5.1 Setup

Deployability claims need the harder question of whether 1.2B gains *transfer* to the next scale. We re-train a subset of the 1.2B methods at 3B under the same iso-recipe (28 layers, hidden 3072, all other details from §3.2). Each 3B run trains to an iso-token milestone of ∼23B tokens ($44{,}000\times2^{19}$). No per-method retuning; a method that diverges at 3B is reported as a scale-up failure, not discarded. A full 20-configuration 3B sweep is approximately 2.5× the 1.2B budget; we instead train 13 configurations: the 3B baseline; seven improver candidates (the five 1.2B nominal improvers other than HybridNorm, plus two sub-Bonferroni catalogue methods QK-Norm + GeGLU and Selective + QK-Norm);

Method	1.2B	3B	Δ 3B	rank
QK-Norm + GeGLU	0.4863	**0.5158**	+0.0169	7→1
QK-Norm	0.4882	0.5129	+0.0139	3→2
Selective + QK-Norm	0.4852	0.5108	+0.0118	10→3
Selective Attention	0.4874	0.5097	+0.0107	6→4
Sandwich Norm	0.4880	0.5088	+0.0099	4→5
Softpick	**0.4922**	0.5084	+0.0095	1→6
ReLU^2	0.4878	0.5083	+0.0093	5→7
Baseline	0.4829	0.4989	0.0000	13→8
AttnRes	0.4218	0.4322	−0.0667	19→9
Sigmoid Attention	0.3217	0.3187	−0.1803	20→10

Table 3: 3B CLIMB-avg at the iso-token milestone (∼23B training tokens) alongside the 1.2B value from Table 2. The 1.2B Baseline value is the three-seed mean ($\bar{x}_{\text{base}}$=0.4829); the 3B Baseline is single-seed. Δ 3B is each method's CLIMB-avg minus the 3B single-seed baseline (0.4989); we have not run multi-seed 3B calibration. "rank" is each method's position (20 at 1.2B; 10 completed at 3B).

two 1.2B significant failures (Sigmoid Attention, AttnRes); and three methods that diverge at 3B (HybridNorm, SSMax, HyperConnections; §5.3). HybridNorm passes 1.2B Bonferroni but is grouped with the 3B divergences. Ten configurations complete; three diverge.

5.2 Improver candidates transfer at 3B; rank reshuffles

The seven 1.2B improver candidates that train to completion at 3B all maintain a positive CLIMB-avg delta against a 3B baseline trained under the identical recipe (Table 3). Deltas range from +0.009 (ReLU^2) to +0.017 (QK-Norm + GeGLU). The two 1.2B-significant failures retain large negative deltas: Sigmoid Attention drops by −0.180 at 3B (vs. −0.161 at 1.2B); AttnRes drops by −0.067 (vs. −0.061). *The sign of every 1.2B signal is preserved at 3B*: 7/7 improvers stay positive, 2/2 failures stay negative. Relative ordering reshuffles substantially: QK-Norm + GeGLU moves from 1.2B rank 7 (z=+1.60, not Bonferroni-significant) to 3B rank 1; Selective + QK-Norm from rank 10 to rank 3; Softpick, the 1.2B Bonferroni survivor that completes 3B, slips to rank 6. Spearman ρ=−0.27 on the seven improvers indicates 1.2B improver *rank* is a weak predictor of 3B improver rank; with n=7 and single-seed measurements per method, this ρ is far from significant (p≈0.56 two-sided) and is reported as a directional signal, not a precise estimate. The transferable signal is in the *sign* of the 1.2B effect, not fine ordering: 1.2B can reliably filter *whether* a modification helps but is too noisy to rank *which* of a set of in-noise improvers will help most at the next scale.

5.3 Three methods diverge at 3B; divergences are mechanism-aligned

Three methods that passed the 1.2B noise floor failed to train at 3B (Table 6 in Appendix E records when each diverged; Figure 1 in Appendix E shows the HybridNorm grad-norm trajectory).

HybridNorm. HybridNorm (Zhuo et al., 2025) combines Pre-LN attention with Post-LN FFN. At 1.2B it is Bonferroni-significant (z=+3.19, p_{Bonf}=0.027). To rule out seed-specific anomalies we retrained at 3B under all three noise-floor seeds (42, 43, 44); all three diverge in a narrow 2800–3500 step window shortly after the cosine peak (step ∼2000), with pre-NaN grad-norm in the 0.18–0.19 range identical to baseline's stable trajectory. This is a known Post-LN deep-decoder instability (Xiong et al., 2020): Post-LN amplifies activation variance with depth, and the 28-layer 3B stack under peak learning rate exceeds the threshold the 24-layer 1.2B stack retains. Three seeds failing in the same window is mechanism, not noise. Under a tuned warm-up schedule or lower peak learning rate HybridNorm may stabilise at 28-layer depth; our claim is restricted to the shared iso-recipe used across all 20 configurations, not to the method in isolation.

SSMax and HyperConnections. Scalable-softmax (Nakanishi, 2025) multiplies pre-softmax logits by $s \cdot \log n$ with learnable s; at 3B the grad-norm grows monotonically from 27 at step ∼4000 to 1578 at step 5600, then NaN at 5800—the softmax-replacement instability documented in Ramapuram et al. (2024). HyperConnections (Zhu et al., 2024) maintain an EMA state across layers; our output-side reimplementation (Appendix B.11)

is stable at 1.2B ($z_{1.2\mathrm{B}}=-9.79$, finite) but at 3B shows sustained grad-norm inflation from step $\sim$15000, NaN at 18300. Zhu et al.'s positive report is at 671B MoE with full input-side lane mixing, which the shared residual interface in our codebase does not expose.

Not hardware artefacts. Ten of thirteen 3B-attempted methods complete; the three that do not each exhibit a mechanism-aligned signature (Post-LN variance growth, softmax-replacement logit blowup, multi-lane EMA drift) independent of the accelerator. An A100 cross-hardware reproduction (Appendix D) confirms this independently.

5.4 3B as a second filter on top of 1.2B Bonferroni

Two methods pass Bonferroni at 1.2B; only one trains to completion at 3B with a positive delta. The 3B robustness experiment therefore removes one of the two 1.2B Bonferroni survivors. Reporting implication: any sub-billion-scale improvement proposed for 1–3B deployment should include either a 3B stability check under the submitter's shared recipe, or an explicit disclosure of the scale at which the gain was measured.

