BORROWED FUTURES: RECIPE-DEPENDENT ATTRIBUTION IN TWO MATCHED REVISION GATES

Anonymous authors
Paper under double-blind review

ABSTRACT

Attention gates such as CASTLE and Selective Attention let later tokens change how a language model reads earlier positions, a lookahead credited with their lower loss. The credit treats the share of the gain that needs the sequence's own future as fixed by the architecture. We build a pooled-lookahead gate and an accumulated mask from the published equations, each matched to a causal baseline in small models. Retraining a gate with its later-token input taken from an unrelated sequence, at unchanged parameters and FLOPs, isolates the future-free part of its gain. In the pooled gate this part is 91% of the gain at the shortest training length and indistinguishable from zero by a third of the Chinchilla ratio. In the accumulated mask it persists at the full Chinchilla ratio, and a causal model with only a data-dependent decay gains about as much as this part. Limiting the trained pooled gate's lookahead to a short window costs more than its whole gain, while retraining with that window costs nothing measurable. In both gates, the part that survives losing the future is not specific to revision. Attributing these gates' gains therefore needs a retrained substitution at the training length in question.

1 INTRODUCTION

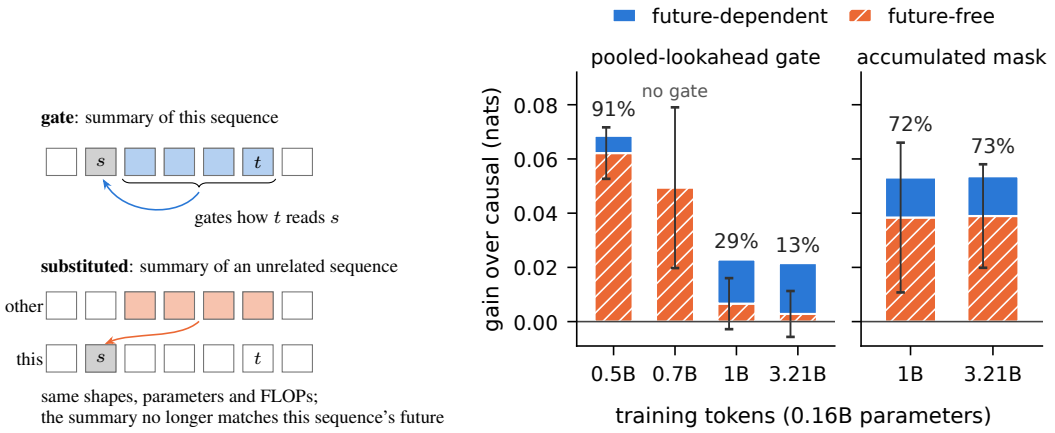
In a causal Transformer, the key that a past position s exposes to a later query t depends only on tokens up to s. Recent attention gates let the tokens between s and t change how t reads s. CASTLE (Causal Attention with Lookahead Keys; Song et al., 2026) pools those tokens into a summary that adjusts the attention logit of the pair, and in Selective Attention (Leviathan et al., 2025) each of them adds to a penalty on that logit. We call such gates *revision gates*, because later input revises how an earlier position is read. Both lower language-modeling loss below a causal baseline's, a gain naturally credited to the sequence's own later tokens reaching back.

This credit treats the split of the gain, into a part that needs the sequence's own future and a part that does not, as a property of the architecture. If it were, an ablation at an affordable training budget would give the split found at the compute-optimal Chinchilla ratio of about 20 tokens per parameter (Hoffmann et al., 2022). Testing this needs an ablation that isolates the future, and the two common ones do not. Restricting the gate in a trained model measures what that model has learned to use, which may differ from what the architecture needs. Removing the gate before training changes the parameters and the attention computation along with access to the future.

We keep the gate and cut its link to the sequence's own future. The own future of position s is the tokens of the same sequence after s, up to the query, and only positions up to the query are ever read. The substituted model trains with a borrowed future. Its gate takes the later-token input from an unrelated sequence, and the model is retrained from scratch with the same parameters, attention computation and FLOPs (Figure 1a). The loss this adds is the *future-dependent* part of the gate's gain, and the gain that remains is its *future-free* part. We measure the split at several training recipes, each a token budget with a learning-rate schedule that ends at that budget.

The split changes with the recipe. In a pooled-lookahead gate built from CASTLE's equations, the future-free part is 91% of the gain at the shortest recipe and indistinguishable from zero by 1B tokens, a third of the Chinchilla ratio (Figure 1b). In an accumulated mask built from Selective Attention's equations, a future-free part remains at the Chinchilla ratio. A causal model with only a data-dependent decay gains about as much as that part, and its decay discounts a key by the tokens between the key and the query without relating them to the key. In both gates, then, the part of the

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.



(a) the substitution (b) the gain, split by the substitution

Figure 1: **The pooled gate's future-free part shrinks to within noise of zero with training, while the accumulated mask's persists.** (a) The substituted model borrows another sequence's lookahead input and is retrained. (b) A gate's gain over the causal baseline, split into the part the substitution removes (solid) and the future-free part it leaves (hatched, 95% intervals, future-free share on top). At 0.7B tokens only the substituted and causal models were trained.

gain that survives losing the sequence's own future is not specific to revision, since it fades with training in one gate and a decay alone gains about as much in the other. We make the following contributions.

- A retrained substitution that measures what a revision gate needs from the sequence's own future (§2).
- The pooled gate's future-free part is an early-training benefit, gone by 1B tokens, while its future-dependent part does not shrink with the gain (§3, §4).
- The accumulated mask's future-free part persists at the Chinchilla ratio, and a causal model with only a data-dependent decay gains about as much (§5).
- Clamping the trained pooled gate's lookahead to 128 tokens costs 4.1× its gain, while retraining under the clamp costs nothing measurable (§6).

