CONTEXTUAL CONTROL OF TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Computational neuroscience models show that gain modulation can alter neural computation while connectivity remains fixed. Inspired by this principle, we ask whether context can similarly provide an explicit control signal over computation in a pretrained Transformer while its weights remain fixed. We introduce contextual control, a lightweight learned pathway that maps contextual states to bounded gains over feed-forward computation. Because the control signal is explicit, we can manipulate it while holding the input and prediction target fixed. Across GPT-2 and Qwen, contextual control improves prediction beyond static and token-only controls. We then ask whether the learned signal is merely context-dependent or whether its assignment to a particular context matters. Moving control between occurrences of the same token in different contexts changes prediction, and in pre-$\tanh$ control coordinates its absolute effect exceeds that of perturbations matched in layer-wise magnitude on both backbones. The assigned control also contributes positively to prediction on average, as neutralizing it increases loss. Together, these interventions show that contextual assignment has causal consequences within the learned control interface, providing an explicit framework for studying how context regulates computation in frozen Transformers.

1 INTRODUCTION

Transformers are context-dependent by construction: the representation of a token changes with what precedes it, allowing the same token to support different predictions in different contexts. Yet this dependence is expressed through internal representations rather than a separate control variable that can be manipulated to test how context regulates subsequent computation. We ask whether contextual information can be made into such a control signal and whether intervening on it reveals consequences of its contextual assignment.

Our motivation comes from computational neuroscience. Gain modulation, a change in neural response amplitude without a change in selectivity, has been studied as a general computational principle (Salinas & Thier, 2000). In a recurrent model of motor cortex, movement-specific gain patterns learned while connectivity remains fixed select different outputs from the same network (Stroud et al., 2018). We borrow this computational principle: separating a fixed computational substrate from a signal that modulates how its computation is expressed.

We instantiate this idea as *contextual control* (Figure 1): a lightweight learned controller added to a pretrained Transformer whose weights remain frozen. At each layer and token position, the controller maps the state entering the feed-forward block to bounded multiplicative gains over groups of intermediate activations. Zero control exactly recovers the frozen computation, making the signal explicit and intervention-accessible: it can be recorded, replayed, reassigned, perturbed, or removed while the recipient text and prediction target remain fixed.

Conditional modulation itself is well established. FiLM conditions feature-wise transformations on side information (Perez et al., 2018), IA^3 learns multiplicative rescaling vectors in frozen language models (Liu et al., 2022), and related approaches intervene on representations or activations for adaptation, steering, and causal analysis (Wu et al., 2024; Stoehr et al., 2024; Vig et al., 2020). Our focus is different: we treat a learned context-derived modulatory signal as an experimental variable and ask whether its assignment to a particular context matters.

We evaluate this question on GPT-2 and Qwen2.5 using WikiText-2 and PG-19. Contextual control improves prediction beyond static and near-parameter-matched token-only controls. We then

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

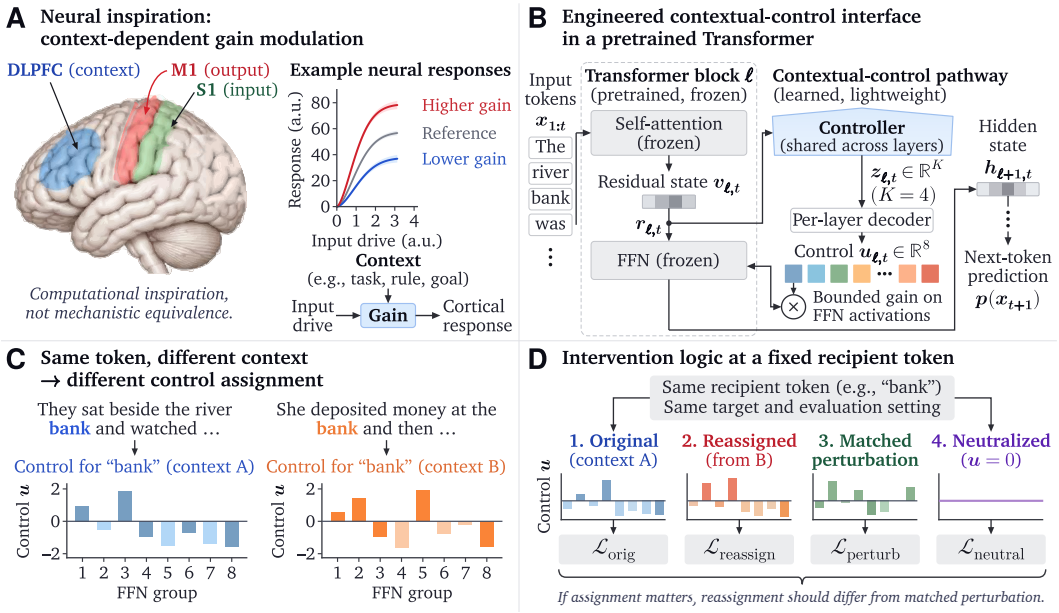

Figure 1: Contextual control of Transformer computation. (A) Gain modulation motivates regulating computation without changing connectivity; this is computational inspiration, not an anatomical correspondence. **(B)** A learned controller maps contextual states to bounded group-wise gains over frozen feed-forward computation. **(C)** The same token in different contexts can receive different controls; the displayed $\boldsymbol{u}$ values are illustrative. **(D)** Reassignment, matched perturbation, and neutralization intervene on control while holding the recipient token and prediction target fixed.

reassign control between occurrences of the same token in different contexts while holding the recipient text and prediction target fixed, and compare the resulting prediction changes with matched perturbations. In pre-tanh control coordinates, the absolute reassignment effect exceeds that of perturbations matched in layer-wise magnitude on both backbones. Complementarily, neutralizing the assigned control increases prediction loss on average.

Our contributions are threefold:

1. We introduce contextual control, an explicit, bounded interface over frozen Transformer computation with an exact neutral point, enabling direct intervention without changing pretrained weights.
2. We show that contextual assignment has measurable consequences: same-token control reassignment changes prediction and, in pre-tanh coordinates, produces a larger absolute effect than perturbations matched in layer-wise magnitude on both GPT-2 and Qwen.
3. We show that assigned control contributes positively to prediction on average and characterize how this contribution varies across corpora and context positions.

Together, these results establish contextual control as an intervention-accessible framework for studying how context regulates computation in frozen Transformers.

2 Contextual Control of Frozen Computation

Site and actuator. Consider a pre-norm Transformer block at layer ℓ and position t, with block input $\boldsymbol{h}_{\ell,t}$ and residual state $\boldsymbol{v}_{\ell,t}$ after attention. The feed-forward block reads $\boldsymbol{r}_{\ell,t} = \mathrm{Norm}_\ell(\boldsymbol{v}_{\ell,t})$ and produces an intermediate activation $\boldsymbol{f}_{\ell,t} \in \mathbb{R}^D$: for GPT-2 small (Radford et al., 2019), $\boldsymbol{f}_{\ell,t} = \mathrm{GELU}_{\tanh}(W_{1,\ell}\boldsymbol{r}_{\ell,t} + \boldsymbol{b}_{1,\ell})$, $D = 3072$; for Qwen2.5-0.5B (Qwen Team et al., 2024), whose feed-forward block is bias-free ($\boldsymbol{b}_{2,\ell} = 0$), $\boldsymbol{f}_{\ell,t} = \mathrm{SiLU}(W_{\mathrm{g},\ell}\boldsymbol{r}_{\ell,t}) \odot (W_{\mathrm{u},\ell}\boldsymbol{r}_{\ell,t})$, $D = 4864$. We modulate this intermediate activation before the frozen output projection: the controlled block computes

$$\boldsymbol{h}_{\ell+1,t} = \boldsymbol{v}_{\ell,t} + W_{2,\ell}\big[\boldsymbol{g}_{\ell,t} \odot \boldsymbol{f}_{\ell,t}\big] + \boldsymbol{b}_{2,\ell}, \qquad \boldsymbol{g}_{\ell,t} = \mathbf{1}_D + P\boldsymbol{a}_{\ell,t}, \qquad \boldsymbol{a}_{\ell,t} = \alpha \tanh(\boldsymbol{u}_{\ell,t}), \quad (1)$$

where $\boldsymbol{u}_{\ell,t} \in \mathbb{R}^G$ is the pre-$\tanh$ control, $G = 8$, and fixed $P \in \{0,1\}^{D\times G}$ assigns channels to eight contiguous groups (384 channels in GPT-2, 608 in Qwen). With $\alpha = 0.25$, gains lie in $[0.75, 1.25]$. All groups execute; this is modulation, not routing. Setting $\boldsymbol{u}_{\ell,t} = 0$ gives unit gains and exactly recovers the frozen forward pass (Appendix D).

Controller. A two-layer encoder shared across layers and positions, with an additive per-layer embedding $\boldsymbol{e}_\ell$, maps $\boldsymbol{r}_{\ell,t}$ to a K-dimensional code that is RMS-scaled (Zhang & Sennrich, 2019) and decoded per layer:

$$\boldsymbol{q}_{\ell,t} = W_b\ \mathrm{GELU}(W_a \boldsymbol{r}_{\ell,t} + \boldsymbol{b}_a + \boldsymbol{e}_\ell) + \boldsymbol{b}_b, \quad W_a \in \mathbb{R}^{128\times d},\ \boldsymbol{b}_a, \boldsymbol{e}_\ell \in \mathbb{R}^{128},\ W_b \in \mathbb{R}^{K\times 128}, \tag{2}$$

$$\boldsymbol{z}_{\ell,t} = \gamma_\ell \odot \boldsymbol{q}_{\ell,t} \Big/ \sqrt{\tfrac{1}{K}\textstyle\sum_{i=1}^{K} q_{\ell,t,i}^2 + \epsilon}, \qquad \boldsymbol{b}_b, \gamma_\ell \in \mathbb{R}^K, \tag{3}$$

$$\boldsymbol{u}_{\ell,t} = S_\ell \boldsymbol{z}_{\ell,t} + \boldsymbol{b}_\ell, \qquad S_\ell \in \mathbb{R}^{G\times K},\ \boldsymbol{b}_\ell \in \mathbb{R}^G. \tag{4}$$

We use $K = 4$ throughout the principal experiments. The code is computed independently at each layer and position from the causal state at t; context dependence is thus built into the controller, while Sections 4–6 test whether context access improves prediction and whether a particular assignment matters under intervention. The controller has 101,012 trainable parameters on GPT-2 and 119,460 on Qwen (Appendix A). Interventions are defined on $\boldsymbol{u}$, whose coordinates correspond to layer-group control values; the latent code $\boldsymbol{z}$ is not uniquely identified under transformations absorbed by its scale and decoder. Within each layer, any controller-generated difference in $\boldsymbol{u}$ lies in $\mathrm{col}(S_\ell)$, of dimension at most K; this is a property of the added controller, not of the frozen backbone.

Context-blind baselines. Two baselines use the same actuator but do not access preceding context. *Static* control learns one $\boldsymbol{u} \in \mathbb{R}^{L\times G}$ shared across positions (96 parameters for GPT-2; 192 for Qwen). *Token-only* controllers derive $\boldsymbol{u}$ from the current token identity alone through the same RMS scale and decoder, using a rank-2 vocabulary table for GPT-2 (101,054 parameters) and either a two-hash Bloom embedding (119,464) or a rank-3 vocabulary table (117,490) for Qwen (Appendix A). These baselines test whether context access improves prediction at comparable parameter budgets; assignment specificity requires intervention.