6 Discussion

Why don't most modifications transfer? Narang et al. (2021) enumerate candidate explanations for their negative finding. *(1) Idiosyncratic codebase.* Unlikely: the baseline is the community-standard decoder (RoPE, GQA, SwiGLU, RMSNorm) of Llama-2/3, Gemma-2, Qwen, Mistral. *(2) Insufficient per-method tuning.* Narang et al. tested this as a UT case study and found 23 of 25 sweeps still under-performed; we hold the recipe fixed because per-method tuning reintroduces the confound we are trying to remove. The QK-Norm and Softpick 3-seed results show that modifications that do transfer do so under the shared recipe. *(3) 1.2B is below scale.* The 3B experiment partially tests this: positive 1.2B deltas are preserved at 3B; methods whose authors report benefits at much larger scale (HyperConnections at 671B MoE, AttnRes at large-MoE) do not reproduce here. *(4) Most post-2021 modifications genuinely do not transfer at 1–3B.* The 1.2B-to-3B filter removes one of two Bonferroni survivors; six of 20 significantly *degrade* baseline, and the degradations are mechanism-aligned (softmax replacement without row-normalization, Post-LN variance growth, EMA drift), suggesting non-transfer is a systematic property of particular interventions rather than random noise.

Three further recommendations. Narang et al. (2021) close with four recommendations (try the modification in multiple codebases; apply it to diverse downstream tasks; keep hyperparameters fixed; report mean and standard deviation across trials), all of which still apply at 1–3B. We add three the post-2021 evidence makes binding.

R5. Multi-seed baseline noise floor. At 1.2B the three-seed σ on CLIMB-avg is ≈ 0.0021; any claimed gain below ~ 0.005 is within seed variance, and seven of 19 modifications fall here. Reporting single-seed numbers without a baseline distribution overstates the count of useful methods.

R6. Downstream evaluation for attention-output modifications. Tay et al. (2023)'s loss–downstream decoupling at architecture-family scale extends with several times the magnitude here. Two significant negatives (Sigmoid Attention, SSMax) reach within 2–3% of baseline validation loss while dropping 6–16 CLIMB-points (Table 2 together with Table 7 in Appendix E); a loss-only ranking would miss them. Residual-side failures (AttnRes, LayerScale) have loss increases consistent with their CLIMB drops; the decoupling is concentrated in attention output.

R7. Cross-scale stability check under the shared recipe. With 19 modifications tested against the same baseline, the family-wise false-positive rate at $\alpha=0.05$ is $\sim$62%; Bonferroni leaves two survivors, of which one fails to train stably at 3B. A Bonferroni-surviving 1.2B improvement is therefore not sufficient evidence of 1–3B deployability. A per-method 3-seed Welch's t-test additionally supplies single-method evidence not inflated by catalogue size.

Closing. The protocol is the contribution. Future 1–3B architecture studies need three prerequisites: multi-seed baselines, downstream evaluation, and cross-scale stability checks.

References

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: Reasoning about physical commonsense in natural language. *AAAI*.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *NAACL*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv:1803.05457*.

DeepSeek-AI. 2024. DeepSeek-V3 technical report. *arXiv:2412.19437*.

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, Rodolphe Jenatton, Lucas Beyer, Michael Tschannen, Anurag Arnab, Xiao Wang, Carlos Riquelme, Matthias Minderer, Joan Puigcerver, Utku Evci, and 2 others. 2023. Scaling Vision Transformers to 22 billion parameters. In *ICML*. ArXiv:2302.05442.

Shizhe Diao, Yu Yang, Yonggan Fu, Xin Dong, Dan Su, Markus Kliegl, Zijia Chen, Peter Belcak, Yoshi Suhara, Hongxu Yin, Mostofa Patwary, Celine Lin, Jan Kautz, and Pavlo Molchanov. 2025. Nemotron-CLIMB: CLustering-based Iterative Data Mixture Bootstrapping for language model pretraining. *arXiv:2504.13161*. CLIMB-Mix-400B mixed-domain English pretraining corpus.

Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. 2021. CogView: Mastering text-to-image generation via transformers. In *NeurIPS*. Introduces Sandwich Normalization for stabilizing large Transformers.

Bradley Efron. 1987. Better bootstrap confidence intervals. *Journal of the American Statistical Association*, 82(397):171–185.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2023. A framework for few-shot language model evaluation. https://github.com/EleutherAI/lm-evaluation-harness.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *ICLR*.

Alex Henry, Prudhvi Raj Dachapally, Shubham Pawar, and Yuxuan Chen. 2020. Query-key normalization for transformers. *EMNLP Findings*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, and 3 others. 2022. Training compute-optimal large language models. *arXiv:2203.15556*.

Andrej Karpathy. 2025. nanochat: The best ChatGPT that $100 can buy. https://github.com/karpathy/nanochat. Open-source small-LM reproduction stack; source of the 65,664-entry BPE tokenizer used in this work.

Kimi Team, Guangyu Chen, Yu Zhang, Jianlin Su, Weixin Xu, Siyuan Pan, Yaoyu Wang, Yucheng Wang, Guanduo Chen, Bohong Yin, Yutian Chen, Junjie Yan, Ming Wei, Y. Zhang, Fanqing Meng, Chao Hong, Xiaotong Xie, Shaowei Liu, Enzhe Lu, and 18 others. 2026. Attention residuals. *arXiv:2603.15031*. Submitted 16 Mar 2026.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *EMNLP*.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2024. Selective attention improves transformer. *arXiv:2410.02703*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *ACL*.

Hong Liu, Sang Michael Xie, Zhiyuan Li, and Tengyu Ma. 2023. Same pre-training loss, better downstream: Implicit bias matters for language models. In *ICML*.

Nicholas Lourie, Michael Y. Hu, and Kyunghyun Cho. 2025. Scaling laws are unreliable for downstream tasks: A reality check. In *Findings of EMNLP*. ArXiv:2507.00885.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *EMNLP*.

Ken M. Nakanishi. 2025. Scalable-Softmax is superior for attention. *arXiv:2501.19399*.

Sharan Narang, Hyung Won Chung, Yi Tay, William Fedus, Thibault Fevry, Michael Matena, Karishma Malkan, Noah Fiedel, Noam Shazeer, Zhenzhong Lan, Yanqi Zhou, Wei Li, Nan Ding, Jake Marcus, Adam Roberts, and Colin Raffel. 2021. Do Transformer modifications transfer across implementations and applications? In *EMNLP*.

Matteo Pagliardini, Amirkeivan Mohtashami, Francois Fleuret, and Martin Jaggi. 2024. DenseFormer: Enhancing information flow in transformers via depth weighted averaging. *arXiv:2402.02622*.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel

Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. *ACL*.

Zihan Qiu, Zekun Wang, Bo Zheng, Zeyu Huang, Kaiyue Wen, Songlin Yang, Rui Men, Le Yu, Fei Huang, Suozhi Huang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2025. Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. In *NeurIPS*. ArXiv:2505.06708; per-head sigmoid gate on attention output.

Jason Ramapuram, Federico Danieli, Eeshan Dhekane, Floris Weers, Dan Busbridge, Pierre Ablin, Tatiana Likhomanenko, Jagrit Digani, Zijin Gu, Amitis Shidani, and Russ Webb. 2024. Theory, analysis, and best practices for sigmoid self-attention. *arXiv:2409.04431*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. WinoGrande: An adversarial Winograd schema challenge at scale. *AAAI*.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019. Social IQa: Commonsense reasoning about social interactions. In *EMNLP*.