2 Retraining with a borrowed future

Because the substituted model is retrained from scratch, it can reorganise around the missing future, and the gain it still loses is what the architecture needs from that future. We write L_c, L_g and L_x for the held-out losses of the causal baseline, the gate and its substituted model, measured as cross-entropy in nats. The gate's gain $L_c - L_g$ splits exactly into the future-dependent part $L_x - L_g$ and the future-free part $L_c - L_x$. The split describes an architecture that may compensate for the substitution, not a trained model's use of its own future at inference, and §6 shows that the two answers can differ by more than the gate's whole gain.

Both gates add a lookahead path to a shared causal backbone, and the pooled gate also changes the head count, as noted below. The backbone is a LLaMA-style decoder (Touvron et al., 2023) with RMSNorm (Zhang & Sennrich, 2019), rotary position embeddings (Su et al., 2024) and SwiGLU feed-forward layers (Shazeer, 2020), with 0.16B parameters and a context of 1024 tokens (Appendix A). We train all models from scratch on FineWeb-Edu (Penedo et al., 2024) with the GPT-2 tokenizer (Radford et al., 2019). We build the gates from the published equations and compare them with CASTLE's published parameter counts and gains (Appendix K).

Pooled-lookahead gate. Our first gate implements the lookahead equations of CASTLE (Song et al., 2026). For a query at position t and a past key at $s \le t$, a lookahead summary $u_s^{(t)}$ pools the

positions strictly after s up to t:

$$u_s^{(t)} = \sum_{s<k\le t} \sigma\big(\tilde{q}_s^\top \tilde{k}_k/\sqrt{d}\big)\,\tilde{v}_k, \qquad \ell_{t,s} = \frac{q_t^\top k_s}{\sqrt{d}} - \phi\Big(\frac{q_t^\top u_s^{(t)}}{\sqrt{d}}\Big). \tag{1}$$

Here $\tilde{q}_s$, $\tilde{k}_k$ and $\tilde{v}_k$ are separate lookahead query, key and value projections, and σ is the logistic function. The ordinary causal query and key are q_t and k_s, $\ell_{t,s}$ is the attention logit of query t on key s, and d is the head dimension. Values stay causal, and later tokens reach the output only through the scalar $\phi(\cdot)$, which shifts the logit of each query–key pair. CASTLE uses $\phi = \mathrm{SiLU}$, which dips slightly negative and lets the future raise a logit a little. We use $\phi = \mathrm{softplus}$, which is positive and lets the future only lower a logit. This form matched or beat the SiLU form at each setting where we trained both (Appendix K). As in CASTLE, the gate matches the causal baseline's parameter count with fewer heads, 8 instead of 14, because each head carries three extra lookahead projections.

Accumulated mask. Our second gate implements Selective Attention (Leviathan et al., 2025), which subtracts an accumulated, non-negative penalty from the logits:

$$\ell_{t,s} = \frac{q_t^\top k_s}{\sqrt{d}} - F_{t,s}, \qquad F_{t,s} = \sum_{s<k<t} \mathrm{ReLU}(S_{k,s}). \tag{2}$$

Here $S_{k,s}$ is the attention logit of query k on key s in the layer's first head, and all heads of the layer share the penalty F. As in the original, the key at the first position receives no penalty. Each intervening token k can lower the attention that later queries pay to an earlier key s, which is how later tokens revise the reading of s. The mask adds no parameters, and with the penalty removed it reproduces the causal model's logits exactly.

The substitution. The substituted model takes the lookahead keys and values $\tilde{k}_k, \tilde{v}_k$ in Eq. (1), or the scores S in Eq. (2), from the previous sequence of the same micro-batch, at the same positions and in the same head. The first sequence of the micro-batch takes them from the last. Tensor shapes, parameters, FLOPs and the number of pooled pairs stay the same. The borrowed input still holds real token representations and loses only its correspondence with this sequence's own future. On the exact training sampler, adjacent sequences share no span of the corpus, and the borrowed tokens coincide with this sequence's next tokens about as often as those of independent draws do (Appendix C). We test at machine precision that each loss computation reads only positions up to its query (Appendix C).

Because each layer borrows representations that have themselves borrowed at earlier layers, a sequence's own context could return to its summary around the micro-batch, and Appendix C shows that the trained models leave this path unused.

Recipes and rounds. We train at recipes from 0.5B tokens up to the Chinchilla ratio at 3.21B tokens, and Tables 1 and 2 list the models at each. Within a recipe the models share all other settings (Appendix B). We score the models of a comparison on the same held-out windows, which cancels window-to-window variation from their differences (Appendix B).

We report 95% intervals across training seeds, written [lo, hi]. The seeds of a model were trained in several rounds, and a seed retrained in a later round can shift its loss. Where a comparison's models come from different rounds, we widen its interval by up to a shift bound of 0.015 nats, the largest shift among the seeds of a first retrain (Appendix D). Appendix D repeats each such comparison at a second shift bound, 0.036, the largest shift of a single seed in a second retrain. The claims that locate the pooled gate's fall and that its future-dependent part does not shrink rest on quantities that each compare models of one round, which need no widening.

3 The pooled gate's split across recipes

In the pooled-lookahead gate, we find the future-free part to be 91% of the gain at the shortest recipe and indistinguishable from zero at 1B tokens and at the Chinchilla ratio (Table 1). An ablation at the shortest recipe would conclude that the gain barely needs the sequence's own future, and the same ablation at either longer recipe that most of it does. The fall from the shortest recipe to the

Table 1: **The pooled gate's future-free part is 91% of its gain at 0.5B tokens and indistinguishable from zero at 1B and 3.21B.** Loss differences in nats over the parameter-matched causal baseline at 0.16B parameters. Ratio: tokens per parameter as a fraction of the Chinchilla ratio. n: seeds of the gate, substituted and causal models. Share: future-free part over gain. Intervals are 95% across seeds, widened by up to 0.015 where models come from different training rounds (Appendix D gives the 0.036 bound). The 1B share comes from models trained in one later round, and models spanning rounds give 49% (Appendix D). Lower rows: each part's change between recipes, positive when the future-dependent part rises or the future-free part falls. †Models of one training round, so no widening applies; °interval across seeds, not widened. The 0.7B recipe has no gate.