Control as an intervention variable. Let $\mathbf{U} \in \mathbb{R}^{L\times T\times G}$ be the control field for a sequence and $\mathbf{U}_t \in \mathbb{R}^{L\times G}$ its column at position t. An intervention replaces $\mathbf{U}_t$ by $\mathbf{U}'_t$ while holding the recipient text, position, and prediction target fixed. Earlier positions are unchanged by causality; later states are recomputed. We score only the next-token loss ℓ_t at the intervened position:

$$d = \ell_t\big(x; \mathbf{U}^{(t\leftarrow \mathbf{U}'_t)}\big) - \ell_t(x; \mathbf{U}), \tag{5}$$

Replay reinserts the recorded original column and yields $d = 0$. The donor column $\boldsymbol{u}_t^{\mathrm{don}}$ is the column the same controller produced for the same token in another context. The isotropic comparator

$$\boldsymbol{u}_t^{\mathrm{rnd}} = \boldsymbol{u}_t^{\mathrm{orig}} + \|\boldsymbol{u}_t^{\mathrm{don}} - \boldsymbol{u}_t^{\mathrm{orig}}\|_2\, \boldsymbol{\delta}/\|\boldsymbol{\delta}\|_2, \qquad \boldsymbol{\delta} \sim \mathcal{N}(\mathbf{0}, I_{LG}), \tag{6}$$

matches the donor displacement's total norm in pre-$\tanh$ coordinates; Section 5 introduces stronger comparators that also match layer-wise magnitude and decoder support. Neutralization sets $\mathbf{U}_t = 0$, yielding unit gains. All causal claims in this paper refer to interventions on this learned control interface.

3 Experimental Design

Backbones and data. We study frozen GPT-2 small (Radford et al., 2019) and Qwen2.5-0.5B (Qwen Team et al., 2024) on WikiText-2-raw (Merity et al., 2017) and PG-19 (Rae et al., 2020). PG-19 supplies bounded book slices truncated to match WikiText-2 token counts, providing a second domain rather than a long-document benchmark (Appendix B). Each split is tokenized as a single stream and divided into independent non-overlapping 512-token blocks, each scoring 511 next-token predictions.

Training and selection. All controllers and baselines in Sections 4–6 use AdamW (Loshchilov & Hutter, 2019) for 2,000 updates of eight 512-token blocks, with learning rate 0.003, no weight decay, a constant schedule, gradient clipping at 1.0, and checkpoint selection every 100 updates on validation blocks 0–79. Each arm has three seeds except the single-seed GPT-2 PG-19 static and token-only baselines. The protocol was fixed before the present controllers were trained, with hyperparameters carried over from an earlier version (Section 9). Throughout, pre-specified means fixed before the results it governs, and pre-registered means written down before execution with a named primary endpoint (order of commitments in Appendix D).

Qwen cohorts. Two Qwen training cohorts use the same protocol but different token-only baselines: a two-hash Bloom embedding (Bloom cohort) and a vocabulary table (vocabulary-table cohort). The vocabulary-table cohort supplies its Section 4 comparison, the Section 6 neutralization summary, and the initial Qwen reassignment cohort; the Bloom cohort supplies the remaining Qwen results, including the follow-up reassignment cohort and the Section 7 analyses. The cohorts use identical token blocks and seeds, are never pooled, and reach near-identical contextual losses; implementation details are in Appendix D.

Data separation. Validation blocks 0–79 are used only for checkpoint selection. Blocks 80–139 were reserved initially for neutralization and blocks from 140 onward for reassignment pairs. Later analyses reuse validation text: the diagnostics and capacity sweep of Section 7 use blocks 80–279, the Qwen follow-up cohort was mined from the same blocks, the later pre-specified studies read blocks 280–511 or previously unused PG-19 windows, and WikiText-2 blocks 280–511 also hold pairs of the initial Qwen cohort. The test split never selected checkpoints or intervention designs, but Qwen test comparisons determined whether additional seeds and the initial Qwen reassignment study were run (Appendix D).

Reassignment cohorts. Each pair comprises two non-overlapping 192-token WikiText-2 validation windows ending in the same token but drawn from different articles; each member serves once as recipient and is scored on its own next token. The initial GPT-2 cohort contains 600 pairs from four surface-form families, fixed before intervention. The expanded GPT-2 cohort contains 2,500 pairs under a broader token rule specified after the initial result. The initial Qwen cohort applies this broader rule to 2,500 pairs using vocabulary-table controllers. A separate 2,500-pair Qwen follow-up uses Bloom-cohort controllers and was pre-registered with its comparators, primary endpoint, and decision criterion before execution. Initial cohorts received only the total-norm isotropic comparator; geometry-matched comparators were introduced afterward and evaluated prospectively on the expanded GPT-2 and Qwen follow-up cohorts.

Estimands. Let d^{don} and d^{null} denote the loss changes under donor and comparator columns. We report the donor change d^{don}, the absolute contrast $A = |d^{\mathrm{don}}| - |d^{\mathrm{null}}|$, and the signed contrast $C = d^{\mathrm{don}} - d^{\mathrm{null}}$. Positive values indicate, respectively, increased donor loss, a larger absolute donor effect, and a larger signed donor effect. Absolute values are taken per direction and comparator draw; draws, directions, and controllers are then averaged within each pair, the unit of analysis. Primary endpoints were A for the initial and expanded GPT-2 analyses, A and C for the pre-registered GPT-2 geometry follow-up, and C for the Qwen follow-up, with A secondary; the initial Qwen cohort named no primary endpoint. Neutralization defines b_t as the loss change after setting the control at one position to zero.

Uncertainty. Reassignment intervals use dyadic cluster-robust variances over shared source articles (Aronow et al., 2015) with t-based 95% intervals; Appendix C gives the estimator, article counts, robustness checks, and the handling of negative variance estimates. Neutralization intervals are clustered by article and are indicative. All intervals are marginal, unadjusted for multiplicity, and conditional on the trained controllers. The dyadic procedure was adopted after early GPT-2 results and fixed for subsequent analyses; the Qwen follow-up pre-registered it.

4 A Functional Contextual-Control Interface

We first ask whether access to preceding context improves prediction. The key comparison is contextual versus token-only control: both use the same actuator and similar parameter budgets, but

Table 1: Test perplexity for contextual (bold) and context-blind controls. Multi-seed entries show mean perplexity (seed SD); reductions are relative to the frozen backbone within each setting, and comparisons are within tokenizer only. Token-only constructions are defined in Section 2; per-seed values are in Appendix B.

Backbone / corpus	Condition	Params	Seeds	Test PPL (SD)	Red.
GPT-2 / WikiText-2	frozen	0	–	35.4833	–
	static gains	96	3	33.7765 (0.00004)	4.81%
	token-only, rank-2	101,054	3	31.3009 (0.0403)	11.79%
	contextual	101,012	3	**28.7192** (0.0347)	**19.06%**
GPT-2 / PG-19 in-domain	frozen	0	–	47.2968	–
	static gains	96	1	35.3721	25.21%
	token-only, rank-2	101,054	1	33.6071	28.94%
	contextual	101,012	3	**32.3717** (0.0195)	**31.56%**
Qwen / WikiText-2	frozen	0	–	17.1921	–
	static gains	192	3	17.0108 (0.0003)	1.05%
	token-only, Bloom	119,464	3	16.3330 (0.0271)	5.00%
	contextual	119,460	3	**15.1387** (0.0250)	**11.94%**
Qwen / PG-19 in-domain	frozen	0	–	24.3604	–
	static gains	192	3	24.3274 (0.0001)	0.14%
	token-only, Bloom	119,464	3	24.2858 (0.0070)	0.31%
	contextual	119,460	3	**24.0758** (0.0089)	**1.17%**

Table 2: Absolute reassignment contrast against matched comparators. $A = |d^{\mathrm{don}}| - |d^{\mathrm{null}}|$; $A > 0$ means the donor changes loss more in absolute value than the comparator. Rows 1–4 progressively match the donor displacement in pre-$\tanh$ coordinates; S matches the post-$\tanh$ gain displacement (approximately on GPT-2; Appendix E). Values are three-seed pair means in nats with dyadic 95% intervals; Qwen A is the pre-registered secondary endpoint.

	Direction	Magnitude matched	GPT-2, expanded cohort	Qwen, follow-up cohort
1	isotropic	total norm	+0.0329 [+0.0278, +0.0380]	−0.0088 [−0.0127, −0.0048]
2	isotropic	per-layer norms	not run	+0.0082 [+0.0060, +0.0104]
3	decoder column space	per-layer norms	+0.0308 [+0.0259, +0.0356]	+0.0077 [+0.0053, +0.0101]
4	real displacement	per-layer norms	+0.0241 [+0.0197, +0.0284]	+0.0056 [+0.0038, +0.0075]
S	decoder column space	per-layer post-$\tanh$ gain	+0.0179 [+0.0128, +0.0230]	+0.0023 [−0.00004, +0.0046]

only the contextual controller reads the causal state at each position; frozen and static models provide reference points (Table 1; Appendix Figure 4). Contextual control outperforms token-only control in every seed and setting evaluated (the GPT-2 PG-19 token-only baseline has one seed). On GPT-2 WikiText-2, test perplexity decreases from 31.3009 to 28.7192, an 8.25% reduction (paired gap 0.08608 nats, block-bootstrap 95% interval [0.08359, 0.08868]); on Qwen WikiText-2, it decreases from 16.3330 to 15.1387, with the vocabulary-table baseline yielding the same conclusion (Appendix B). A GPT-2 token-only table with twice the parameters does not improve performance, arguing against insufficient token-only capacity under the evaluated protocol.

The advantage is smaller on PG-19: 0.037 nats for GPT-2 (no interval) and 0.00868 ± 0.00065 nats (mean $\pm$ seed SD) for Qwen, positive in every seed but 8.7 times smaller than the Qwen WikiText-2 gap (0.0759 ± 0.0032 nats). Section 7 examines this boundary condition.

A rank-1 LoRA with slightly fewer parameters adapts better in our comparison (Appendix K); our contribution is instead an explicit, bounded control variable that supports direct intervention.

5 Consequences of Reassigning Control Between Contexts

Section 4 showed that access to preceding context improves prediction. We now ask whether the particular control assigned to a context matters. Holding the recipient's text, token, position, and prediction target fixed, we replace its control with that produced by the same controller for the same token in another context. Because any perturbation can change prediction, we compare donor reassignment with increasingly matched alternatives (Table 2): an isotropic perturbation matched in total norm; perturbations matched in per-layer magnitude, either isotropic or restricted to the decoder column space; and another real controller-generated displacement matched per layer. A sensitivity analysis instead matches displacement in post-$\tanh$ gain space.