Noam Shazeer. 2020. GLU variants improve transformer. *arXiv:2002.05202*.

David R. So, Wojciech Mańke, Hanxiao Liu, Zihang Dai, Noam Shazeer, and Quoc V. Le. 2022. Primer: Searching for efficient transformers for language modeling. In *ICML*. Introduces non-gated ReLU^2 activation; adopted by PaLM-540B.

Yi Tay, Mostafa Dehghani, Samira Abnar, Hyung Won Chung, Sharan Narang, Dani Yogatama, Ashish Vaswani, and Donald Metzler. 2023. Scaling laws vs. model architectures: How does inductive bias influence scaling? In *Findings of EMNLP*. ArXiv:2207.10551.

Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. 2021. Going deeper with image transformers. In *ICCV*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 2 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv:2307.09288*.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. 2020. On layer normalization in the transformer architecture. In *ICML*. Demonstrates Post-LN activation variance growth with depth; motivates Pre-LN.

Tianzhu Ye, Li Dong, Yuqing Xia, Yutao Sun, Yi Zhu, Gao Huang, and Furu Wei. 2024. Differential transformer. *arXiv:2410.05258*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *ACL*.

Zhanchao Zhou, Tianyi Wu, Zhiyun Jiang, Fares Obeid, and Zhenzhong Lan. 2025. Value residual learning for alleviating attention concentration in transformers. In *ACL*. ArXiv:2410.17897.

Defa Zhu, Hongzhi Huang, Zihao Huang, Yutao Zeng, Yunyao Mao, Banggu Wu, Qiyang Min, and Xun Zhou. 2024. Hyper-connections. *arXiv:2409.19606*.

Zhijian Zhuo, Yutao Zeng, Ya Wang, Sijun Zhang, Jian Yang, Xiaoqing Li, Xun Zhou, and Jinwen Ma. 2025. HybridNorm: Towards stable and efficient transformer training via hybrid normalization. *NeurIPS*. ArXiv:2503.04598; Attention block Pre-Norm, FFN block Post-Norm.

Zayd M. K. Zuhri, Erland Hilman Fuadi, and Alham Fikri Aji. 2025. Softpick: No attention sink, no massive activations with rectified softmax. *arXiv:2504.20966*.

Limitations

Scale ceiling. A method we report as diverging at 3B may become stable under a retuned recipe; our claim is that it does not train under the *shared* recipe, the strong condition for reproducing the "single protocol, many methods" comparison of Narang et al. Some 1.2B failures may become beneficial at scales we do not reach (Zhu et al. 2024 at 671B MoE; Kimi Team et al. 2026 in a large-MoE setting). The scope of our claims is "at 1.2–3B on CLIMB-12 under a shared recipe."

Hardware generality. The main 1.2B and 3B comparisons run on a single brand of AI accelerator in bf16 (vendor anonymized for double-blind review; to be disclosed in the camera-ready version). To test for hardware-specific numerical sensitivity we re-trained the 1.2B baseline end-to-end on NVIDIA A100 under bit-identical model code, recipe, data shuffle, and 23B-token budget. The two baselines reach CLIMB-avg 0.4820 (primary) vs. 0.4821 (A100), a difference of 1×10^{-4}, with per-task deltas within ± 0.025 (Appendix D). This makes a blanket infrastructure explanation for the three 3B divergences unlikely: the baseline reproduces to 10^{-4} on A100 under the same recipe, while the divergences each have documented theoretical fragility independent of accelerator. We did not re-run the three 3B divergence cases on A100, so we cannot fully rule out an accelerator-specific contribution to those particular failures; the mechanism-aligned grad-norm signatures (§5.3)

are the strongest evidence we offer against that interpretation.

Single recipe, single corpus. Optimizer, schedule, warmup, initialization, and weight decay are held fixed across all 20 configurations, on a single mixed-domain English corpus (23B training tokens per run; Appendix A). A method whose original paper recommends a different recipe, or whose benefits are tied to multilingual or code data, is disadvantaged here. Per-method tuning would reintroduce the confound we removed; the choice narrows the reach of the claims.

Noise-floor generalization. σ is estimated from three calibration sets (baseline, QK-Norm, Softpick): 0.00208, 0.00146, 0.00133; we use σ_{baseline} as the most conservative. For $|z|>5$ this is robust by inspection; for borderline methods (Diff-Attn -2.45, Selective Attention $+2.15$, ReLU^2 $+2.36$, Sandwich Norm $+2.45$), the method's own σ could change the call. Bootstrap CIs at $n=3$ under-cover relative to nominal (Efron, 1987); the 95% interval is a nominal threshold rather than calibrated coverage.

Sub-Chinchilla token budget. All runs use a 23B-token iso-token milestone chosen to match per-method compute. This is close to Chinchilla-optimal for 1.2B models ($\sim$19 tokens/parameter), but under-trains the 3B models ($\sim$7.7 tokens/parameter). Within-scale comparisons remain valid because every method sees the same token stream for the same number of steps, but the absolute 3B CLIMB values are below a compute-optimal ceiling. This likely compresses 3B method gaps and makes single-seed 3B differences harder to separate from noise. We therefore restrict the 3B claims to sign preservation and stability outcomes, not fine ranking.

Single-seed coverage at the borderline. Per-method 3-seed evidence exists only for QK-Norm and Softpick (the two configurations selected as noise-floor anchors). The other four nominal improvers above $|z|>2$ (HybridNorm $z=+3.19$, Sandwich Norm $+2.45$, ReLU^2 $+2.36$, Selective Attention $+2.15$) are evaluated under a single seed each, with significance assessed against the baseline 3-seed bootstrap distribution. A 3-seed Welch's t on each would be a stricter test that does not inflate with catalogue size; the compute cost (8 additional 1.2B runs) was beyond this study's budget. The Bonferroni and Holm–Bonferroni corrections we apply are conservative substitutes that already rule all four as "not significant" after correction; the limitation is that the per-method-evidence framing of R5 is met fully only for our two anchor methods.

3B noise floor is single-seed except at the failure point. The 3B CLIMB-avg numbers in Table 3 use one seed per method except for HybridNorm, which we re-ran under all three noise-floor seeds (42, 43, 44) specifically to characterize its divergence (§5.3). The 3B baseline is therefore single-seed, so a 3B Δ within the 1.2B baseline noise band ($\sim$0.0021 CLIMB-avg) cannot be distinguished from baseline- specific seed variance at 3B. Our 3B claims are restricted to (i) the *sign* of each method's Δ relative to the 3B baseline (every 3B-completing improver shows $\Delta>0$ and both 3B-completing failures show $\Delta<0$ at sufficient magnitude that single-seed seed variance does not flip the sign), and (ii) *stability outcomes* (which methods diverge under the shared recipe). We do not draw fine rank conclusions among the 3B-completing improvers from single-seed data; the Spearman $\rho=-0.27$ result in §5.2 is reported as a coarse cross-scale rank-instability signal, not an estimate. A multi-seed 3B calibration would refine these claims and is left to future work.