tokens	ratio	n	gain	future-dependent $L_x - L_g$	future-free $L_c - L_x$	share
0.5B	$0.16\times$	4/4/6	$+0.069$ [$+0.056$, $+0.081$]	$+0.006$ [-0.007, $+0.019$]	$+0.062$ [$+0.053$, $+0.072$]	91%
0.7B	$0.22\times$	–/3/3	–	–	$+0.049$ [$+0.020$, $+0.079$]†	–
1B	$0.31\times$	3/3/3	$+0.023$ [$+0.018$, $+0.028$]	$+0.016$ [$+0.008$, $+0.025$]	$+0.007$ [-0.003, $+0.016$]	29%
3.21B	$1.0\times$	6/6/9	$+0.022$ [$+0.016$, $+0.028$]	$+0.019$ [$+0.009$, $+0.028$]	$+0.003$ [-0.006, $+0.011$]	13%
change between recipes				rise	fall	
0.5B → 3.21B				$+0.012$ [-0.008, $+0.033$]	$+0.059$ [$+0.045$, $+0.074$]	
0.5B → 1B, one round				–	$+0.055$ [$+0.010$, $+0.100$]†	
0.7B → 1B, one round				–	$+0.043$ [$+0.016$, $+0.069$]†	
1B → 3.21B				$+0.002$ [-0.006, $+0.011$]°	$+0.004$ [-0.004, $+0.012$]°	

Chinchilla ratio compares models of different rounds, and its interval excludes zero at both shift bounds (Appendix D). Between 1B tokens and the Chinchilla ratio, neither part changes detectably.

Within one training round, the future-free part falls between 0.7B and 1B tokens (Table 1, rows marked †). The part at the lower budget exceeds the part at the higher by an interval that excludes zero, and it differs from the shortest recipe's part by an interval that includes zero (Appendix D). We therefore locate the fall in this range.

The future-dependent part does not shrink with the gain. Each quantity in this check compares models of one training round. A future-dependent part that shrank in proportion to the gain from the shortest recipe to 1B tokens would be at most $+0.006$ there, and the measured interval starts at $+0.008$ (Table 1).

4 An early-training benefit

Measured in training tokens, the pooled gate's future-free part is a speedup early in training that the substituted model loses and the full gate keeps. We express each model's loss as a multiple of its training tokens: the causal baseline's tokens needed to reach that loss, divided by the tokens the model trained on (Appendix D). The substituted model's multiple falls from $1.10\times$ to $1.03\times$ across the range where its future-free part falls, while the full gate's multiple stays at $1.09\times$ at the longer recipes.

The benefit follows the token count, as models stopped partway through the Chinchilla-ratio schedule show. Stopped at 1B tokens with the learning rate still high, these models keep a future-free share whose interval contains the 49% of fully annealed models of their round and excludes the 13% at the Chinchilla ratio (Appendix E).

At 1B tokens, what remains of the benefit needs more than the gating computation. There the same machinery pointed at the sequence's past (the backward model of Appendix F) has a loss 0.026 higher than causal, and fed a learned constant or magnitude-matched noise it does no better than causal (Appendix F).

At the Chinchilla ratio, the gate's gain minus that of a deeper causal model with the same FLOPs per token is -0.006 [-0.024, $+0.012$], which bears on whether the gate is worth its compute (Appendix H).

Table 2: **The two gates differ in what the substitution leaves, not detectably in what it removes.** Loss differences in nats at 0.16B parameters. Each recipe's models are scored together against the full causal baseline ($n = 9$ seeds). Table 1's 1B row instead uses models trained in one round. n: seeds of the gate and substituted models. Share: future-free part over gain. Intervals as in Table 1; °not widened. Difference rows: the pooled gate's future-dependent part minus the mask's, and the mask's future-free part minus the pooled gate's. The WikiText-103 row rescores the same checkpoints (Appendix J; interval across seeds, not widened).

gate, tokens	n	gain	future-dependent $L_x - L_g$	future-free $L_c - L_x$	share
pooled gate, 3.21B	6/6	+0.022	+0.019 [+0.009, +0.028]	+0.003 [−0.006, +0.011]	13%
accumulated mask, 1B	3/4	+0.053	+0.015 [+0.002, +0.028]°	+0.038 [+0.011, +0.066]	72%
accumulated mask, 3.21B	6/6	+0.054	+0.015 [+0.011, +0.018]	+0.039 [+0.020, +0.058]	73%
difference, 3.21B			+0.004 [−0.007, +0.015]	+0.036 [+0.018, +0.054]	
on WikiText-103				+0.054 [+0.042, +0.066]	

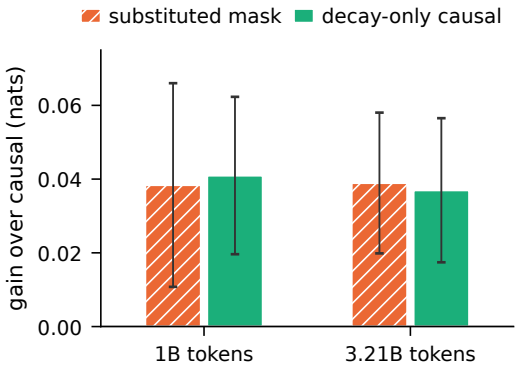


Figure 2: **A decay-only causal model gains about as much as the accumulated mask's future-free part.** Gain over the shared causal baseline of the substituted mask and of a causal model with only a data-dependent decay. At 1B tokens the substituted mask's gain minus the decay-only model's is −0.003 [−0.031, +0.026]. Error bars: 95% intervals (§2).

5 A MATCHED ACCUMULATED MASK

The accumulated mask keeps a future-free part at the Chinchilla ratio, where the pooled gate's is indistinguishable from zero. There the mask's future-free part is +0.039 against the pooled gate's +0.003, and their difference excludes zero (Table 2). Rescored on WikiText-103 (Merity et al., 2017), a corpus no model trained on, the mask's future-free part again exceeds the pooled gate's, although this comparison depends on where end-of-text tokens sit in the evaluation text (Appendix J).