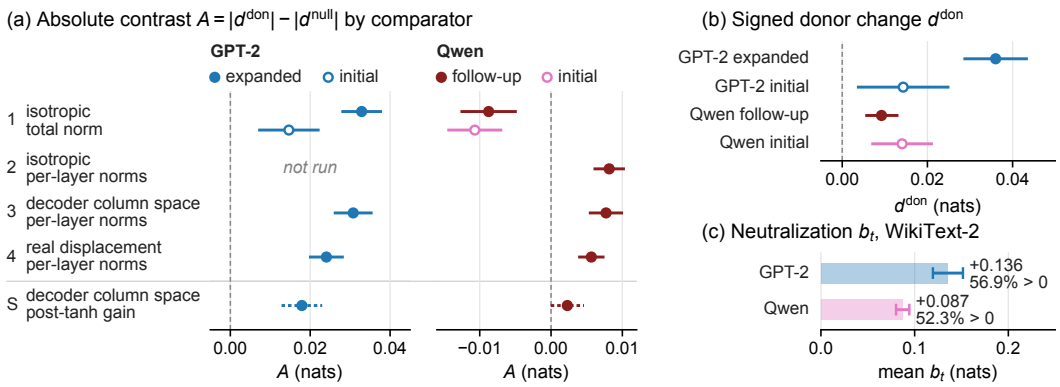

Figure 2: Reassignment and single-position neutralization. (a) Absolute contrast A across the comparators of Table 2; filled markers show the expanded and follow-up cohorts and open markers the initial cohorts, evaluated only against row 1 (Qwen A: secondary endpoint). Axes differ by backbone. **(b)** Signed donor change d^{don}. **(c)** Mean neutralization effect b_t and fraction of positions with $b_t > 0$. 95% intervals: dyadic over source articles; pair bootstrap for the Qwen follow-up in (b); article-clustered in (c) (indicative).

Donor reassignment raises loss on average in all four cohorts (Figure 2b), but so do the comparators; reassignment alone therefore establishes an effect, not donor specificity. The initial 600-pair GPT-2 cohort, fixed before intervention, gives an absolute contrast of the same sign as the expanded 2,500-pair cohort but roughly half as large in every seed, for reasons not identified (Appendix E).

Matching total norm alone leaves layer-wise displacement unmatched. Under this comparator, the donor's absolute effect is larger on GPT-2 but smaller on Qwen. On Qwen, this reversal disappears when displacement magnitude is matched within each layer, and restricting the comparator further to the decoder column space changes the result little (Table 2). The reversal therefore appears to arise primarily from unmatched allocation of displacement across layers rather than from directions outside the decoder column space.

With layer-wise magnitude matched in pre-tanh control coordinates, the donor produces a larger absolute loss change than both decoder-subspace and real-displacement comparators on GPT-2 and Qwen (Figure 2a; Table 2). This conclusion depends on the coordinate in which magnitude is matched. Under post-tanh gain-space matching, 3.46–4.29% of GPT-2 layer cells reach the scaling cap and remain under-matched (Appendix E); the GPT-2 contrast remains positive but smaller, whereas the Qwen secondary contrast includes zero. Thus the cross-backbone absolute effect is established for the pre-tanh comparisons, not generally for gain-space matching.

The signed donor-minus-comparator effect does not generalize across backbones. On GPT-2, the pooled signed contrast is positive against both geometry-matched comparators under the pre-registered follow-up criterion, although per-controller evidence is less stable and the effect disappears under post-tanh matching. On the Qwen follow-up, where the signed contrast was the pre-registered primary endpoint, the corresponding results are unresolved: all three-seed signed contrasts are negative, and the intervals against the total-norm isotropic and real-displacement comparators lie below zero (Table 6; Appendix E). Donor reassignment is therefore not uniformly harmful.

Together, these interventions show that, within the learned interface and in pre-tanh control coordinates, the magnitude of the prediction change depends on which context-derived control displacement is applied, beyond its overall magnitude and layer-wise allocation. They do not establish that the original assignment is optimal, that donor reassignment is generally harmful, or that the residual difference reflects semantic structure.

6 Removing the Assigned Control Raises Loss on Average

Section 5 compared alternative control assignments; neutralization instead asks what the assigned control contributes relative to unit gains. At one position, we set the control column to zero while

leaving control elsewhere unchanged, and measure the resulting loss change b_t (Section 3). Because the reference is unit gains, b_t includes any context-independent part of the control.

Neutralization raises loss on average on both backbones (Figure 2c). On WikiText-2 validation blocks 80–139 (30,660 positions from 11 articles; b_t averaged over the three controllers at each position), mean b_t is +0.1355 nats for GPT-2 (article-clustered 95% interval [+0.1195, +0.1516]) and +0.0873 for the Qwen vocabulary-table cohort ([+0.0801, +0.0944]). The effect is heterogeneous: the medians are +0.0137 and +0.0005 nats, and the original control lowers loss at 56.9% and 52.3% of positions, respectively. Thus assigned control contributes positively on average but is not beneficial at every position. Because control remains unchanged elsewhere, b_t is a position-local effect, not a decomposition of the corpus-level predictive gain.

Additional analyses show that neutralization effects are non-additive across layers and partly reproducible across independently trained controllers (Appendix F). These observations characterize the learned interface but do not identify interactions among layers or a unique solution.

7 Boundary Conditions Across Corpora and Context Lengths

The contextual advantage is smaller on PG-19 than on WikiText-2 for both backbones and, on Qwen, about 8.7 times smaller under the same controller and training protocol (Section 4). We examine this attenuation using the Bloom-cohort Qwen controllers on validation blocks 80–279 (the diagnostic region); unless noted as pre-specified, the analyses below are post hoc.

On Qwen–PG-19, the benefit is concentrated near the start of each 512-token block. Removing all learned control increases loss by 0.0715 nats at $t < 64$, but only 0.0050 at $64 \leq t < 128$ and 0.0003 at $t \geq 256$; WikiText-2 retains a substantial effect at later positions (Figure 3a). Single-position neutralization shows the same attenuation (Figure 3b). A pre-specified splice test found no recovery of the early benefit after artificial cross-book boundaries, while an exploratory analysis found elevated benefit near starts of retained book slices; WikiText-2 instead shows strong header-adjacent concentration with a persistent effect far from headers, confirmed under a pre-specified criterion on previously unused blocks (Appendix J). These observations associate the benefit with context structure but do not identify its cause.

The attenuation is not explained by lack of leverage at the control site. Random gain perturbations affect the frozen model similarly on both corpora across position bins (Figure 3c), while reducing the learned PG-19 control at later positions slightly improves loss. Thus the site remains capable of influencing prediction, but the learned later-position control is not correspondingly useful.

Increasing controller width does not clearly resolve the boundary. In a pre-specified sweep, $K = 16$ and $K = 64$ increase the mean $t \geq 64$ benefit only from 0.00213 to 0.00361 and 0.00394 nats, below the pre-specified 0.010-nat criterion (Appendix H). Because group resolution and gain range remain fixed, this sweep neither establishes nor excludes controller capacity as the limiting factor.

Changing training-data exposure helps only partially. Retraining on 16,000 unique PG-19 blocks from 1,118 documents raises the $t \geq 64$ advantage over token-only control from +0.00241 to +0.00510 nats on reserved validation blocks and from +0.00211 to +0.00414 on a further held-out set, but the pre-specified criteria are only partly met (Appendix I). A follow-up decomposition does not isolate document diversity from repetition. Thus the PG-19 attenuation is not explained by control-site leverage, and wider codes or altered training exposure provide at most partial relief; its cause remains unidentified.

8 Related Work

Multiplicative and conditional modulation are well established. FiLM applies feature-wise affine transformations conditioned on side information (Perez et al., 2018), while IA^3 learns input-independent rescaling vectors for frozen language models (Liu et al., 2022). HyperNetworks generate parameter modifications from the current input and state (Ha et al., 2017), and related mechanisms can make adaptation input-dependent (Zadouri et al., 2024). Other methods adapt or intervene on internal representations: adapters and LoRA modify frozen models through trainable modules or low-rank weight updates (Houlsby et al., 2019; Hu et al., 2022), ReFT learns low-rank represen-

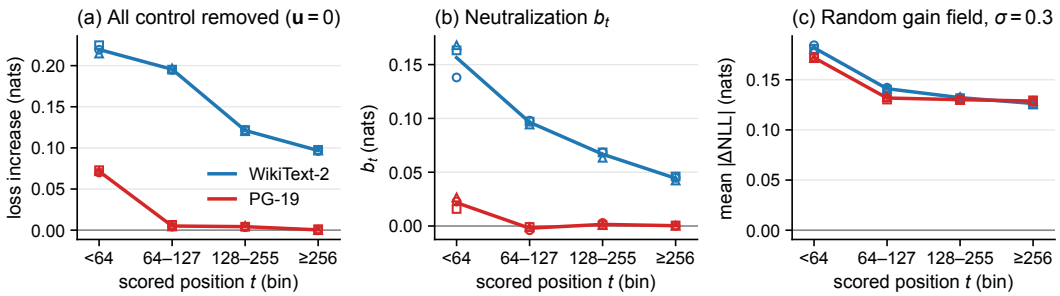

Figure 3: Context-position dependence of control benefit and perturbation sensitivity. Qwen Bloom-cohort controllers on WikiText-2 and PG-19 (validation blocks 80–279; lines: three-seed means; markers: seeds). **(a)** Loss increase when all learned control is removed. **(b)** Single-position neutralization b_t at every eighth position of blocks 80–179. **(c)** Mean absolute loss change of the frozen model under a random gain field, $\boldsymbol{u}$ i.i.d. $\mathcal{N}(0, \sigma^2)$ over positions, layers and groups, $\sigma = 0.3$ (three draws). Bins denote the scored position within each 512-token block.

tation edits (Wu et al., 2024), and activation steering directly modifies hidden activations (Stoehr et al., 2024; Lee et al., 2025).

Prior work has also intervened directly on learned modulatory signals: FiLM parameters were ablated and perturbed at test time (Perez et al., 2018), while Vecoven et al. (2020) froze a learned neuromodulatory signal across a task change. Activation patching similarly studies internal variables by replacing their values across inputs (Zhang & Nanda, 2024). Our focus is different: contextual control introduces an explicit, bounded signal whose neutral value exactly recovers the frozen model, and tests its contextual assignment through same-token reassignment against matched perturbations and single-position neutralization. These claims concern the added control interface rather than the frozen backbone's native circuitry. Unlike routing methods that skip computation (Raposo et al., 2024), contextual control uses bounded gains while always executing the feed-forward block.

9 Discussion and Limitations

Contextual control separates three questions that are otherwise difficult to distinguish: whether access to context improves prediction, whether the magnitude of the prediction change depends on which context-derived control is applied, and whether the assigned control contributes relative to unit gains. Sections 4–6 provide evidence for each within the learned interface. These results do not imply that the pretrained backbone contains an analogous controller or computes in the controller's imposed $K = 4$ space.

The reassignment result is conditional on the comparator. The absolute effect is established on both backbones when displacement magnitude is matched in pre-$\tanh$ coordinates, but not on Qwen under gain-space matching; the signed effect also does not generalize across backbones. Moreover, decoder-column-space perturbations need not correspond to controls realizable from valid contexts, and the real-displacement comparator differs in token as well as context. The residual effect therefore cannot be attributed to context alone, and which properties of the donor displacement produce it remains open.

The neuroscience connection is computational rather than mechanistic. Gain modulation in recurrent models motivates separating fixed weights from a multiplicative control signal (Stroud et al., 2018), but our control is generated from the Transformer's own state and we make no claim of biological correspondence.