Reproducibility

We aim to make the protocol of this paper reproducible end-to-end.

Anonymous reproducibility artifact. We submit an anonymized supplementary artifact containing configuration files for all 20 configurations (1 baseline plus 19 modifications), raw per-task CLIMB-12 outputs, seed-level results for the baseline / QK-Norm / Softpick anchors, 3B divergence summaries, and scripts that recompute the deltas, z-scores, Bonferroni / Holm / BH corrections, and Stouffer combinations reported in the paper. The artifact does not include checkpoints, the training corpus, or full training code; those will be released after review.

Code. All 20 configurations described in Appendix B are written under a single shared training graph; only the module under study differs between runs. Source code and configuration files for every 1.2B and 3B run will be released at the camera-ready stage. During review the implementations

are described in Appendix B at sufficient detail to re-implement from scratch (per-method equations, initialization values, and any deviation from the original paper).

Training recipe. Optimizer (AdamW, lr 3×10^{-4}, β_1=0.9, β_2=0.95, weight decay 0.1), gradient clipping (1.0), warmup (2,000 linear), schedule (cosine to 10% over the full horizon), and precision (bf16 with FP32 master weights) are held fixed across all 20 configurations. The same configuration runs at both 1.2B and 3B (only the model dimensions differ: {24 layers, hidden 2048} versus {28 layers, hidden 3072}).

Data and evaluation. The mixed-domain English pretraining corpus is pre-tokenized once, and the same pack-and-shuffle order is used across every 1.2B run; the 3B runs consume a superset of the same corpus under the same shuffle construction (Appendix A; the 1.2B subset is 23.28B tokens, the 3B superset 60.04B tokens, with each individual training run consuming 23B tokens at the iso-token milestone of 44,000 steps). All evaluation uses the lm-evaluation-harness (Gao et al., 2023) default few-shot configuration on the 12 CLIMB tasks listed in §3.4; CLIMB-avg is the unweighted macro-average.

Seeds and significance. Three independent seeds (42, 43, 44) are used for the baseline, QK-Norm, and Softpick noise-floor calibrations. Each seed varies both initialization and data-shuffle RNG. The bootstrap that produces the z-scores in Table 2 uses N=10,000 resamples with replacement from the 3-seed baseline distribution; the random-state seed for this bootstrap is fixed at 0 for reproducibility.

Hardware. The main 1.2B and 3B runs use eight-way data-parallel nodes of a single brand of AI accelerator (vendor anonymized for review, to be disclosed in the camera-ready). To verify the results do not depend on this hardware choice, we additionally re-trained the 1.2B baseline end-to-end on NVIDIA A100 GPUs (8×A100 80GB) under bit-identical model code, optimizer recipe, tokenizer, data shuffle, and 23B-token budget; the two runs match to 1×10^{-4} CLIMB-avg (Appendix D).

Software versions. Training: PyTorch 2.x, Transformers 4.x, FlashAttention 2.x; evaluation: lm-evaluation-harness v0.4.x. Specific patch versions and Docker images will be released alongside the code at the camera-ready stage.

A Data Composition

Corpus. All experiments train on publicly-available shards of the CLIMB-Mix-400B pretraining corpus (Diao et al., 2025), a mixed-domain English collection released alongside the CLIMB clustering-based data-mixture method. CLIMB-Mix-400B is curated by domain-conditional clustering of web, code, math, encyclopedia, and dialogue text. We adopt the 12-task downstream evaluation protocol from the same work (referred to as CLIMB-12 in this paper, §3.4). The 1.2B experiments use a fixed 23.28B-token subset; the 3B experiments use a 60.04B-token superset of the same shards. Both subsets are drawn from the same pre-shuffled shard sequence, so the 3B corpus contains the 1.2B corpus verbatim as a prefix.

Tokenizer. We tokenize with the 65,664-entry BPE tokenizer released with Karpathy (2025)'s open-source small-LM reproduction stack. The tokenizer is shared across baseline and all modifications; no tokenizer change is part of the experimental variable.

Packing. Raw documents are concatenated with document separators and split into 1024-token sequences (sequence length matches training context; cross-document attention is permitted within a packed sequence, matching standard practice for Llama-2-style pretraining). Packing produces 474 shards for the 1.2B subset (22,738,794 sequences, 23.28B tokens, 1.6×10^{-4}% tokens discarded by pack-boundary trimming) and 1268 shards for the 3B superset (58,637,625 sequences, 60.04B tokens, 1.1×10^{-4}% discarded).

Shuffle. All runs use an identical shuffle RNG seed at the sampler level, so the ith training example seen by method A and method B at step i is the same token sequence. Per-seed variants (baseline 43, 44; QK-Norm 43, 44; Softpick 43, 44) use the seed to initialize the weight and dropout RNGs; the shuffle seed is held fixed across all runs so that seed effects are isolated from curriculum effects.

Validation. A held-out 131,072-sequence validation pack from the same corpus is used to compute validation loss at every 100-step checkpoint. This is distinct from the downstream CLIMB-12 evaluation (§3.4), which uses task-specific public evaluation sets.

Reproducibility. The packing metadata (one JSON file each for 1.2B and 3B) is released with

the code. These include the shard count, total tokens, tokenizer reference, sequence length, and the tokens discarded by pack trimming, so a re-pack from the CLIMB-Mix-400B public shards reproduces our exact training stream.

B Implementation Notes

This appendix describes, per method, how our implementation maps to the published algorithm. This is especially important for methods whose 1.2B CLIMB-avg departs from the original paper's claim, and for methods that required light adaptation to fit the shared training graph.

B.1 AttnRes (pseudo-query variant)

The AttnRes implementation follows Kimi Team et al. (2026) Equation 2 and Table 4 row 2 ("AttnRes Full, pseudo-query"). At layer ℓ the residual output is

$$\alpha_{i\to\ell} = \mathrm{softmax}_i\Big(w_\ell^\top \cdot \mathrm{RMSNorm}(v_i)\Big),$$

$$h_\ell = \sum_i \alpha_{i\to\ell}\, v_i,$$

where $w_\ell \in \mathbb{R}^d$ is a learnable per-layer pseudo-query, v_i is the output of sublayer i (with v_0 the input embedding, following Kimi (2026) Eq. 3), and i ranges over $\{0, \ldots, \ell-1\}$, matching Kimi (2026) Eq. 4. Pseudo-query initialization: $w_\ell \sim \mathcal{N}(0, 0.02^2)$; RMSNorm $\epsilon = 10^{-5}$.

B.2 Softpick

Softpick (Zuhri et al., 2025) is implemented as $\mathrm{softpick}(z) = \mathrm{ReLU}(\mathrm{softmax}(z) - 1/n)$, where n is the number of keys. No post-ReLU renormalization is applied; this is the "rectified softmax" variant from the original paper. All other attention plumbing (RoPE, GQA, KV cache) matches the baseline full-attention implementation line-by-line.