The two gates' future-dependent parts at the Chinchilla ratio differ by an interval that includes zero (Table 2). The gates thus differ in what the substitution leaves, and not detectably in what it removes.

What survives the substitution in the mask is consistent with the distance term that the substitution preserves. The penalty on key s in Eq. (2) has two ingredients. The pairwise term is how strongly each intervening token scores s. The distance term arises because the penalty sums one score per intervening token, and so it grows with the distance $t - s$ even when the scores are scrambled. Borrowed scores break the pairwise term but keep the count of terms, and the substituted penalty still grows with distance, the profile the Forgetting Transformer (Lin et al., 2025) is built around.

To test this, we train a decay-only model: a causal model with only a data-dependent decay, which gains about as much as the substituted mask at both recipes (Figure 2). Its decay is the Forgetting Transformer's, which adds $\sum_{s<k\le t} \log \sigma(w^\top x_k + b)$ to each logit $\ell_{t,s}$, with x_k the input at position k and a learned w, b per head. It reads the tokens between a key and the query but has no pairwise term. At 1B tokens the difference between the two gains has an interval that includes zero, and at the Chinchilla ratio the gains are +0.037 and +0.039. The mask's future-free parts, the difference between the gates and the decay-only gains compare models of different rounds, and at the larger shift bound their intervals reach zero (Appendix D).

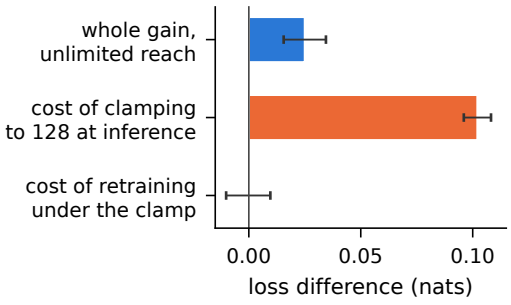


Figure 3: **Ablating the trained gate calls a long window essential, while retraining under the same clamp finds no measurable need for it.** The pooled gate at context length 2048, trained to 1B tokens: its gain over causal with an unlimited window, the loss increase from clamping its window to 128 tokens at inference, and the loss increase of the same architecture retrained under that clamp. The retrained interval excludes a cost above 38% of the gain. Error bars: 95% intervals across training seeds.

6 Ablating the lookahead window after training

We find that an inference-time ablation of the pooled gate's lookahead window and a retrained one give opposite answers about what the gate needs. CASTLE's best published variant limits the lookahead summary to the 128 tokens after each key (Song et al., 2026), and we ask whether our gate needs a longer window. We ask at a context of 2048 tokens, where the gate trained with an unlimited window gains $+0.025$ over causal (Figure 3).

At inference, a clamp of the trained gate's window to 128 tokens, with all weights unchanged, raises its loss by 0.102, $4.1\times$ that gain. The same clamp changes the causal baseline's loss only at machine precision, and the cost comes from the gate alone.

When we retrain the gate under the same clamp, the cost is 0.000 $[-0.010, +0.010]$. Ablating a trained model and retraining under a restriction are known to measure different quantities (Hooker et al., 2019; Leavitt & Morcos, 2020; Li & Janson, 2024; Cai et al., 2026). Here, at this context length and recipe, the two answers differ by more than the gate's whole gain, and only the retrained answer bears on what the architecture needs.

7 Related work

Revision and lookahead architectures. CASTLE (Song et al., 2026) lets later tokens update the keys earlier positions expose, and reports gains that grow with model size. Lookahead attention (Du et al., 2023) attends to sampled continuations and observes a model "benefiting from the extra computation without actually using the lookahead information". Its lookahead reads sampled continuations and has no counterfactual model. We supply one with our substitution, and our recipes show that its answer changes with training length. Closed-loop Transformers (Anbar Jafari & Anbarjafari, 2025) iterate within a position, and latent lookahead training (Noci et al., 2026) rolls hidden states forward before committing to a token, supervised by the next ground-truth tokens. Both act within a position or on predicted states, outside the reach of our substitution. Prefix-LM and bidirectional encoders give earlier positions access to later ones by masking (Raffel et al., 2020; Wang et al., 2022; Katz et al., 2025; BehnamGhader et al., 2024), and Madureira et al. (2024) analyse how Transformers that re-read the whole prefix at each new token revise their readings of local ambiguities.

Accumulated and decaying attention biases. Selective Attention (Leviathan et al., 2025) accumulates pairwise selection scores, the form of our second gate. The Forgetting Transformer (Lin et al., 2025) accumulates a data-dependent decay that is not pairwise, as do retention decays (Sun et al., 2023) and stick-breaking attention (Tan et al., 2025), and ALiBi (Press et al., 2022) fixes the decay in advance. GAPE (Ali et al., 2026) gates a distance-based suppression of the logits by query and key content. StableMask (Yin et al., 2024) and gated attention (Qiu et al., 2025) improve causal attention with extra structure that reads only the past. These papers report each design's whole gain, and with our second gate we find that, in an accumulated mask, the substitution preserves a distance decay of the kind these designs are built around.

Ablation protocols. Removing features and retraining gives different answers from removing them in a trained model (Hooker et al., 2019; Leavitt & Morcos, 2020). Optimal ablation replaces a component with the constant that least harms the loss (Li & Janson, 2024), and Cai et al. (2026) revisit retraining-based evaluation of feature attributions. Borobia et al. (2026) ablate the components of hybrid language-model architectures to measure what each contributes. Activation patching (Meng et al., 2022; Zhang & Nanda, 2024) and Future Lens (Pal et al., 2023), which transplants hidden states into unrelated contexts, intervene at inference on a fixed model, whereas we retrain under the substitution. Seed-to-seed variation in pretraining is large enough to call for seed-level intervals (Sellam et al., 2022), which our tables report.