Several limitations bound the broader interpretation. We study two small backbones, two corpora processed in independent 512-token blocks, and one controller family, with a learning rate carried over rather than tuned per setting. Token-only baselines lack both preceding context and position, and no position-only baseline was evaluated, so the predictive comparison does not isolate contextual content from context length; reassignment, which holds token and position fixed, is unaffected. Other context-blind alternatives, a channel-wise static baseline, and parameter-matched K variants were not evaluated. Most conditions use three seeds, and the reported intervals are conditional on

the trained controllers, unadjusted for multiplicity, and based on limited article counts. Validation text was reused, some Qwen continuation decisions used test comparisons, and several Section 7 analyses are post hoc (Section 3). An earlier implementation produced smaller GPT-2 effects; its complete training configuration is unavailable and the discrepancy remains unexplained.

10 Conclusion

Contextual control adds an explicit, bounded control signal to a frozen Transformer that can be recorded, reassigned, perturbed, and removed while the recipient text and prediction target remain fixed. Context-dependent control improves prediction beyond static and token-only controls, and interventions show that, in pre-$\tanh$ coordinates, the magnitude of the prediction change depends on which context-derived control displacement is applied, beyond its magnitude and layer-wise allocation. Neutralization further shows that assigned control contributes positively to prediction on average. Together, these results establish contextual control as an intervention-accessible framework for studying how context regulates computation in frozen Transformers, without implying an analogous mechanism in the pretrained backbone.

AI Use Statement

Generative AI tools were used, under the author's direction, for coding, consistency checks, and manuscript editing. The author reviewed all AI-assisted work and takes responsibility for the final content.

References

Peter M. Aronow, Cyrus Samii, and Valentina A. Assenova. Cluster-robust variance estimation for dyadic data. *Political Analysis*, 23(4):564–577, 2015.

A. Colin Cameron, Jonah B. Gelbach, and Douglas L. Miller. Robust inference with multiway clustering. *Journal of Business & Economic Statistics*, 29(2):238–249, 2011.

David Ha, Andrew M. Dai, and Quoc V. Le. HyperNetworks. In *International Conference on Learning Representations*, 2017. arXiv:1609.09106.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Bruce W. Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehling, Pierre Dognin, Manish Nagireddy, and Amit Dhurandhar. Programming refusal with conditional activation steering. In *International Conference on Learning Representations*, 2025. arXiv:2409.05907.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *Advances in Neural Information Processing Systems*, 2022.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

Qwen Team, An Yang, Baosong Yang, Beichen Zhang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. Compressive Transformers for long-range sequence modelling. In *International Conference on Learning Representations*, 2020.

David Raposo, Sam Ritter, Blake Richards, Timothy Lillicrap, Peter Conway Humphreys, and Adam Santoro. Mixture-of-depths: Dynamically allocating compute in Transformer-based language models. *arXiv preprint arXiv:2404.02258*, 2024.

Emilio Salinas and Peter Thier. Gain modulation: a major computational principle of the central nervous system. *Neuron*, 27(1):15–21, 2000.

Joan Serrà and Alexandros Karatzoglou. Getting deep recommenders fit: Bloom embeddings for sparse binary input/output networks. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pp. 279–287, 2017.

Niklas Stoehr, Kevin Du, Vésteinn Snæbjarnarson, Robert West, Ryan Cotterell, and Aaron Schein. Activation scaling for steering and interpreting language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024. arXiv:2410.04962.

Jake P. Stroud, Mason A. Porter, Guillaume Hennequin, and Tim P. Vogels. Motor primitives in space and time via targeted gain modulation in cortical networks. *Nature Neuroscience*, 21(12): 1774–1783, 2018.

Nicolas Vecoven, Damien Ernst, Antoine Wehenkel, and Guillaume Drion. Introducing neuromodulation in deep neural networks to learn adaptive behaviours. *PLOS ONE*, 15(1):e0227922, 2020. doi: 10.1371/journal.pone.0227922.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, 2020.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. ReFT: Representation finetuning for language models. In *Advances in Neural Information Processing Systems*, 2024. arXiv:2404.03592.

Ted Zadouri, Ahmet Üstün, Arash Ahmadian, Beyza Ermiş, Acyr Locatelli, and Sara Hooker. Pushing mixture of experts to the limit: Extremely parameter efficient MoE for instruction tuning. In *International Conference on Learning Representations*, 2024. arXiv:2309.05444.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, 2019.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *International Conference on Learning Representations*, 2024. arXiv:2309.16042.

A Parameter counts and baseline constructions

Table 3 lists the counts. The GPT-2 token-only baseline maps each token id, without access to preceding context, to a two-dimensional embedding, projects it linearly to the $K = 4$ code, and passes it through the same per-layer RMS scale and decoder as the contextual controller. The Qwen Bloom construction (Serrà & Karatzoglou, 2017) hashes each token id with two independent hash functions into 14,801 buckets each and sums the two four-dimensional rows; the pair of bucket indices is injective over the 151,936-row vocabulary (zero collisions), but the additive representation cannot realize an arbitrary table. The vocabulary-table construction assigns a rank-3 row to each of the 38,805 token types that occur in the training split and one shared row to the remaining types (0.49% of validation and 0.56% of test tokens). In Equation 3, $\epsilon = 10^{-8}$; γ_ℓ is initialized to ones, S_ℓ and $\boldsymbol{b}_\ell$ to zero, and the encoder weights and layer embeddings to $\mathcal{N}(0, 0.02^2)$ with zero biases.

Table 3: Parameter counts by component (trainable unless marked frozen; $K = 4$, hidden size 128, $G = 8$).

Component	GPT-2 small ($d = 768$, $L = 12$)	Qwen2.5-0.5B ($d = 896$, $L = 24$)
Encoder $W_a, \boldsymbol{b}_a$	98,432	114,816
Layer embeddings $\boldsymbol{e}_\ell$	1,536	3,072
Code projection $W_b, \boldsymbol{b}_b$	516	516
RMS scales γ_ℓ	48	96
Decoders $S_\ell, \boldsymbol{b}_\ell$	480	960
Contextual total	**101,012**	**119,460**
Static gains	96	192
Token-only (GPT-2 rank-2 table; Qwen Bloom / vocabulary table)	101,054	119,464 / 117,490
Token-only, full table (GPT-2 only)	201,556	–
Frozen backbone	124,439,808 (148 tensors)	494,032,768 (290 tensors)

B PER-SEED TEST PERPLEXITIES AND WITHIN-SETTING EFFECTS

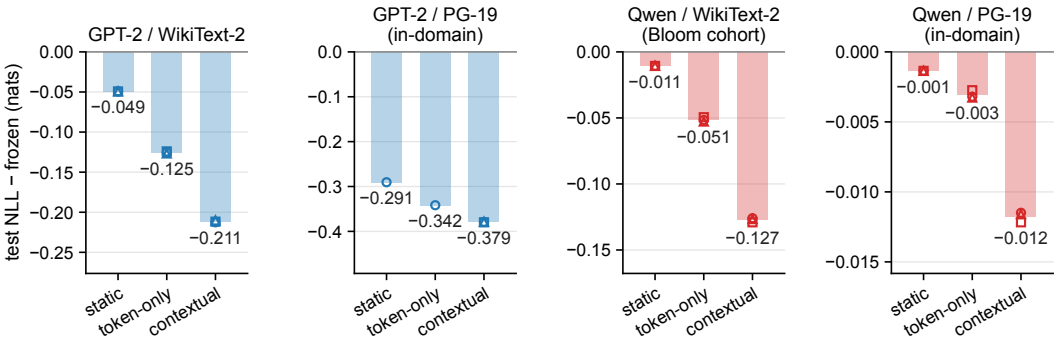

Figure 4: Within-setting predictive effect. Test negative log-likelihood (NLL) minus the frozen backbone's, in nats, for the three controller types in four settings (bars: mean over seeds; markers: individual seeds; the single-seed GPT-2 PG-19 baselines appear as one marker). Axes differ across panels because the effects differ by an order of magnitude; nothing is compared across tokenizers.

Table 4: Per-seed test perplexity (seeds 0 / 1 / 2) for the multi-seed rows of Table 1 and for the Qwen vocabulary-table cohort.

Setting	Condition	Per-seed test PPL
GPT-2 / WikiText-2	token-only, rank-2	31.2991 / 31.3421 / 31.2616
	contextual	28.6886 / 28.7119 / 28.7570
GPT-2 / PG-19 in-domain	contextual	32.3751 / 32.3508 / 32.3893
Qwen / WikiText-2, Bloom cohort	token-only, Bloom	16.3274 / 16.3624 / 16.3092
	contextual	15.1586 / 15.1107 / 15.1469
Qwen / WikiText-2, vocabulary-table cohort	token-only, vocab. table	15.9362 / 15.9516 / 15.9633
	contextual	15.1523 / 15.1088 / 15.1373
Qwen / PG-19 in-domain	token-only, Bloom	24.2826 / 24.2938 / 24.2810
	contextual	24.0820 / 24.0657 / 24.0799

Table 4 gives the per-seed values behind Table 1. Test sets contain 561 blocks for GPT-2 and 584 for Qwen, each block scoring 511 tokens. PG-19 slices are taken in the dataset's fixed order: the first 20,000 characters of each book are skipped and the next 30,000 kept; slices, like WikiText-2 rows, are joined with a two-newline separator, each split is tokenized as one stream, and the stream is truncated to the exact WikiText-2 token count of the same split and tokenizer (the 373/39/45 GPT-2 and 387/41/46 Qwen train/validation/test slice counts are before truncation; 343 Qwen training slices contribute tokens). For the Qwen vocabulary-table cohort, test perplexity is 17.1913 (frozen), 17.0100 (0.0003; static, 1.05%), 15.9504 (0.0136; token-only vocabulary table, 117,490 parameters, 7.22%) and 15.1328 (0.0221; contextual, 11.97%). The frozen model's test perplexity differs slightly between the two Qwen cohorts (17.1921 and 17.1913) because each cohort evaluated it on its own CPU (x86-64 and Apple M5, both in float32) on the identical 298,424 scored tokens; the difference, 4.7×10^{-5} nats, is the size of the float32 platform differences recorded in Appendix D,

and each cohort's reductions are relative to its own evaluation. Selected checkpoints (updates; seeds 0/1/2): GPT-2 WikiText-2 contextual 2000/1900/1700; Qwen Bloom-cohort WikiText-2 contextual 1700/1700/1800, token-only 1700/1700/1600, static 1700/1400/2000; Qwen vocabulary-table-cohort WikiText-2 contextual 1700/1800/1800, token-only 1200/1200/1800, static 1700/1400/2000; Qwen PG-19 contextual 1200/1200/1100, token-only 600/300/600, static 1700/1300/1100. On GPT-2 a full $50{,}257 \times 4$ token-only table (201,556 parameters, one seed) reaches test perplexity 31.6035, slightly worse than the rank-2 matched table (31.3009); the larger token-only table did not improve performance under the evaluated protocol.

C STATISTICAL CONSTRUCTION

Estimator. With e the vector of pair-level deviations from the mean and S the pair-by-pair matrix with $S_{ij} = 1$ when pairs i and j share an endpoint article (including $i = j$), the variance of the mean is $e^\top Se/n^2$, multiplied by the conventional one-way small-sample factor $M/(M-1)$ with M the number of distinct articles (38 in the GPT-2 cohorts, 39 in the initial and 24 in the follow-up Qwen cohort); intervals use the 0.975 quantile of $t(M-1)$.