B.3 QK-Norm

QK-Norm applies an RMSNorm to Q and K independently, along the head dimension, after the linear projection and before RoPE (Henry et al., 2020; Dehghani et al., 2023). Norm parameters are shared across heads within each projection and initialized to unity. No learnable temperature is added.

B.4 Selective Attention

Selective Attention follows Leviathan et al. (2024) with one adaptation: instead of accumulating the selective mask S across layers (which would require modifying the shared cross-layer interface), the mask is computed and applied *within* each layer. Heads are split into an attention group and a masking group; masking-head scores are passed through ReLU, zeroed on the diagonal and on the BOS column, and subtracted from the attention-head logits. The Q/K projections are shared across the two head groups, so the modification adds zero new parameters.

B.5 Sigmoid Attention

Sigmoid Attention replaces $\mathrm{softmax}(QK^\top/\sqrt{d_h})$ with $\mathrm{sigmoid}(QK^\top/\sqrt{d_h} + b)/n$, where b is a learnable scalar initialized to zero and n is the sequence length (Ramapuram et al., 2024). The $1/n$ normalization keeps attention magnitudes comparable to softmax at matched sequence lengths but does not enforce row sums to unity.

B.6 Scalable-Softmax (SSMax)

SSMax multiplies pre-softmax logits by $s \cdot \log n$, with $s = \mathrm{softplus}(s_{\text{logit}}) + 0.5$ a per-head learnable scale (Nakanishi, 2025). s_{logit} is initialized to 0, giving $s \approx 1.19$ at step 0. At the 1024-context training length used here, $\log n \approx 6.9$, so the effective temperature is $1/(s \cdot \log n) \approx 0.12$ at initialization, matching the $1/\sqrt{d_k} = 0.125$ baseline (per-head $d_k = 64$ at hidden 2048, 32 heads) within 4%.

B.7 Differential Transformer

Diff-Attn computes two softmax distributions per head by splitting Q and K (each into halves of width $d_h/2$) while keeping V at full width. The output is $(\mathrm{softmax}(Q_1K_1^\top/\sqrt{d_h/2}) - \lambda \cdot \mathrm{softmax}(Q_2K_2^\top/\sqrt{d_h/2}))V$, with λ reparameterized per the original paper's depth schedule $\lambda_{\text{init}} = 0.8 - 0.6\exp(-0.3(\ell-1))$ (Ye et al., 2024). Per-head GroupNorm is applied to the attention output.

B.8 Value-Residual

Value-Residual implements the ResFormer V-residual of Zhou et al. (2025): $U_\ell = \mathrm{softmax}(Q_\ell K_\ell^\top/\sqrt{d_h})((1-\lambda)V_\ell + \lambda V_1)$, where V_1 is read from the first layer's outputs and λ is a learnable scalar initialized to 0.5. Layer 0 degenerates to standard attention because no first-layer reference is yet available.

B.9 DenseFormer

DenseFormer replaces the standard residual with $Y_\ell = \sum_{j=0}^{\ell} \alpha_{\ell,j} X_j$, where $\{\alpha_{\ell,j}\}$ are learnable scalars initialized to the identity ($\alpha_{\ell,\ell} = 1$, others 0) (Pagliardini et al., 2024). This adds $N(N+1)/2 = 300$ parameters at 24 layers and does not alter per-step FLOPs.

B.10 LayerScale

LayerScale multiplies the sublayer output by a learnable per-channel gate γ initialized at 10^{-4}, giving $x_{\ell+1} = x_\ell + \gamma \odot f(x_\ell)$ (Touvron et al., 2021). $\gamma \in \mathbb{R}^d$ is a hidden-size-dimensional vector. Initialization value matches the CaiT paper.

B.11 HyperConnections

HyperConnections implements a lightweight output-side two-lane variant of Zhu et al. (2024). A fast lane follows the standard residual; a slow lane maintains an exponential moving average of sublayer outputs with learnable rate $\beta = \sigma(\beta_{\text{logit}})$, β_{logit} initialized to -2.2 ($\beta_0 \approx 0.1$). The slow lane contributes to the output with weight α initialized to 0, so the module reduces to standard residual at step 0. The full-rank input-side lane mixing A and output-side re-mixing R from the original paper are not reimplemented, because our shared residual interface does not currently expose the sublayer input. The full diagnostic in Appendix E discusses this together with other reasons the 1.2B dense setting here differs from the paper's 671B MoE setting.

B.12 GeGLU

GeGLU substitutes GELU for SiLU in the SwiGLU form: $y = W_{\text{down}}(\text{GELU}(W_{\text{gate}}x) \odot W_{\text{up}}x)$ (Shazeer, 2020). The intermediate size matches the SwiGLU baseline (5632 at 1.2B); parameter count and per-step FLOPs are identical.

B.13 Softmax-Cap

Softmax-Cap clips pre-softmax logits element-wise at ± 50, a recipe adopted in several long-context training reports to avoid bf16 overflow. At the 1024-context used here, the cap is almost never active; it is included as a minimal-surface-area baseline variant.

B.14 Sandwich Normalization

Sandwich Norm follows Ding et al. (2021): each sublayer output is bracketed by two RMSNorms, one before the sublayer and one before the residual add, giving $x_{\ell+1} = x_\ell + \text{RMSNorm}(f(\text{RMSNorm}(x_\ell)))$. This is distinct from HybridNorm in that both attention and FFN use the same sandwich wrapper; it adds zero new parameters beyond the extra norm scales.

B.15 HybridNorm

HybridNorm follows Zhuo et al. (2025): the attention block uses Pre-LN (RMSNorm before projection, residual add after), while the FFN block uses Post-LN (RMSNorm after residual add). At 1.2B this configuration clears Bonferroni significance ($z = +3.19$, $p_{\text{Bonf}} = 0.027$); at 3B it diverges at step 3500 with a 4.4× grad-norm spike, consistent with the Post-LN deep-layer variance growth documented by Xiong et al. (2020) (§5.3).

B.16 ReLU^2 activation

ReLU^2 implements the non-gated two-matrix FFN from So et al. (2022): $y = W_{\text{down}}\big(\text{ReLU}(W_{\text{up}}x)^2\big)$. We match baseline parameter count and FLOPs by expanding the intermediate dimension to $1.5 \times d_{\text{inter}}$, so that $2Hd_{\text{inter-primer}} = 3Hd_{\text{inter-SwiGLU}}$. At 1.2B it reaches $z = +2.36$ (nominally significant, does not clear Bonferroni).

B.17 Gated Attention with QK-Norm

Gated + QK-Norm follows Qiu et al. (2025): a per-head sigmoid gate multiplies the attention output before the output projection, $h = (\sigma(W_g x)) \odot \text{Attn}(QKV)$, with W_g applied head-wise. QK-Norm is stacked on the Q/K projections as described above. The combination adds approximately 0.1% to both parameter count and per-step FLOPs; at 1.2B it lands at $z = +1.23$, indistinguishable from baseline under Bonferroni.