Regime-dependent comparisons and diagnostic studies. Architecture rankings can change with scale and training budget (Kaplan et al., 2020). Tay et al. (2022) find that the best architecture at small scale is often not the best at large scale, and Ni et al. (2025) find that diffusion and autoregressive language models swap rank as data are repeated. Short-context pretraining can also match long-context pretraining (Zhu et al., 2025). In these studies the ranking of architectures moves with the regime. We fix the architecture, and the split of its gain moves. Diagnostic studies test what an architecture's score is made of (Arora et al., 2024; Jelassi et al., 2024; Merrill et al., 2024). Bulut (2026) add missing controls to claims about reasoning traces and find that most of the claimed effect disappears. That study varies a fixed model's generation budget at inference and measures the effect at several budgets, as our recipes do for the training budget.

8 Limitations

The findings hold for the two gates we built from published equations and trained from scratch on one corpus, at 0.16B parameters with a check at 0.35B, and neither published system is reproduced. CASTLE's largest reported effects lie at larger models trained beyond the Chinchilla ratio, beyond our recipes (Appendix K). At the larger size the future-free share at a similar number of tokens per parameter is smaller, and the size of the early benefit depends on model size (Appendix I). A recipe changes the token budget and the learning-rate schedule together. Our controls rule out the schedule's decay state, and they make less likely, without excluding, an account in which the shared peak learning rate suits one budget better than the other (Appendix E). Comparisons across training rounds are widened under the assumption that a round shifts the compared models alike (Appendix D).

9 Conclusion

In the two revision gates we built, the part of the gain that survives losing the sequence's own future is not specific to revision. In the pooled-lookahead gate it is an early-training benefit that longer recipes remove, and in the accumulated mask a causal model with only a data-dependent decay gains about as much. The part specific to revision is the future-dependent one, and in the pooled gate it does not shrink with the gain.

These findings bear on three design decisions for such gates. The first is whether to keep a long lookahead window. An ablation of the trained gate calls the window essential, while retraining under the clamp finds a cost indistinguishable from zero, which at this scale favours the windowed variant that CASTLE reports as its best. The second is what a pooled gate buys at the Chinchilla ratio. Its future-free part is indistinguishable from zero there, and its gain over a causal model with the same FLOPs in depth has an interval that includes zero. The third is the baseline an accumulated mask should beat. We find that a decay alone is that baseline, since the decay-only model gains about as much as the part of the mask's gain that survives the substitution. What the pairwise term adds beyond the decay is the smaller, future-dependent part. An ablation after training would have reversed the answer on the window, and a split measured at the shortest recipe alone would have reversed the answer on the pooled gate's gain. Attributing these gates' gains therefore takes a retrained substitution at the recipe the attribution is meant for.

AI USE STATEMENT

AI assistants based on large language models were used for part of the execution of this work, under the authors' direction: implementing the models and the training and evaluation code, and tokenizing and packing the datasets. All research decisions were made by the authors, who verified every AI-assisted output and every reported result. The authors take full responsibility for the content of this paper.

REPRODUCIBILITY STATEMENT

An anonymized code supplement contains the models, the training and evaluation code, the analyses and one entry script per table and figure, with a README that maps each to the command that regenerates it. Appendix A lists the architecture, optimiser, schedule and data settings, Appendix B the evaluation and interval protocol, and Appendix C the correctness tests and the random-initialisation check on each loss computation. Per-seed losses will be released with the paper.

REFERENCES

Riccardo Ali, Alessio Borgi, Christopher Irwin, Mario Severino, and Pietro Liò. Remember to forget: Gated adaptive positional encoding. *arXiv preprint arXiv:2605.10414*, 2026.

Akbar Anbar Jafari and Gholamreza Anbarjafari. Closed-loop Transformers: Autoregressive modeling as iterative latent equilibrium. *arXiv preprint arXiv:2511.21882*, 2025.

Simran Arora, Sabri Eyuboglu, Aman Timalsina, Isys Johnson, Michael Poli, James Zou, Atri Rudra, and Christopher Ré. Zoology: Measuring and improving recall in efficient language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. LLM2Vec: Large language models are secretly powerful text encoders. In *Conference on Language Modeling (COLM)*, 2024.

Hector Borobia, Elies Seguí-Mas, and Guillermina Tormo-Carbó. Component ablation for efficient hybrid language model architectures: Performance, resilience, and compression implications. *arXiv preprint arXiv:2603.22473*, 2026.

Yigit Utku Bulut. It's the problem, not the path: Budget and difficulty confounds in LLM reasoning trajectories. *arXiv preprint arXiv:2609.03436*, 2026.

Yi Cai, Thibaud Ardoin, Mayank Gulati, and Gerhard Wunder. Rethinking explanation evaluation under the retraining scheme. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

Li Du, Hongyuan Mei, and Jason Eisner. Autoregressive modeling with lookahead attention. *arXiv preprint arXiv:2305.12272*, 2023.

Bradley Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7:1–26, 1979.

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, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Samy Jelassi, David Brandfonbrener, Sham M. Kakade, and Eran Malach. Repeat after me: Transformers are better than state space models at copying. In *International Conference on Machine Learning (ICML)*, 2024.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Shahar Katz, Liran Ringel, Yaniv Romano, and Lior Wolf. Segment-based attention masking for GPTs. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

Matthew L. Leavitt and Ari Morcos. Towards falsifiable interpretability research. *arXiv preprint arXiv:2010.12016*, 2020.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Selective attention improves Transformer. In *International Conference on Learning Representations (ICLR)*, 2025.

Maximilian Li and Lucas Janson. Optimal ablation for interpretability. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Zhixuan Lin, Evgenii Nikishin, Xu Owen He, and Aaron Courville. Forgetting Transformer: Softmax attention with a forget gate. In *International Conference on Learning Representations (ICLR)*, 2025.

Brielen Madureira, Patrick Kahardipraja, and David Schlangen. When only time will tell: Interpreting how Transformers process local ambiguities through the lens of restart-incrementality. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

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

William Merrill, Jackson Petty, and Ashish Sabharwal. The illusion of state in state-space models. In *International Conference on Machine Learning (ICML)*, 2024.

Jinjie Ni, Qian Liu, Longxu Dou, Chao Du, Zili Wang, Hang Yan, Tianyu Pang, and Michael Qizhe Shieh. Diffusion language models are super data learners. *arXiv preprint arXiv:2511.03276*, 2025.