Negative variance and leave-one-article-out diagnostic. The estimate is not guaranteed to be nonnegative and is not clamped. A negative full-sample estimate leaves the interval undefined, and the pair-bootstrap interval printed beside it is a reference that ignores article sharing; a pooled three-controller contrast whose estimate turns negative when any single article, with every pair containing it, is deleted is labeled unresolved, and per-controller cells list the articles whose deletion does so. With so few, unevenly used articles the intervals are approximate and, for signed contrasts, can be narrower than a pair bootstrap.

Robustness. A set-membership extension that counts every article inside each 192-token window (3.25% of initial-cohort windows and 3.62% of expanded-cohort windows contain a second article) changes the initial cohort's intervals by less than 0.0004. Two-way cluster-robust variances over recipient and donor articles (Cameron et al., 2011) give standard errors within 35% of the dyadic ones for the absolute contrast and the signed donor change in both GPT-2 cohorts, and no sign conclusion depends on the construction. Leave-one-article-out sweeps are computed for every dyadic quantity of the GPT-2 cohorts and the initial Qwen cohort (none is reported for the Qwen follow-up cohort); in the initial GPT-2 cohort no deletion gives a negative variance.

Chronology and interpretation. The dyadic construction and the deletion check were adopted after the initial GPT-2 cohort's results and the expanded cohort's first-controller results were known (those analyses had used pair and article-cluster bootstraps and, for the expanded cohort, a two-way variance); they were then fixed in advance for the expanded cohort's other two controllers, the initial Qwen cohort and the GPT-2 geometry follow-up (the Qwen follow-up cohort pre-registered the dyadic interval without the deletion check), and were not chosen for the significance they yield. Where an i.i.d. bootstrap and the dyadic construction disagree (the signed random change in the initial GPT-2 cohort excludes zero under the former and not under the latter), the dyadic result is reported; where the dyadic interval excludes zero but a single-article deletion makes the variance negative (the expanded cohort's pooled signed contrast against the isotropic comparator), the contrast is reported as unresolved under the expanded cohort's own pre-specified rule, although the criterion of the GPT-2 geometry follow-up, which applies no deletion check and treats that comparator as a reference rather than a primary endpoint, would classify it as donor $>$ null. Ablation intervals use the same factor with clusters defined by article ($M = 11$) or block ($M = 60$); an article cluster bootstrap gives intervals within 0.002 of the article-clustered ones.

D INTEGRITY CHECKS AND NUMERICAL CONVENTIONS

Hardware and implementation integrity. Hardware: the GPT-2 cohort was trained on Apple MPS and evaluated on CPU; the Qwen Bloom cohort on an A100 with TF32 disabled (CUDA float32 training, CPU float32 evaluation); the Qwen vocabulary-table cohort on Apple MPS with microbatch 4 and two-step gradient accumulation, evaluated on CPU. Before training, we verified on both backbones, with bitwise results where exactness is expected: neutral equivalence ($\boldsymbol{u} = 0$

reproduces the frozen logits, maximum absolute error 0.0); prefix causality (changing tokens after position t leaves controls and logits at $\leq t$ unchanged, 0.0); a frozen backbone (every pretrained tensor excluded from gradient computation, the optimizer referencing only controller tensors, and per-tensor SHA-256 hashes unchanged after an update); finite, non-zero controller gradients with no backbone gradients; gain bounds ($[0.75, 1.25]$ attained, never exceeded); sequence independence at fixed batch shape on Qwen (perturbing one sequence in a batch leaves another's logits and controls unchanged, 0.0; the GPT-2 test suite checks only that mean NLL agrees between batch sizes 2 and 1 on four blocks, 0.0, and does not test per-sequence independence directly); exact replay of the recorded original control; deterministic evaluation (repeated CPU passes bitwise; on Qwen two three-step MPS training runs from one seed gave bitwise-identical parameters); and, on Qwen, hook correctness (the modulation delta at layer 5 equals $W_2[(\boldsymbol{g}' - \boldsymbol{g}) \odot \boldsymbol{f}]$ to 0.0 while $\boldsymbol{r}$, attention and other layers' controls are unchanged). Per-tensor hashes were re-checked at the end of every training and intervention run; 0 of 148 (GPT-2) and 0 of 290 (Qwen) backbone tensors changed in every case.

Numerical and evaluation conventions. The Bloom cohort added a CPU-versus-CUDA equivalence test with pre-specified tolerances; the aggregate NLL differed by 5×10^{-5} nats, which exceeded the pre-specified 10^{-5} tolerance, so its checkpoint selection and test metrics were computed on CPU; the validation-only diagnostics and capacity-sweep evaluation of Section 7 and the Qwen follow-up cohort ran on the A100 in float32 (TF32 disabled in the diagnostics), each paired comparison within one batch composition. Controls used in interventions are captured in the same batch composition that the intervention uses, because float32 matrix products depend on batch shape at the 10^{-6} level; this is what makes replay exact rather than approximately exact. Single-position neutralization on Qwen used an efficient implementation (causal truncation to the neutralized position, unperturbed row in the same batch, LM head at scored positions only), validated against full forward passes on one block per seed (maximum $|\Delta b_t| \leq 7 \times 10^{-5}$ nats, sign agreement 100%), a narrower validation than the GPT-2 implementation received. Two evaluation-only replay runs reloaded the GPT-2 checkpoints and reproduced all ten Table 1 WikiText-2 instances bitwise and a pre-selected intervention panel (144 reassignment instances, 180 neutralization positions) exactly; they do not cover PG-19 or the expanded cohorts. Forward passes run in float32 without mixed precision, Qwen's bfloat16 release weights being upcast at load; GPT-2 single-position and layer-local neutralization recompute the intervened position in float64; losses are accumulated in float64. Checkpoint selection runs on CPU over validation blocks 0–79, keeps the earliest checkpoint on ties, and never considers the untrained state.

Test-set use. The test split was scored for Table 1 and Appendices B and G, re-read by the replay runs above and by a frozen-model sweep that fixed the evaluation convention before any controller was trained, and in two decisions: in each Qwen setting a seed-0 test comparison with the token-only controller (pre-specified for the Bloom cohort and the PG-19 arms; applied, but not recorded in advance, for the vocabulary-table cohort) decided whether seeds 1 and 2 were trained, and a three-seed test comparison decided whether the initial Qwen reassignment cohort was run. The Qwen–PG-19 protocol was pre-specified for seed 0; seeds 1 and 2 were added after the seed-0 test comparison, with the training code, data and selection rule unchanged.

Order of commitments. The training protocol, the neutralization and pair regions, and the initial GPT-2 cohort's mining rule and pair list were fixed before any intervention was run; the signed contrasts were computed after that cohort's results were known. The expanded cohort was specified after those results and run with one controller; the dyadic construction and its deletion check were adopted after the initial cohort's results and the expanded cohort's first-controller results, and were applied unchanged, under a pre-specified rule naming A primary, to the expanded cohort's other two controllers and to every later reassignment analysis. The Qwen training protocol was fixed before any Qwen controller was trained, and the initial Qwen cohort's mining rule, estimands and dyadic analysis before any Qwen result; the Qwen–PG-19 protocol was fixed for seed 0, and seeds 1 and 2 were added after the seed-0 test comparison under the same protocol. The capacity sweep and the single-pass study were pre-specified before their controllers were trained, and the diagnostics of Section 7 were post hoc. The geometry-matched comparators were introduced after the initial results on both backbones: on GPT-2 under a protocol pre-registered before any new inference, whose first step reproduced the expanded-cohort results exactly, and on Qwen in the pre-registered follow-up cohort, run on the Bloom cohort's existing controllers.

E REASSIGNMENT COHORTS: PER-SEED VALUES AND DISTRIBUTIONS

Table 5: Per-controller reassignment quantities. Total-norm isotropic comparator; nats; dyadic cluster-robust 95% intervals with the $M/(M-1)$ correction and $t(M-1)$ critical values ($M = 38$ for the GPT-2 expanded and 39 for the initial Qwen cohort). LOO (leave-one-article-out): articles whose deletion makes the variance negative.

Cohort	Quantity	Seed 0	Seed 1	Seed 2
GPT-2 expanded	Absolute A	+0.0323 [+0.0264, +0.0382]	+0.0392 [+0.0311, +0.0473]	+0.0271 [+0.0176, +0.0367]
	Donor d^{don}	+0.0330 [+0.0288, +0.0371]	+0.0375 [+0.0267, +0.0483]	+0.0376 [+0.0297, +0.0455]
	Comparator d^{null}	+0.0240 [+0.0157, +0.0323]	+0.0274 [+0.0218, +0.0331]	+0.0253 [+0.0211, +0.0296]
	Signed C	+0.0090 [+0.0055, +0.0124] LOO 22,44,56	+0.0100 [+0.0016, +0.0185]	+0.0123 [+0.0045, +0.0200]
Qwen initial	Absolute A	−0.0098 [−0.0142, −0.0053]	−0.0087 [−0.0129, −0.0045]	−0.0136 [−0.0198, −0.0074]
	Donor d^{don}	+0.0163 [+0.0070, +0.0255]	+0.0150 [+0.0061, +0.0239]	+0.0109 [+0.0059, +0.0160]
	Comparator d^{null}	+0.0178 [+0.0122, +0.0234]	+0.0208 [+0.0138, +0.0279]	+0.0157 [+0.0097, +0.0217]
	Signed C	−0.0015 [−0.0058, +0.0027] LOO 39	−0.0058 [−0.0147, +0.0030]	−0.0048 [−0.0111, +0.0016]

The initial GPT-2 cohort's per-seed values of A are +0.01697 / +0.01529 / +0.01171; in the expanded cohort the per-seed values (Table 5) are +0.0323, +0.0392 and +0.0271, per-pair values of A (Figure 6) correlate at 0.22–0.26 across seeds with sign agreement 55–56% (independence baseline about 51%), and the signed donor change correlates at 0.48–0.52. The signed donor change has median +0.0098 with 56.4% of pairs positive in the expanded GPT-2 cohort and median +0.0016 with 54.6% positive in the initial Qwen cohort (means +0.0360 and +0.0141); these statistics are not recorded for the initial GPT-2 and Qwen follow-up cohorts. In the GPT-2 expanded cohort the three-seed average signed contrast C is +0.0104 [+0.0074, +0.0134] but its leave-one-article-out variance is negative for article 39, so it is reported as unresolved. Two GPT-2 arms that were not norm-matched, a control from an arbitrary other prefix and a layer-permuted original control (the original column with its layers shuffled; not comparator 4), produce larger raw changes than the donor only because their displacements are about 54% larger; per unit displacement they fall below the donor. Family-level effects in the initial GPT-2 cohort (pair bootstrap over 150 pairs): auxiliary/copula -0.0018 $[-0.0138, +0.0104]$; content word $+0.0356$ $[+0.0188, +0.0532]$; pronoun/determiner $+0.0191$ $[+0.0043, +0.0351]$; preposition/conjunction $+0.0057$ $[-0.0064, +0.0179]$; the families are mining heuristics, not validated linguistic contrasts. The mean donor displacement norm in the 96 GPT-2 control coordinates is 6.33 / 6.27 / 6.52 per seed in the initial cohort and 7.73 in the expanded cohort. On Qwen, the donor displacement has mean norm 4.2–4.4 in the 192 control coordinates, its energy concentrates in layers 1–2 (about 10–11% each) and layer 22, and the gain-space displacement of the random control exceeds the donor's by 8% on average (median ratio 1.02).