B.18 Combination variants (Selective + QK-Norm; QK-Norm + GeGLU)

The two combination variants reuse the components specified above without modification. *Selective + QK-Norm* stacks Selective Attention's per-position learnable mask on top of the QK-Norm RMSNorm of Q/K projections; the mask is applied to the post-softmax attention weights, after QK-Norm has stabilized the pre-softmax logits. *QK-Norm + GeGLU* replaces both the attention's Q/K normalization and the FFN gating in a single run, matching the production stack used by Llama-3 and Gemma-2. Neither variant introduces components not already documented in §B.17 above; we list them separately because their 1.2B–3B ranking

trajectories (rank 10→3 and 7→1 respectively) are central to the rank-reshuffle finding of §5.2.

C FLOPs Accounting

C.1 Per-step FLOPs derivation

We decompose one training-step forward pass into attention-score, attention-projection, and FFN components:

$$F_{\text{attn-score}}(\ell) = 2 \cdot s^2 \cdot d$$
$$F_{\text{attn-proj}}(\ell) = 2 \cdot (2d^2 + 2d \cdot d_{\text{kv}})$$
$$F_{\text{FFN}}(\ell) = 2 \cdot k_{\text{ffn}} \cdot d \cdot d_{\text{inter}}$$

where s is sequence length, d is hidden size, d_{kv} accounts for grouped-query attention, d_{inter} is FFN intermediate size, and k_{ffn} is 3 for three-matrix variants (SwiGLU, GeGLU, QK-Norm + GeGLU) and 2 for the two-matrix ReLU^2 with $1.5\times$ intermediate width (yielding the same $k \cdot d_{\text{inter}}$ product).

The per-sequence forward FLOPs total to $\sum_\ell [F_{\text{attn-score}}(\ell) + s \cdot (F_{\text{attn-proj}}(\ell) + F_{\text{FFN}}(\ell))]$, times 3 for forward, backward, and activation-checkpointing accounting.

Our computations are per-method and use each config's actual d, d_{inter}, number of heads, and GQA ratio.

C.2 Per-method FLOPs and parameter counts

Table 4 reports per-step FLOPs and total parameters for each of the 20 configurations in the main comparison.

D Cross-Hardware Baseline Reproduction

To test whether our results carry hardware-specific numerical sensitivity, we re-trained the 1.2B baseline end-to-end on NVIDIA A100 GPUs under the bit-identical model code, optimizer recipe, tokenizer, data shuffle, and 23B-token budget used for the primary accelerator. The A100 run uses the same Llama-style 1.2B configuration described in §3.2 (24 layers, $d = 2048$, 32 heads, 8 KV groups, RoPE, GQA, SwiGLU, RMSNorm), the same AdamW recipe (β_1=0.9, β_2=0.95, weight decay 0.1), the same cosine learning-rate schedule with linear warm-up, and the same 44,000-step training horizon. The only intentional differences are the accelerator vendor (NVIDIA A100 vs. the primary accelerator) and the precision pathway (bf16 tensor cores on both, but with vendor-specific fused kernels).

Table 5 reports the per-task CLIMB accuracies on both platforms. The aggregate CLIMB-avg differs by 1×10^{-4} (0.4820 vs. 0.4821), within the third-decimal precision of the metric. Per-task deltas span ±0.029 with no systematic direction: the primary platform is higher on PIQA, HellaSwag, WinoGrande, SocialIQA, MMLU, and OpenBookQA; A100 is higher on ARC-E, BoolQ, RACE, LAMBADA, and TruthfulQA-MC2; ARC-C is identical.

A separate validation-loss check yields $\Delta_{\text{val-loss}} = 0.015$ between the two runs, with the primary platform marginally lower. This loss gap does not propagate to the downstream metric in the same direction or magnitude—a pattern we already document in Appendix E for several attention-output modifications, and which here serves as an additional null control: the loss difference between two baseline trainings on different hardware is smaller than several of the loss–downstream gaps we report on attention-output methods.

This cross-hardware reproduction is the strongest single piece of evidence against an infrastructure-artefact explanation for the three 3B divergences (HybridNorm, SSMax, HyperConnections, §5.3). The baseline reproduces to within 10^{-4} on A100 under the same recipe, while the divergences each have documented theoretical fragility (Post-LN variance growth at depth (Xiong et al., 2020), softmax-replacement logit blowup (Ramapuram et al., 2024), multi-channel EMA drift (Zhu et al., 2024)) that is independent of accelerator vendor. We did not re-run the modified methods at 1.2B on A100 under the same protocol: the cost of replicating a 20-configuration suite on a second hardware brand is comparable to the original 1.2B run. The single-method (baseline) cross-hardware check is intended as a sanity bound on the platform-as-confound hypothesis, not as a full replication.

E Diagnostic Analysis of Significant Failures

Six of the 19 modifications we tested significantly underperform baseline at 1.2B (Table 2). Two are softmax replacements (Sigmoid Attention, SSMax); one is a softmax-subtraction variant (Diff-Attn); and three are residual-connection modifications (AttnRes, LayerScale, HyperConnections).

Method	Category	Params	ΔP	ΔF
baseline	ref	1.217 B	+0.00%	+0.00%
softpick	attention	1.217 B	+0.00%	+0.00%
qknorm	attention	1.217 B	+0.00%	+0.00%
selective_attn	attention	1.192 B	−2.07%	+0.00%
selective_qknorm	attention	1.192 B	−2.07%	+0.00%
value_residual	attention	1.217 B	+0.00%	+0.00%
diff_attn	attention	1.217 B	+0.01%	+0.00%
sigmoid_attn	attention	1.217 B	+0.00%	+0.00%
ssmax	attention	1.217 B	+0.00%	+0.00%
softmax_cap	attention	1.217 B	+0.00%	+0.00%
gated_attn_qknorm	attention	1.218 B	+0.13%	+0.13%
geglu_ffn	ffn	1.217 B	+0.00%	+0.00%
qknorm_geglu	ffn	1.217 B	+0.00%	+0.00%
relu_squared	ffn	1.217 B	+0.00%	+0.00%
sandwich_norm	norm	1.217 B	+0.01%	+0.00%
hybrid_norm	norm	1.217 B	+0.00%	+0.00%
denseformer	residual	1.217 B	+0.00%	+0.00%
layerscale	residual	1.217 B	+0.00%	+0.00%
hyper	residual	1.217 B	+0.00%	+0.00%
attnres	residual	1.217 B	+0.02%	+0.00%

Table 4: Per-method parameter count and per-step training FLOPs relative to baseline, for the 20 configurations in the main comparison. Columns: method, taxonomy category, parameter count, ΔP (% vs baseline params), ΔF (% vs baseline per-step FLOPs). All 19 modifications are within $\pm 0.13\%$ of the baseline on per-step FLOPs; on parameter count, 18 are within $\pm 0.13\%$ and Selective Attention is 2.07% cheaper.