Lorenzo Noci, Gregor Bachmann, Seyed-Mohsen Moosavi-Dezfooli, and Moin Nabi. Thinking into the future: Latent lookahead training for transformers. *arXiv preprint arXiv:2603.20219*, 2026.

Koyena Pal, Jiuding Sun, Andrew C. Yuan, Byron Wallace, and David Bau. Future lens: Anticipating subsequent tokens from a single hidden state. In *Conference on Computational Natural Language Learning (CoNLL)*, 2023.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro von Werra, and Thomas Wolf. The FineWeb datasets: Decanting the web for the finest text data at scale. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations (ICLR)*, 2022.

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. Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text Transformer. *Journal of Machine Learning Research*, 21:1–67, 2020.

Thibault Sellam, Steve Yadlowsky, Jason Wei, Naomi Saphra, Alexander D'Amour, Tal Linzen, Jasmijn Bastings, Iulia Turc, Jacob Eisenstein, Dipanjan Das, Ian Tenney, and Ellie Pavlick. The MultiBERTs: BERT reproductions for robustness analysis. In *International Conference on Learning Representations (ICLR)*, 2022.

Noam Shazeer. GLU variants improve Transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Zhuoqing Song, Peng Sun, Huizhuo Yuan, and Quanquan Gu. Causal attention with lookahead keys. In *International Conference on Machine Learning (ICML)*, 2026.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced Transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to Transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023.

Shawn Tan, Songlin Yang, Aaron Courville, Rameswar Panda, and Yikang Shen. Scaling stick-breaking attention: An efficient implementation and in-depth study. In *International Conference on Learning Representations (ICLR)*, 2025.

Yi Tay, Mostafa Dehghani, Samira Abnar, Hyung Won Chung, William Fedus, Jinfeng Rao, Sharan Narang, Vinh Q. Tran, Dani Yogatama, and Donald Metzler. Scaling laws vs model architectures: How does inductive bias influence scaling? In *Findings of the Association for Computational Linguistics (EMNLP Findings)*, 2022.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Thomas Wang, Adam Roberts, Daniel Hesslow, Teven Le Scao, Hyung Won Chung, Iz Beltagy, Julien Launay, and Colin Raffel. What language model architecture and pretraining objective work best for zero-shot generalization? In *International Conference on Machine Learning (ICML)*, 2022.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392, 2020.

Bernard L. Welch. The generalization of 'Student's' problem when several different population variances are involved. *Biometrika*, 34:28–35, 1947.

Qingyu Yin, Xuzheng He, Xiang Zhuang, Yu Zhao, Jianhua Yao, Xiaoyu Shen, and Qiang Zhang. StableMask: Refining causal masking in decoder-only Transformer. In *International Conference on Machine Learning (ICML)*, 2024.

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

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

Tongyao Zhu, Qian Liu, Haonan Wang, Shiqi Chen, Xiangming Gu, Tianyu Pang, and Min-Yen Kan. SkyLadder: Better and faster pretraining via context window scheduling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

A CONFIGURATION

All models share one training script and one backbone, and a single setting selects the attention variant. Table 3 lists the settings. The causal baseline, the accumulated mask and the decay-only model use the full head count; the pooled-lookahead models use $4/7$ of it, because their attention block holds seven projections against four, which is CASTLE's own parameter-matching rule. At 0.16B parameters this gives exactly $160{,}689{,}536$ parameters in the causal, gate, substituted and accumulated-mask models. The decay-only model adds one forget-gate projection per head, 0.09% more parameters. The FLOP-matched depth baselines have 14 layers at 0.16B and 27 at 0.35B, chosen analytically from the per-token FLOPs of the gate.

Table 3: Training and evaluation settings. A step is a micro-batch of 8 sequences of 1024 tokens (4 of 2048 at context length 2048) with 64 accumulation steps.

	0.16B	0.35B
layers / width / head dim	12 / 896 / 64	24 / 1024 / 64
heads (causal / pooled gate)	14 / 8	16 / 9
normalisation, position, MLP	RMSNorm, rotary, SwiGLU	
tokenizer	GPT-2 BPE, 50,304 types	
training data	FineWeb-Edu, 10B-token sample	
tokens per step	524,288	
training tokens	0.5B, 1B, 3.21B	1B
peak learning rate	6×10^{-4}	3×10^{-4}
warmup	2.1% of steps	
schedule	cosine to 10% of peak	
optimiser	AdamW, $\beta = (0.9, 0.95)$	
weight decay, clipping	0.1, 1.0	
precision	bf16 autocast	
evaluation windows	$800 \times 8 \times 1024$ tokens	

B EVALUATION AND INTERVALS

At 0.16B parameters every model is trained to 0.5B, 1B or 3.21B tokens, $0.16\times$, $0.31\times$ and $1.0\times$ the Chinchilla ratio; the causal and substituted models are also trained to 0.7B tokens, $0.22\times$. Each model anneals its cosine schedule to its own token budget, so a recipe changes the budget and the schedule together. Within a recipe, models share token budget, steps, batch size, peak learning rate, warmup and schedule. Every number comes from one evaluation run that scores all models of a comparison on the same draw of held-out windows. Because the draw is shared, it cancels from every difference, and the quoted intervals are 95% intervals across training seeds: Welch intervals for differences (Welch, 1947) and seed-level bootstrap intervals for shares (Efron, 1979). Tables give seed counts per model. Evaluation runs that share models agree on them within 10^{-4} nats, and a pass repeated on the same hardware reproduces its scores exactly.

C CORRECTNESS TESTS AND THE DONOR CHECK

Each substitution path is tested in float64 before any model trains on it. For the pooled gate, the parallel and recurrent forms agree to machine precision, and so do the chunked and dense kernels. Outputs at position t do not change when any later token changes, and the substituted summary does not change when this sequence's own future changes. For the accumulated mask, the fast path, the dense logits and a step-by-step transcription of the published equation agree to 2×10^{-16}, and multiplying the penalty by zero reproduces the causal logits exactly. Every loss computation is also run on a randomly initialised model of the exact architecture it scores, which must score within 5% of chance, $\ln 50{,}304$ nats, before it is used.