Geometry-matched comparators on GPT-2. Before any comparator was run, under a protocol pre-registered in advance, the three controllers reproduced the expanded-cohort results exactly (replay $= 0$ on all 5,000 instances per controller; original, donor and isotropic-random losses identical to the earlier run, maximum $|\Delta| = 0.0$; headline A and d^{don} unchanged). Every decoder S_ℓ has rank 4 in every layer, so its column space is four-dimensional (in the trained parametrization, singular values 0.098–2.41 and condition numbers 1.9–22.4; these depend on how the code is parametrized, whereas the rank and column space do not; Appendix F). *Decoder-column-space null:* $\boldsymbol{\delta}_\ell = \|\mathbf{d}_\ell\| \, Q_\ell \mathbf{g}_\ell / \|\mathbf{g}_\ell\|$ with Q_ℓ the thin QR basis of S_ℓ and $\mathbf{g}_\ell \sim \mathcal{N}(0, I_4)$ per layer, instance and draw (8 draws; RNG seeds $770000 + 1000\,s + k$), so $\|\boldsymbol{\delta}_\ell\| = \|\mathbf{d}_\ell\|$ and $\boldsymbol{\delta}_\ell \in \mathrm{col}(S_\ell)$ exactly. *Real displacement (comparator 4):* another directed instance's displacement, assigned by one derangement π per controller (no fixed point, no same-pair assignment; RNG $990000 + 1000\,s$), $\boldsymbol{\delta}_\ell = \mathbf{d}_\ell^{(\pi(i))} \, \|\mathbf{d}_\ell^{(i)}\| / \|\mathbf{d}_\ell^{(\pi(i))}\|$. *Post-*$\tanh$ *gain match (pre-specified as secondary):* draw 0 of the column-space null rescaled per layer by a scalar $s_\ell \in [0, 8]$, found by bisection, so that the gain-space displacement $\|\boldsymbol{a}(\boldsymbol{u} + s_\ell \boldsymbol{\delta}) - \boldsymbol{a}(\boldsymbol{u})\|_\ell$ equals the donor's (median scale 1.03–1.04; the other cells are matched to a relative error below 2×10^{-5}). In 3.67 / 3.46 / 4.29% of (instance, layer) cells for the three controllers, in 27–35% of instances, the donor's gain-space displacement is unreachable along the null direction because $\tanh$ saturates; these cells keep the capped scale $s_\ell = 8$ and are not excluded, so the comparator's gain-space displacement falls below the donor's there (median 0.84–0.85 of the target; 9–12% of the donor's total gain-space displacement lies in capped cells, and over all cells the comparator reaches 97.5–98.1% of the donor's). The shortfall favors the donor, so the row-S contrast is, if anything, slightly overstated. Per-controller values (dyadic $t(37)$; "undefined" = negative full-sample variance estimate, pair bootstrap in brackets for reference only):

Table 6: Full comparator ladder. Three-seed pair averages (nats; directions averaged within pair, then seeds) for all four reassignment cohorts; comparator rows 1–4 and S as in Table 2 ("–": not run). Dyadic cluster-robust 95% intervals over source articles with $t(M-1)$ critical values ($M = 38$ for GPT-2, 39 for the initial and 24 for the follow-up Qwen cohort), except the Qwen follow-up cohort's signed donor change, which carries a pair-bootstrap interval. A was fixed before the interventions in the initial GPT-2 cohort and declared primary for the expanded cohort's second and third controllers; A and C were co-primary in the pre-registered GPT-2 geometry follow-up; in the Qwen follow-up cohort C was the pre-registered primary and A secondary. "unres.": a single-article deletion makes the dyadic variance negative (GPT-2), or the result under the pre-registered rule is unresolved (Qwen follow-up cohort).

	Row	GPT-2 initial (600)	GPT-2 expanded (2,500)
Signed donor d^{don}	–	+0.0143 [+0.0034, +0.0252]	+0.0360 [+0.0284, +0.0436]
Absolute A	1	+0.0147 [+0.0070, +0.0224]	+0.0329 [+0.0278, +0.0380]
	2	–	–
	3	–	+0.0308 [+0.0259, +0.0356]
	4	–	+0.0241 [+0.0197, +0.0284]
	S	–	+0.0179 [+0.0128, +0.0230]
Signed C	1	+0.0052 [−0.0057, +0.0160]	+0.0104 [+0.0074, +0.0134] unres.
	2	–	–
	3	–	+0.0095 [+0.0039, +0.0151]
	4	–	+0.0079 [+0.0019, +0.0140]
	S	–	+0.0048 [−0.0004, +0.0100]
Median A (row 1); pairs > 0		+0.00307; 52.5%	+0.0103; 58.2%
	Row	Qwen initial (2,500)	Qwen follow-up (2,500)
Signed donor d^{don}	–	+0.0141 [+0.0068, +0.0213]	+0.0092 [+0.0054, +0.0132]
Absolute A	1	−0.0107 [−0.0145, −0.0068]	−0.0088 [−0.0127, −0.0048]
	2	–	+0.0082 [+0.0060, +0.0104]
	3	–	+0.0077 [+0.0053, +0.0101]
	4	–	+0.0056 [+0.0038, +0.0075]
	S	–	+0.0023 [−0.00004, +0.0046]
Signed C	1	−0.0040 [−0.0074, −0.0007]	−0.0053 [−0.0104, −0.0001] unres.
	2	–	−0.0024 [−0.0058, +0.0010] unres.
	3	–	−0.0017 [−0.0055, +0.0020] unres.
	4	–	−0.0031 [−0.0059, −0.0002] unres.
	S	–	−0.0037 [−0.0077, +0.0003] unres.
Median A (row 1); pairs > 0		−0.0093; 38.4%	–

Comparator	Quantity	Seed 0	Seed 1	Seed 2
decoder column space (3)	A	+0.0308 [+0.0244, +0.0372]	+0.0349 [+0.0282, +0.0417]	+0.0266 [+0.0197, +0.0334]
	C	+0.0087 [undefined; boot +0.0020, +0.0157]	+0.0110 [+0.0014, +0.0205]	+0.0089 [+0.0025, +0.0153]
real displacement (4)	A	+0.0241 [+0.0207, +0.0275]	+0.0276 [+0.0212, +0.0340]	+0.0205 [+0.0132, +0.0278]
	C	+0.0091 [+0.0067, +0.0114]	+0.0057 [−0.0028, +0.0142]	+0.0091 [−0.0007, +0.0189]
post-$\tanh$ gain (S)	A	+0.0209 [+0.0136, +0.0281]	+0.0222 [+0.0128, +0.0316]	+0.0107 [+0.0021, +0.0193]
	C	+0.0069 [+0.0012, +0.0127]	+0.0079 [−0.0035, +0.0193]	−0.0003 [−0.0052, +0.0046]

Leave-one-article-out sweeps flag two cells of this table: the seed-0 column-space C (variance negative for the full sample and under 17 single-article deletions) and the seed-0 real-displacement C (negative under 8 deletions); the pooled contrasts and every A cell are unaffected. Every single draw of the column-space null gives positive A in every controller (range 0.023–0.037); single-draw C is always positive but varies several-fold across draws (0.002–0.020), and the primary A averages $|d^{\mathrm{null}}|$ over the 8 draws within each instance. Fractions of the isotropic contrast remaining: A 94% (column space), 73% (real displacement), 54% (post-$\tanh$); C 91%, 76%, 46% (ratios of point estimates on the same pairs; descriptive). Medians of A under the four comparators are +0.0103, +0.0072, +0.0082, +0.0041 with 53–58% of pairs positive; medians of C are within ± 0.0014 of zero with 48–50% of pairs positive. Figure 5 shows the per-controller intervals, and Table 6 collects every comparator for all four cohorts.

Qwen follow-up cohort. This same-token/different-context cohort on Qwen WikiText-2 (2,500 pairs, 1,324 token types, 24 articles) was mined from validation blocks 80–279, the diagnostic region of Section 7, under the expanded cohort's rule (192-token windows ending in the same token, final tokens in different articles, non-overlapping windows, final eight tokens not identical, at most three pairs per token type; mining seed 20260923) and run on the Bloom cohort's existing contextual controllers (selected at updates 1,700/1,700/1,800; Appendix B), with nothing trained for it. Its comparators, estimands and decision rule were pre-registered before execution; the pre-registered primary was the signed contrast C against each null, with A secondary. Comparators per directed instance: isotropic total-norm (one draw), decoder column space with per-layer norms (eight draws), the column-space null rescaled to the post-$\tanh$ gain displacement (four draws; corresponding matching-quality diagnostics were not available for this analysis), per-layer isotropic (four) and real displacement with per-layer norms (eight); the permutation runs over the 5,000 directed instances and excludes self-mapping but not the opposite direction of the same pair, so the real displacement comes from another directed instance, not necessarily from a different pair. Absolute values are taken per draw; draws are averaged within the directed instance, then directions and controllers within the pair. Decision intervals are dyadic over the 24 endpoint articles ($M/(M-1)$, $t(23)$); 95% pair-bootstrap intervals (10,000 resamples), which ignore article sharing, are also computed, and the signed donor change below carries that interval. Replay was exactly zero and every decoder has rank 4. Per-seed values (dyadic intervals; the signed donor change with its pair-bootstrap interval): signed donor change d^{don} +0.0045 [−0.0005, +0.0096] / +0.0114 [+0.0064, +0.0163] / +0.0117 [+0.0065, +0.0172]; A isotropic −0.0072 [−0.0136, −0.0009] / −0.0105 [−0.0150, −0.0060] / −0.0085 [−0.0137, −0.0033]; A decoder column space +0.0077 [+0.0045, +0.0110] / +0.0056 [+0.0029, +0.0083] / +0.0098 [+0.0062, +0.0134]; A real displacement +0.0050 [+0.0017, +0.0083] / +0.0041 [+0.0020, +0.0062] / +0.0078 [+0.0050, +0.0106]; C isotropic −0.0074 [−0.0173, +0.0024] / −0.0029 [−0.0073, +0.0016] / −0.0055 [−0.0129, +0.0019]; C decoder column space −0.0067 [−0.0122, −0.0013] / +0.0000 [−0.0034, +0.0034] / +0.0015 [−0.0026, +0.0056]; C real displacement −0.0079 [−0.0134, −0.0024] / −0.0005 [−0.0023, +0.0013] / −0.0008 [−0.0052, +0.0036]; C post-$\tanh$ gain match −0.0095 [−0.0158, −0.0031] / −0.0015 [−0.0065, +0.0035] / −0.0003 [−0.0043, +0.0038]; C per-layer isotropic −0.0077 [−0.0127, −0.0027] / +0.0007 [−0.0029, +0.0042] / −0.0001 [−0.0062, +0.0059] (three-seed −0.0024 [−0.0058, +0.0010]). The two remaining pre-registered absolute contrasts, available as three-seed values only, are +0.0082 [+0.0060, +0.0104] against the per-layer isotropic null, which matches the donor's per-layer norms without confinement to the decoder column space, and +0.0023 [−0.00004, +0.0046] against the post-$\tanh$ gain-matched null, whose interval includes zero. Every signed result is unresolved under the pre-registered rule, a dyadic-interval criterion with a ± 0.005-nat margin; the absolute donor advantage over the column-space and real-displacement nulls is positive in all three seeds, and the absolute contrast against the isotropic null is negative in all three. Mean donor displacement norm 4.25 / 4.41 / 4.15 in the 192 coordinates.