Task	Primary	A100	Δ
piqa	0.7492	0.7448	−0.0044
arc_challenge	0.3933	0.3933	0.0000
arc_easy	0.7226	0.7243	+0.0017
hellaswag	0.5589	0.5564	−0.0025
winogrande	0.5620	0.5406	−0.0213
social_iqa	0.4176	0.4130	−0.0046
mmlu	0.2736	0.2685	−0.0051
openbookqa	0.3840	0.3720	−0.0120
boolq	0.5872	0.6165	+0.0294
race	0.3282	0.3435	+0.0153
lambada_openai	0.4174	0.4203	+0.0029
truthfulqa_mc2	0.3905	0.3919	+0.0014
CLIMB-avg	**0.4820**	**0.4821**	+0.0001

Table 5: 1.2B baseline cross-hardware reproduction. "Primary" denotes the AI accelerator used in the main 1.2B and 3B experiments (vendor anonymized for review); "A100" is an end-to-end re-training on NVIDIA A100 GPUs under bit-identical model code, optimizer recipe, tokenizer, data shuffle, and 23B-token budget. CLIMB-avg differs by 1×10^{-4} between the two runs, and per-task deltas span ± 0.029 with no systematic direction.

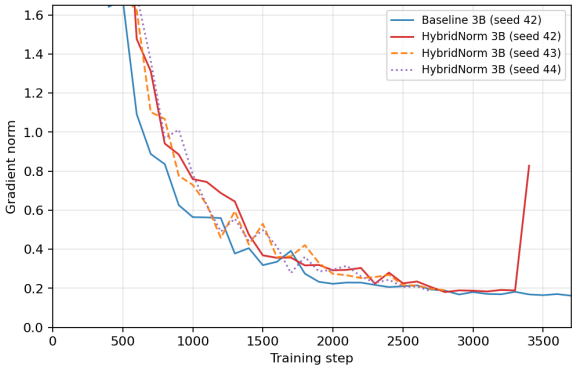

Figure 1: 3B training grad-norm through step 3700 for the three baseline noise-floor seeds. Baseline (blue) tracks a stable ∼0.15–0.20 range. HybridNorm under seeds 42 (red, solid), 43 (orange, dashed), and 44 (violet, dotted) each track the same range before collapsing to NaN; each terminal × marks the last finite step before divergence at steps 3500, 2900, and 2800 respectively (seed 42 first spikes to grad-norm 0.83 at step 3400 before NaN at 3500). All three diverge shortly after the cosine peak. Pre-NaN grad-norm 0.18–0.19 across seeds and a narrow divergence window indicate Post-LN-on-FFN variance growth at 28-layer 3B depth (Xiong et al., 2020) rather than seed noise.

This section examines *how* these failures manifest. Two views are used: final validation loss compared to CLIMB-avg (Figure 2), and the per-task delta heatmap (Figure 3).

E.1 Validation loss does not predict the attention-side failures

Figure 2 plots validation loss against training step for six methods: two significant winners (QK-Norm, Softpick), baseline, and three significant

Method	Seed	NaN step	Pre-NaN grad signature	Mechanism
HybridNorm	42	3,500	0.19 → 0.83 → NaN (1 step)	
HybridNorm	43	2,900	0.19 → NaN (no intermediate spike)	Post-LN FFN variance growth
HybridNorm	44	2,800	0.18 → NaN, then sustained ∼14	
SSMax	42	5,800	monotone rise 27 → 1578 over 1,800 steps	Softmax-replacement logit blowup
HyperConnections	42	18,300	sustained > 100 peaks from step 15,000	Multi-lane EMA residual drift

Table 6: 3B training divergences under the shared iso-recipe. HybridNorm is reported across all three baseline noise-floor seeds (42, 43, 44); all three diverge in a 2800–3500 step window with pre-NaN grad-norm in the 0.18–0.19 range, identical to baseline's stable trajectory. Seed 42 shows a single-step spike visible at 100-step logging; seeds 43 and 44 collapse directly from the stable range without a visible intermediate step. SSMax and HyperConnections are each reported on the single seed that was attempted, with the pre-NaN grad-norm signatures listed above.

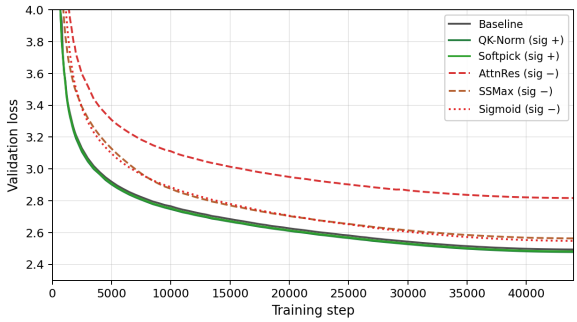

Figure 2: Validation-loss curves for six representative methods at 1.2B over 44,000 training steps. The two attention-side failures (SSMax, Sigmoid Attention) converge to within ∼0.07 nats of baseline—a 2–3% relative gap. The residual-side failure (AttnRes) ends 0.33 nats above baseline. The two improvers (QK-Norm, Softpick) track baseline within visual resolution.

Method	val loss	CLIMB-avg	z
Softpick	2.4814	0.4922	+4.47
QK-Norm	2.4771	0.4882	+2.51
Baseline	2.4890	0.4829	0.00
Sigmoid Attention	2.5474	0.3217	−77.7
SSMax	2.5630	0.4208	−29.9
AttnRes	2.8165	0.4218	−29.5

Table 7: Final validation loss, CLIMB-avg, and bootstrap z-score for six representative methods at 1.2B.

failures. At the end of training the final losses are close for the two attention-side failures but not for AttnRes (Table 7).

Sigmoid Attention's validation loss exceeds baseline's by only 2.4%, and SSMax's by 3.0%. A pretraining-perplexity ranking—the primary metric in Narang et al. (2021)—would place both methods near baseline. Their CLIMB-avg scores, however, drop by 16 and 6 points respectively, on a scale where the entire 1.2B range across our 20 configurations is roughly three points.

The loss-downstream decoupling observed here is therefore specific to attention-output modifications: Sigmoid Attention and SSMax reach near-baseline losses while failing downstream. The third failure in the figure, AttnRes, behaves differently. Its validation loss is 0.33 nats above baseline (13% relative) and its CLIMB-avg drop tracks that loss gap; a loss-based ranking would correctly flag AttnRes as substantially worse than baseline. The decoupling claim is not a universal loss-is-unreliable statement—it is a scoped observation that methods that replace or post-process the attention-output distribution can reach low loss without transferring that low loss to downstream accuracy. This narrows but does not contradict Tay et al. (2023), who document a smaller-magnitude version of the same gap across ten architecture families (15M to 40B parameters). The magnitude observed here (up to 16 CLIMB-points) is several times the gap they document.