The donor check runs on the exact training sampler. Among $16{,}000$ rows, no two adjacent rows share any span of the corpus, and donor tokens equal the target's next tokens 0.765% of the time, against 0.760% for independently drawn sequences. The substitution rolls each layer's lookahead input by one row of the micro-batch, which holds eight sequences; every substituted model is trained

at context length 1024. A chain of such hops, one per layer, can therefore carry the target row's own context back into its substituted summary from the eighth layer on, five of the twelve layers at 0.16B parameters. Along the chain the context passes through seven other sequences and enters each only through the scalar per query–key pair by which the gate adjusts an attention logit. The tests above exclude any path from positions after the query, but not this one. Its direction is fixed: any benefit the model draws from it lowers the substituted model's loss and so inflates the future-free part. To measure what the trained models draw from it, we rescored 38 substituted models from nine substituted settings of both gates, all at 1024 tokens of context, on the same held-out windows. The donor cycle ran over 16 sequences, so that a chain needs 16 hops to return and cannot do so within the 12 layers of these models. The path is reachable: in float32, replacing the row eight hops back changes a trained model's logits by 1.1–1.7×10^{-5}. It is unused: per setting, the mean change in loss is at most 1.2×10^{-5} nats in magnitude and every 95% interval lies within $\pm 4.0 \times 10^{-5}$. Four models with no donor path (a causal model, a gate, and two accumulated masks), regrouped the same way, move by up to 1.5×10^{-5}, the numerical floor of regrouping. No main-text contrast moves by more than 2×10^{-5} when recomputed from the regrouped scores.

D TRAINING ROUNDS AND INTERVAL WIDENING

Seeds of one model were trained in separate rounds, and the model code changed between rounds while every setting stayed identical. To size the effect, we retrained seeds 0–2 of the causal, gate and substituted models at 1B tokens in one round, with one code version, and scored them beside their originals. The nine same-seed shifts (retrained minus original) have a largest magnitude of 0.015 nats, with mean shifts of -0.008 for causal, $+0.002$ for the gate and $+0.004$ for the substituted model. The difference between the gate's and the substituted model's mean shifts, -0.002 [-0.012, $+0.007$], shows no shift specific to the substitution. We treat 0.015 nats as a shift bound per unit of round imbalance. For each contrast we split every model's ± 1 weight in the contrast over the rounds of its seeds, sum the signed weight on each round, and call half the summed absolute value the contrast's imbalance m. Both ends of its 95% interval are widened by $0.015\,m$. A two-model contrast whose models share no round has $m = 1$; one whose terms cancel within every round has $m = 0$. The widening assumes that a round shifts every model alike, so that shifts cancel within a round; the mean shifts above differ by architecture, so contrasts that mix architectures within a round may carry more bias than the widening allows. The 1B values of Table 1 come from these nine retrained models ($n = 3$ each). Their future-dependent part, $+0.016$ [$+0.008$, $+0.025$], matches that of the 1B models of Table 5, which span rounds, while their future-free part is $+0.007$ [-0.003, $+0.016$] against $+0.014$ there. This is why the 1B future-free share reads 29% in Table 1 and 49% in Table 5. The two future-free estimates differ by -0.007 [-0.016, $+0.001$], less than the largest same-seed shift, and the round-spanning one is itself indistinguishable from zero after widening, $+0.014$ [-0.004, $+0.032$]. The fall from the shortest recipe holds with either: measured to the round-spanning models it is $+0.048$ [$+0.026$, $+0.070$] after widening.

A second retrain sized the shifts at 0.5B tokens. Seeds 0–2 of the causal and substituted models, retrained in the round of the 1B models above and of the 0.7B models, shifted from their originals by -0.004, $+0.001$ and $+0.036$ (causal) and by -0.002, $+0.016$ and $+0.004$ (substituted). One causal seed thus shifted by more than twice the 1B bound. What biases a comparison of rounds is the shift of the future-free part itself, the causal model's mean shift minus the substituted model's: -0.013 at 1B tokens and $+0.005$ at 0.5B, opposite in sign and both inside 0.015. Within that round the 0.5B future-free part is $+0.062$ [$+0.015$, $+0.109$], the same estimate as from the original models, $+0.062$ [$+0.053$, $+0.072$], with a wider interval from three seeds, one of them the causal seed that shifted by 0.036. In the same round the 0.5B future-free part minus the 0.7B one is $+0.012$ [-0.029, $+0.054$], and the 0.7B part is $+0.049$ [$+0.020$, $+0.079$]. Table 4 widens every main-text contrast whose models span rounds by both bounds. At 0.036 the fall of the pooled gate's future-free part from 0.5B to 3.21B tokens still excludes zero. The pooled gate's 3.21B future-dependent part, the mask's future-free parts and the decay-only gains keep their sign, but their intervals reach zero.

The token multiples of §4 convert a loss into causal training tokens. The causal baseline's annealed loss at 0.5B, 0.7B, 1B and 3.21B tokens is interpolated linearly in the logarithm of the token count. A model's multiple is the number of causal tokens that reach its loss, divided by the tokens it trained on, with intervals from resampling seeds. The 0.5B, 0.7B and 1B causal points come from the one round, and the 3.21B point pools rounds; above 3.21B tokens the last segment is extended, so values

Table 4: **Every main-text contrast whose models span training rounds, widened by both measured bounds.** Loss differences in nats at 0.16B parameters; 95% intervals across seeds, widened by the contrast's round imbalance times 0.015 (the largest of nine same-seed shifts at 1B tokens) or 0.036 (the largest of six at 0.5B tokens). Contrasts measured within one round (Table 1, †) need neither.