F LAYER-LOCAL AND CROSS-CONTROLLER OBSERVATIONS

Layer-local neutralization. On GPT-2, neutralizing one layer's control at a time at the neutralized position (WikiText-2 validation blocks 80–89: 10 blocks from 2 articles; no interval for text sampling) gives a positive effect in every layer; the late group (layers 8–11) contributes 1.6–1.7 times the larger of the early and middle groups and the final layer is the largest single contributor, although the depth profile is not monotone and mean $|\boldsymbol{a}|$ is 0.125–0.129 in every layer. The twelve single-layer effects recover 73–80% of the joint effect (+0.1315 / +0.1374 / +0.1397 nats per seed) and three coarse groups 90–93%, so single-layer effects are not shares of the total; because the loss is convex in the logits, this non-additivity does not by itself identify interactions between layers.

Cross-controller agreement. Independently trained GPT-2 controllers produce similar controls and effects on the same text: the per-position cosine between their $\boldsymbol{u}$ vectors, each centered on its controller's mean, is 0.43 to 0.46 (shuffled-position null 0.02 to 0.03), the norms of these centered vectors correlate at 0.94, and the output-space effects of neutralization have cosine 0.47 to 0.50; per-layer centered cosine is flat across depth (0.31–0.57). The latent code $\boldsymbol{z}$ is never compared across controllers because it is not identified: the RMS normalization is invariant to any orthogonal transformation of the code, which the shared code projection and the per-layer decoders can absorb, and a per-coordinate scale can move between γ_ℓ and S_ℓ; the rank and column space of each

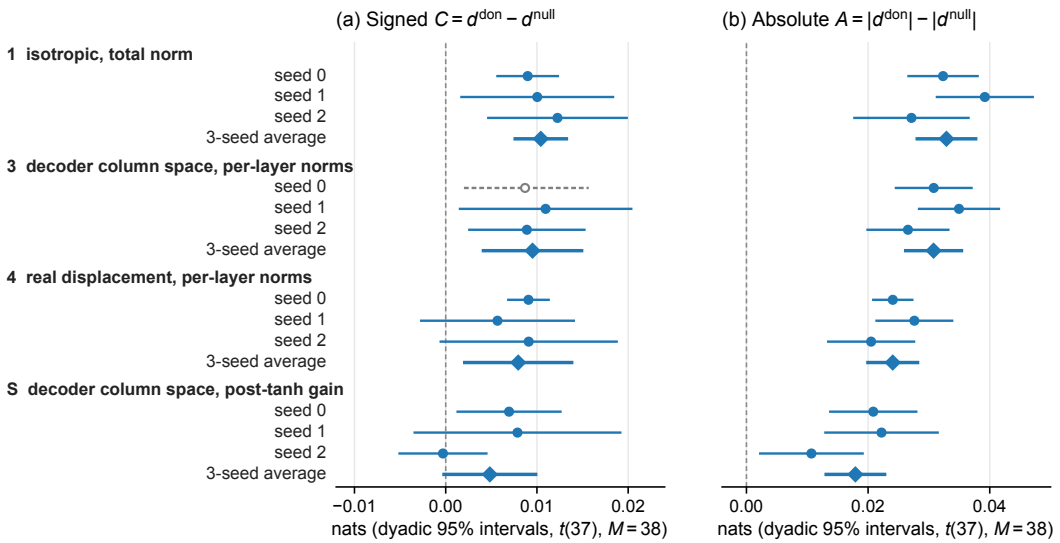

Figure 5: Geometry-matched comparators on the GPT-2 expanded cohort. Signed (C) and absolute (A) contrasts per controller and averaged over the three controllers, against comparators 1, 3, 4 and S of Table 2; dyadic 95% intervals. Where the dyadic variance estimate is negative the interval is undefined, and the pair-bootstrap interval (gray, dashed) is shown for reference only.

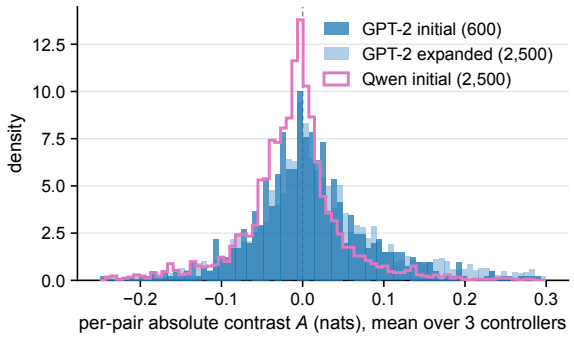

Figure 6: Pair-level distribution of the absolute contrast. Densities of per-pair A against the total-norm isotropic comparator (row 1 of Table 2), averaged over the three controllers, in three of the four reassignment cohorts; the Qwen follow-up cohort is not shown.

decoder are unaffected. On GPT-2, per-position neutralization effects across seeds have Spearman correlation 0.59–0.61 and sign agreement about 71% against a 51% independence baseline; for the vocabulary-table cohort the corresponding values are 0.43–0.46 and 65–66% against 50%. The Pearson correlations (0.92–0.93 on GPT-2, 0.93–0.94 on Qwen) are driven by the right tail and are not used as headlines.

Position-level associations. For the Bloom-cohort Qwen controllers, a post hoc analysis of the diagnostic region (validation blocks 80–279) gives correlation coefficients between b_t and the per-position advantage of the contextual over the token-only controller of 0.40 / 0.40 / 0.37 on WikiText-2 and 0.43 / 0.47 / 0.43 on PG-19, and the advantage is positive at 68–72% of positions with $b_t > 0$ against 35–43% elsewhere; both quantities contain the contextual controller's loss, so the association is partly built in. On GPT-2, across eight pre-registered predictors of b_t, target-token frequency explains 3.7% of the variance and every other predictor under 0.7%; in the lowest-entropy decile of the frozen model 23–25% of positions are entirely unaffected by control.

G GPT-2 CROSS-CORPUS TRANSFER

Evaluating the WikiText-2-trained GPT-2 controllers on PG-19 without retraining gives test perplexities of 39.7648 (static, 1 seed), 39.0016 (token-only, 3 seeds: 43.8854 / 35.9681 / 37.1512) and 37.3249 (contextual, 3 seeds: 36.6156 / 37.7581 / 37.6010) against the frozen 47.2968, reductions of 15.92%, 17.54% and 21.08%. The static row shows that input-independent modulation accounts for a substantial fraction of the transferred improvement: static control reduces perplexity by 15.92% relative to the frozen model, compared with 21.08% for contextual control; the contextual margin over static shrinks from 14.97% in-domain to 6.14% under transfer, and the token-only controller is brittle under transfer (a spread of 7.9 perplexity across seeds, driven by a few positions with very large losses; the cause is not established). Local neutralization makes the boundary precise: transferred controllers help at about half of PG-19 positions (50.87 / 49.93 / 51.74%; medians +0.0013 / −0.00004 / +0.0036), whereas in-domain controllers help at 55.46 / 55.50 / 55.62% (medians +0.0091 / +0.0096 / +0.0098); Figure 7. Article identifiers were not tracked on PG-19, so only block-clustered intervals exist there.

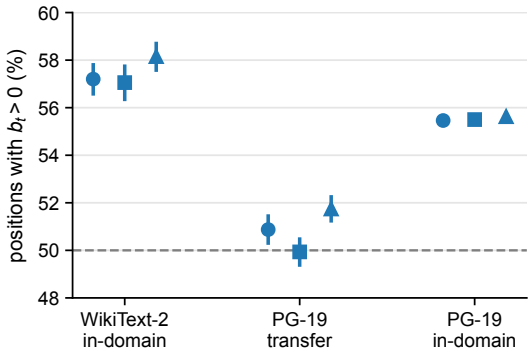

Figure 7: Where the original GPT-2 control helps. Percentage of held-out positions with $b_t > 0$ for each controller: in-domain on WikiText-2, transferred to PG-19 and in-domain on PG-19 (block-clustered 95% intervals where available; dashed line: 50%). Figure 2c and Section 6 average b_t over the three controllers first (56.9% on WikiText-2).

H QWEN–PG-19 CAPACITY SWEEP

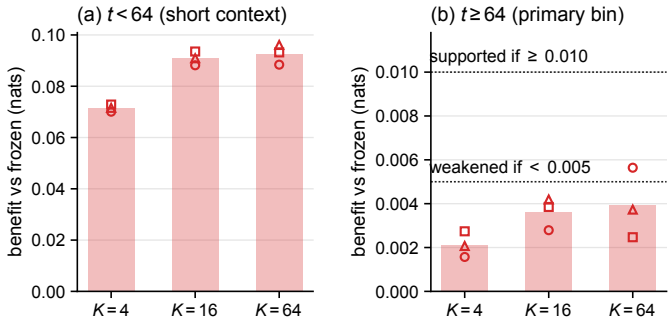

Figure 8: Capacity sweep on Qwen–PG-19. Benefit of contextual control over the frozen model on validation blocks 80–279 by code width K (three seeds each; bars: means; markers: seeds), at short context and in the pre-specified primary bin $t \geq 64$ with the pre-specified decision thresholds; under this rule the result is unresolved.

Pre-specified rule on the long-context ($t \geq 64$) benefit (Figure 8): *supported* if $K = 64$ reaches 0.010 nats in every seed with $K = 16$ in between, *weakened* if it stays below 0.005 in every seed; otherwise unresolved. Controllers: 123,600 ($K = 16$) and 140,160 ($K = 64$) parameters. Three-seed means (nats): whole-region benefit 0.0108 ± 0.0007 ($K = 4$), 0.0145 ± 0.0009 ($K = 16$), 0.0150 ± 0.0011 ($K = 64$); $t \geq 64$ benefit 0.00213, 0.00361, 0.00394 (±0.00059, 0.00073,

0.00160); train-to-validation transfer of the contextual-minus-static advantage 0.275, 0.248, 0.241; post-selection degradation of the selection curve 0.0038, 0.0048, 0.0060. Every $K = 16$ seed exceeds every $K = 4$ seed at long context, with a minimum cross-condition separation of 0.00005 nats; $K = 64$ is not consistently above $K = 16$, and one $K = 64$ seed was selected at update 600. A parameter-matched control for K was not run.

I QWEN–PG-19 SINGLE-PASS TRAINING STUDY

Protocol (pre-specified before training): the contextual ($K = 4$, 119,460 parameters) and Bloom token-only (119,464) controllers were retrained with three seeds each on a pool of 16,000 unique 512-token PG-19 training blocks (1,118 documents), each visited exactly once in 2,000 updates of eight blocks, instead of the original 4,918 blocks (343 documents) visited 3.25 times; optimizer, schedule, update count, selection rule (validation blocks 0–79) and initialization were unchanged, and the static controller was not retrained. The evaluation region V_B (validation blocks 280–511, 232 blocks, not used by any earlier analysis) was read once for the six new and six original checkpoints and the frozen model; the test split was not loaded. The pre-specified criterion classified the result as *rescued* only if, in every seed, the long-context advantage over token-only reached $+0.005$ with an interval excluding $+0.0016$, the long-context benefit over the frozen model reached $+0.008$, the whole-region loss improved, short-context benefit did not fall, and token-only did not exceed contextual at long context; a positive paired change below those bars is *partial*. Intervals are 95% block bootstraps over the 232 blocks within each seed; seeds are descriptive. Table 7 summarizes the results.