E.2 Failure signatures

Sigmoid Attention: LAMBADA drop. The accuracy drop for Sigmoid Attention is not uniform across tasks (Figure 3). LAMBADA (OpenAI) falls from baseline's 0.42 to near 0; the other 11 tasks each lose 0.03–0.08 points. LAMBADA requires predicting the final word of a passage given its preceding narrative, which requires attention to concentrate on a specific retrieval-relevant token. Without softmax's normalization, Sigmoid Attention cannot guarantee that attention weights sum to one, and in practice produces more diffuse attention distributions that fail to select a single key position. General language modelling behaves adequately—the conditional distribution over next tokens is still usable for most CLIMB tasks—but this specific kind of look-up fails.

Figure 3: Per-task CLIMB deltas vs baseline at 1.2B. Rows are methods sorted by CLIMB-avg; columns are the 12 CLIMB tasks. Red cells indicate accuracy below baseline, blue above.

SSMax: uniform degradation. SSMax's drops are smaller but uniform: 0.014–0.025 points on nearly every task (Figure 3, row SSMax). SSMax replaces softmax with a variant that includes an additional learned per-head temperature intended to prevent entropy collapse at long contexts (Nakanishi, 2025). In our 1024-token-context, 1.2B setting the temperature does not stabilize near softmax's effective value; the result is a small but consistent drift of attention away from the per-task optimum. No single task shows a localized drop, but the aggregate drop is large.

AttnRes: broad regression, not a local bug. AttnRes (Kimi Team et al., 2026) replaces the identity residual after attention with a learned cross-layer mixture, weighted by a pseudo-query dot-producted against previous layers' outputs. Our implementation reproduces Equation 2 and Table 4 row 2 of the original paper (Appendix B.1); the $z = -29.5$ result is not attributable to a coding error. Unlike Sigmoid Attention and SSMax, AttnRes's validation loss also rises substantially (0.33 nats higher than baseline), and its CLIMB drops track the loss gap. The model is globally worse at next-token prediction, not subtly misbehaving on a specific downstream skill. Kimi Team et al. (2026) report gains at larger MoE scale on GPQA; our result scopes that claim: at 1.2B dense on CLIMB-12, pseudo-query cross-layer residual does not help.

Diff-Attn: subtraction of two softmaxes. Differential attention (Ye et al., 2024) computes two softmax distributions per head and subtracts one from the other. Each individual softmax is preserved, but the subtraction produces un-normalized attention weights that need not sum to one and can be negative; under the operational criterion of §3.1 it is therefore *hard*. The 1.2B result ($z = -2.45$) is significantly below baseline. One plausible mechanism is that when the two subtracted heads become correlated during training, the subtraction amplifies noise rather than cancelling distractors, but the single-seed experiment reported here cannot

distinguish this from alternative explanations.

LayerScale: uniform degradation, mechanism unresolved. LayerScale (Touvron et al., 2021) multiplies the sublayer output by a learnable per-channel gate γ initialized at 10^{-4}, giving $x_{\ell+1} = x_\ell + \gamma \odot f(x_\ell)$. The small-init design intends to let deep networks train stably by starting near the identity and learning per-layer contribution magnitudes. In the 1.2B setting the $z = -4.42$ degradation is accompanied by a roughly uniform 0.01–0.02 drop across CLIMB tasks (Figure 3), rather than a task-localized failure. The failure signature is therefore "global under-performance" rather than a specific downstream skill loss. One possible mechanism is that under a fixed optimizer recipe tuned for the identity residual, the small-init residual branch trains more slowly and does not close the gap within 23B tokens; CaiT reports gains at hyperparameters co-tuned with the LayerScale init value, which is not varied here. Distinguishing this hypothesis from alternatives (effective-depth change, initialization interacting with weight decay, etc.) would require an init sweep that is left to future work.

HyperConnections: multi-lane overhead at dense 1.2B. HyperConnections (Zhu et al., 2024) replaces the single residual stream with n parallel lanes at different update rates, mixed at both the sublayer input (matrix A) and output (matrices R, D). The original paper reports gains on a 671B-parameter MoE and shows increasing benefit at larger scale. Our implementation uses the n=2 variant with slow-lane blend rate β and output weight α initialized so that the module reduces to standard residual at step 0. Despite that identity-preserving initialization, the 1.2B dense run degrades uniformly across CLIMB tasks with $z = -9.79$ (Figure 3, row HyperConnections). Three conditions present in Zhu et al. (2024)'s 671B setting are absent from ours: (i) mixture-of-experts routing, which changes what a residual stream needs to carry; (ii) model depth of ∼60 layers, where the lane mechanism has more room to differentiate; (iii) separate warmup for the lane-mixing parameters. The result here is consistent with a scale-dependent benefit that 24-layer dense 1.2B does not capture, rather than a failure of the method in general.

E.3 A common thread: sharp attention and matched recipe

Two patterns emerge across the six failure signatures above. The three attention-side failures (Sigmoid Attention, SSMax, Diff-Attn) each reduce softmax's ability to concentrate attention on a single key position, and the CLIMB tasks that require such concentration—LAMBADA and, to a lesser degree, ARC-Easy and HellaSwag—show the largest per-method drops in Figure 3. The three methods that significantly improve over baseline (QK-Norm, Selective Attention, Softpick) all preserve this concentration ability while adding regularization or filtering. The pattern is not airtight: Diff-Attn preserves individual softmaxes but still fails, suggesting that preserving the softmax computation is necessary but not sufficient to preserve the output attention distribution's useful properties.

The three residual-side failures (AttnRes, LayerScale, HyperConnections) share a different signature: each is reported to improve a *different* base architecture or scale—671B MoE, larger-MoE GPQA, co-tuned CaiT—and each degrades uniformly rather than task-specifically when transplanted into 1.2B dense under a fixed recipe. This is the scale-and-recipe sensitivity documented by Narang et al. (2021): a modification that helps under its authors' setup does not automatically help under a different scale and an unmodified optimizer recipe.

E.4 Per-task statistical cross-checks

The bootstrap z-scores in Table 2 aggregate across the 12 CLIMB-12 tasks via the macro-average. As a complementary cross-check that does not depend on the macro-average or the bootstrap reference distribution, we compute per-task Welch's t-tests against the three baseline seeds and combine the 12 per-task one-sided p-values via Stouffer's $Z = \sum_i \Phi^{-1}(1-p_i)/\sqrt{12}$, treating each task as an independent piece of evidence. For QK-Norm this gives Z=+2.68, one-sided p=0.0037 (two-sided 0.0074); 9 of 12 tasks have higher QK-Norm mean than baseline (the three exceptions: PIQA, ARC-C, SocialIQA). For Softpick the corresponding combination is Z=+2.85, one-sided p=0.0022 (two-sided 0.0044), with 11 of 12 task means above baseline (only SocialIQA below). Stouffer combination is conservative under positively-correlated tasks, so the p-values are upper bounds on the true significance. Both methods remain significant under

this alternative aggregation, providing independent evidence beyond the macro-average bootstrap and the 3-seed Welch's t on CLIMB-avg reported in §3.3.