Table 7: Single-pass versus original training on the evaluation region V_B (validation blocks 280–511; nats): three-seed mean ± SD for each condition and, per seed (0 / 1 / 2), the paired change (single-pass − original) with its 95% block-bootstrap interval. λ multiplies the learned control $\boldsymbol{u}$ at every position ($\lambda = 1$: unmodified).

Quantity	original	single-pass	change per seed (0 / 1 / 2)
Advantage over token-only, $t \geq 64$	+0.00241 ± 0.00010	+0.00510 ± 0.00020	+0.00286 [+0.0018, +0.0039] +0.00257 [+0.0016, +0.0035] +0.00263 [+0.0017, +0.0036]
Benefit over frozen, $t \geq 64$	+0.00166 ± 0.00023	+0.00447 ± 0.00015	+0.00315 [+0.0023, +0.0040] +0.00288 [+0.0021, +0.0036] +0.00240 [+0.0016, +0.0032]
Benefit over frozen, $t \geq 256$	+0.00035	+0.00268	–
Benefit over frozen, $t < 64$	+0.0645 ± 0.0026	+0.0827 ± 0.0047	–
Token-only benefit, $t \geq 64$	−0.00075	−0.00062	+0.00028 [−0.0003, +0.0009] +0.00031 [−0.0002, +0.0009] −0.00023 [−0.0008, +0.0003]
Whole-region loss improvement	–	–	+0.00569 [+0.0048, +0.0066] +0.00517 [+0.0043, +0.0060] +0.00335 [+0.0025, +0.0042]
In-sample to V_B transfer of $t \geq 64$ benefit	0.12	0.54	–
$\lambda = 0.5$ rescaling at $t \geq 64$ (vs. $\lambda = 1$)	−0.00145	−0.00007	–

Held-out replication and decomposition (V_C). A second held-out region V_C (550 blocks from 28 books, none read by any model before, although some of these books also contribute other windows to V_B; a separate development region of 196 blocks from 14 other books was used for all interim decisions) was read once under three training conditions: the original condition (4,918 blocks from 343 documents, 3.25 passes), the single-pass condition (16,000 blocks from 1,118 documents, each seen once) and a third, diverse-repeated condition that keeps the original 4,918-block budget and 3.25-pass repetition but draws the blocks from 1,118 documents (three contextual and three token-only controllers; every training run passed the integrity checks of Appendix D and changed no backbone tensor). Long-context advantage over token-only: original +0.00211 ± 0.00026, diverse-repeated

+0.00283 ± 0.00065, single-pass +0.00414 ± 0.00018 (per seed, original +0.00215 / +0.00183 / +0.00235; single-pass +0.00397 / +0.00412 / +0.00433); long-context benefit over frozen: original +0.00193, diverse-repeated +0.00289, single-pass +0.00450. Paired changes per seed with block-bootstrap intervals: single-pass − original +0.0018 [+0.0011, +0.0025] / +0.0023 [+0.0017, +0.0029] / +0.0020 [+0.0014, +0.0026] (book-cluster intervals also above zero); diverse-repeated − original +0.0008 [+0.0001, +0.0014] / +0.0016 [+0.0010, +0.0022] / −0.0002 [−0.0009, +0.0005]; single-pass − diverse-repeated +0.0010 [+0.0004, +0.0017] / +0.0007 [+0.0001, +0.0013] / +0.0022 [+0.0016, +0.0028]; token-only single-pass − original at $t \geq 64$ +0.0012 [+0.0008, +0.0017] / +0.0002 [−0.0002, +0.0006] / +0.0002 [−0.0002, +0.0006]. Under the pre-specified criteria the replication passed, but *neither* document diversity (diverse-repeated − original above zero in 2 of 3 seeds) nor the unique-block schedule (single-pass − diverse-repeated above zero in 3 of 3 but below the $+0.0014$ margin in two) met its bar, so the cause remains unresolved; the long-context benefit of the best condition (+0.00450) stays below the 0.008 bar. $\lambda = 0.5$ rescaling at $t \geq 64$ improves the diverse-repeated condition in every seed (−0.0015 [−0.0017, −0.0012] / −0.0004 [−0.0006, −0.0002] / −0.0016 [−0.0019, −0.0014]) and does not change the single-pass condition (−0.0001 [−0.0003, +0.0001] / −0.0001 [−0.0003, +0.0001] / −0.0001 [−0.0003, +0.0002]). For the original controllers, halving the control ($\lambda = 0.5$) changes the loss at $t \geq 64$ by −0.0020 / −0.0012 / −0.0018 nats per seed on the diagnostic region (validation blocks 80–279) and by −0.0016 / −0.0014 / −0.0013 on V_B; after single-pass training the change on V_B is −0.00007 on average. Frozen V_B loss 3.13122 nats (perplexity 22.902); contextual whole-region loss per seed 3.12213 / 3.12181 / 3.12110 (original) and 3.11644 / 3.11664 / 3.11776 (single-pass). Under the pre-specified criterion the single-pass result is partial (the $+0.008$ benefit bar failed in all seeds and the $+0.005$ advantage bar in one). The new controllers were selected at or near the final checkpoint in every run, so no inference about longer training is drawn. At a fixed update count, the single-pass condition increased unique-document exposure 3.3-fold while reducing repetition; the observed improvement therefore accompanies altered training-data exposure, but the experiments do not isolate the contributions of document diversity and repetition. The related training-split decomposition (410 training blocks per corpus, every twelfth block, against the validation values of Section 7) gives the in-sample long-context benefit 0.0134 nats versus 0.0021 on validation for PG-19 (ratio 0.16; 0.03 beyond 256 tokens; short context 0.80) and a ratio of 0.86 on WikiText-2. The three transfer ratios quoted in this paper are different quantities and are never interchanged: the contextual controller's whole-block advantage over static gains, from training batches to the selection subset (0.28 on PG-19, 0.87 on WikiText-2; Appendix H gives it by K); the long-context benefit on training blocks versus the diagnostic region (0.16 on PG-19, 0.86 on WikiText-2; Section 7); and the long-context benefit on in-sample blocks versus V_B (0.12 before and 0.54 after single-pass training).

J Context structure: splice test and WikiText-2 header analysis

Splice test (Qwen–PG-19). On the 196-block development region an artificial boundary was created at position 256 by splicing the second half of a block from another book (cross-book) or the same book. The pre-specified prediction was that, if the short-context benefit responded to such boundaries, it would re-emerge after the splice; the re-emergence statistic R (benefit in the 64 tokens after the splice relative to the same positions unspliced) was compared with half the short-context benefit (0.034–0.045). Observed cross-book R: +0.0005 / −0.0063 / −0.0014 (original controllers), −0.0060 / +0.0020 / −0.0046 (single-pass controllers); the prediction was falsified for these artificial splices in 6 of 6 controllers. The controller hurts by 0.03–0.066 nats over the first 8 tokens after a splice and is inert beyond 32 tokens, while the frozen model's loss rises by 1.05 (cross-book) or 0.62 (same-book) nats there; token-only and static arms show no re-emergence either.

Starts of retained PG-19 book slices (exploratory). In the diagnostic region (validation blocks 80–279), 14 starts of retained PG-19 book slices fall inside blocks; a slice begins 20,000 characters into its book and follows the previous slice after a blank line (Appendix B), so each such start is an abrupt switch between books, not a book opening. At positions with $t \geq 64$ that lie within 64 tokens after such a start (890 positions), the contextual advantage over token-only is +0.0298 / +0.0406 / +0.0455 nats per seed, against +0.0009 / +0.0016 / +0.0013 at the other positions with $t \geq 64$; the elevation persists at $t \geq 256$. It is positive for 10 of the 14 starts and carried disproportionately by a few books, and the frozen model's loss is higher at these positions (3.84 against 3.09 nats). This

analysis is exploratory and was not tested prospectively; the splice result above concerns artificial splices only.

WikiText-2 header structure. On the diagnostic region (validation blocks 80–279) the contextual benefit at $t \geq 64$ is concentrated within 63 tokens after the most recent section header (positions at most 63 tokens after a header, $t \geq 64$; the far bin is at least 256 tokens after, $t \geq 256$) (41-42% of the summed benefit from 13% of positions), with a floor of $+0.074$ to $+0.086$ nats at every larger distance (far/near ratios 0.151 / 0.152 / 0.155). Confirmed once on validation blocks 280–511, which were not used to form the hypothesis: near-header benefit (+0.3872 [+0.3651, +0.4103] for seed 0; per seed +0.387 / +0.395 / +0.386) versus far benefit (+0.0528 [+0.0468, +0.0588]; +0.053 / +0.052 / +0.053), ratios 0.136 / 0.131 / 0.137 (meeting the pre-specified criterion for a header-driven pattern); contextual benefit at $t \geq 64$ on that region +0.1122 / +0.1142 / +0.1141, advantage over token-only +0.0648 / +0.0689 / +0.0652, static benefit $+0.009$. The WikiText-2 long-context claim is therefore a header-adjacent concentration plus a persistent header-independent component.

K Runtime measurements and a LoRA performance reference

Overhead. A dedicated benchmark on one A100 in eager float32 PyTorch, with no optimized kernel, measured the controller's cost. Its analytic matrix-multiply FLOPs, cross-checked exactly by a FLOP counter, are 0.55% of the forward pass for Qwen2.5-0.5B at 512 tokens and 0.91% for GPT-2 small. Measured latency overhead relative to the frozen forward pass depends strongly on batch size and sequence length: Qwen 4.4% (batch 8, 512 tokens), 6.0% (batch 8, 128), 7.0% (batch 1, 512) and 43.5% (batch 1, 128); GPT-2 5.9%, 9.1%, 33.4% and 62.2% in the same order. In the higher-throughput settings (batch 8, or 512 tokens) the overhead is about 4–7% for Qwen; in small-batch, short-sequence settings the forward pass is launch-bound and the relative overhead is substantially larger. Peak-memory increase is at most 0.47 MiB (Qwen) and 0.39 MiB (GPT-2).

LoRA reference. A rank-1 LoRA on the query, value and output projections of frozen Qwen2.5-0.5B (110,592 trainable parameters, 7% fewer than the controller) was trained with three seeds under the controller's protocol and evaluated once on WikiText-2 validation blocks 280–511, which were not used for its selection. Whole-region loss: frozen 2.974 (perplexity 19.58); LoRA 2.731 / 2.730 / 2.728 (perplexity 15.34 / 15.33 / 15.31); contextual 2.848 / 2.847 / 2.847; token-only 2.922 on average. The paired per-block difference LoRA minus contextual is -0.1179 [-0.1234, -0.1125] / -0.1165 [-0.1225, -0.1108] / -0.1186 [-0.1244, -0.1129] per seed (negative: LoRA lower; the same differences of means are +0.118 / +0.117 / +0.119 for contextual minus LoRA), and LoRA is lower on 100% of the 232 blocks in every seed; benefit at $t \geq 64$ is +0.198 / +0.198 / +0.199 for LoRA versus +0.112 / +0.114 / +0.114 for the controller. The controller closes about half as much of the loss gap as a LoRA with 7% fewer trainable parameters; its purpose here is not competitive parameter-efficient adaptation, but to provide an explicit, bounded control variable that supports direct intervention.