contrast	estimate	widened by 0.015	widened by 0.036
pooled gate			
gain, 3.21B	+0.022	[+0.016, +0.028]	[+0.012, +0.031]
future-dependent part, 3.21B	+0.019	[+0.009, +0.028]	[−0.001, +0.039]
future-free part, 3.21B	+0.003	[−0.006, +0.011]	[−0.013, +0.018]
rise of the future-dependent part, 0.5B → 3.21B	+0.012	[−0.008, +0.033]	[−0.019, +0.043]
fall of the future-free part, 0.5B → 3.21B	+0.059	[+0.045, +0.074]	[+0.038, +0.081]
gain minus the 14-layer causal model's, 3.21B	−0.006	[−0.024, +0.012]	[−0.045, +0.033]
loss of the past-pointing gate minus causal, 1B	+0.026	[+0.005, +0.047]	[−0.016, +0.068]
accumulated mask and decay-only model			
mask gain, 3.21B	+0.054	[+0.035, +0.072]	[+0.014, +0.093]
mask future-free part, 1B	+0.038	[+0.011, +0.066]	[−0.010, +0.087]
mask future-free part, 3.21B	+0.039	[+0.020, +0.058]	[−0.001, +0.079]
mask future-free part minus the pooled gate's, 3.21B	+0.036	[+0.018, +0.054]	[−0.003, +0.075]
pooled future-dependent part minus the mask's, 3.21B	+0.004	[−0.007, +0.015]	[−0.018, +0.026]
decay-only gain, 1B	+0.041	[+0.020, +0.062]	[−0.001, +0.083]
decay-only gain, 3.21B	+0.037	[+0.017, +0.057]	[−0.004, +0.078]
substituted mask's gain minus the decay-only model's, 1B	−0.003	[−0.031, +0.026]	[−0.052, +0.047]

for the 3.21B models are lower bounds. The substituted pooled gate's multiple is 1.11× at 0.5B tokens and 1.10× [1.09, 1.12] at 0.7B. It is 1.03× [1.01, 1.04] at 1B and 1.01× at 3.21B. The full gate's multiple is 1.09× [1.08, 1.11] at 1B and 1.09× at 3.21B.

E SCHEDULE CONTROLS

The shortest and longest recipes differ in more than tokens, so we trained models that test whether the schedule's decay state, the placement of the shared learning rate or noise regularisation could produce the collapse.

Decay state. Nine models (three each of gate, substituted and causal) on the 3.21B schedule, stopped at 1B tokens at 8.2× that schedule's final learning rate, keep 47% [39%, 54%] of their gain future-free. That interval contains the 49% future-free share of fully annealed 1B models trained in the same round and excludes the 13% at 3.21B. The 1B models of Table 1, trained together in a later round, keep 29% (Appendix D), so the stopped models are compared only with 1B models of their own round. The stopped models still followed a different learning-rate path over their first 1B tokens than fully annealed 1B models did.

Learning-rate placement. The shared peak rate of 6×10^{-4} is below the best rate at both horizons, more so at the shorter one, so the gate might make up for more of the under-tuned rate there. That account predicts that re-tuning both ends raises the 3.21B future-free part and closes the gap. Re-tuned to 9×10^{-4}, the 1B future-free part is +0.016 [−0.007, +0.038] and the 3.21B part is −0.002, the opposite of the prediction. The re-tuned fall between them is +0.018 [−0.010, +0.045]; the re-tuned 3.21B setting has no gate model, so it gives no share.

Noise regularisation. Noise added to a causal model's attention logits over a ten-fold range of scales gains at most +0.003 at 1B tokens, with the gain peaking inside that range; the peak is 22% of the future-free part of the 1B models of Table 5.

F ALTERNATIVE ACCOUNTS OF THE FUTURE-FREE PART

Table 5 places the substituted model among five other models at 0.16B parameters and 1B tokens, all differing only in where the gate's lookahead keys and values come from. The backward model points the identical machinery at the past. The noise model replaces the lookahead input with Gaussian keys

and values whose magnitudes track the real ones to within 2% through a frozen running average. The register model learns one content-free key and value per layer.

Table 5: Held-out loss at 0.16B parameters and 1B tokens on one shared draw (seed mean), and the gain over the causal baseline. These models span training rounds (Appendix D).

lookahead input	loss	n	gain
own future (gate)	3.410	7	+0.029
unrelated sequence (substituted)	3.424	6	+0.014
none (causal)	3.438	9	—
learned register	3.442	3	−0.004
own past (backward)	3.464	3	−0.026
magnitude-matched noise	3.485	3	−0.047

Two models that remove the mismatch between the lookahead input and the sequence, instead of removing correspondence with the future, do worse (noise) or no better (register) than causal. That ordering is the opposite of what an account predicts in which the substituted model merely routes around an incompatible input. The backward model's deficit, +0.026 [+0.005, +0.047] in loss, excludes zero after widening, while the register's deficit against the substituted model, +0.019 [−0.011, +0.050], does not. A distribution-matched synthetic input and a position-dependent codebook remain untested, so these models rule out a benefit from any extra attendable position without showing that real token content is required.

G Two controls for the accumulated mask's distance term

Two further models test what the accumulated mask keeps under the substitution. A substituted mask that accumulates by the mean, which divides the distance term out and keeps the pairwise term, gains +0.004 at 1B tokens, 9% of the substituted mask's future-free part. Its deficit against the substituted mask, +0.035 [−0.003, +0.072], includes zero after widening, and mean accumulation also changes the penalty's scale, so this control does not separate the distance profile from overall suppression strength. The decay-only model's gain falls by +0.004 [−0.015, +0.023] between 1B and 3.21B tokens. Head to head at 1B tokens, the full accumulated mask gains +0.012 [−0.011, +0.036] more than the decay-only model.

H The same FLOPs spent as depth

Whether the gate is the best use of its own compute depends on the operating point. Matching the gate's per-token FLOPs analytically at 0.16B gives a 14-layer causal baseline; depth also buys 12% more parameters, and at matched parameters the gate wins at every point. The gate's gain over the depth model is +0.003 [−0.010, +0.016] at 0.16B and 1B tokens and −0.006 [−0.024, +0.012] at the Chinchilla ratio across seven seeds. At 0.35B and 1B tokens it is +0.019 [−0.004, +0.042] against a 27-layer baseline. All three intervals include zero after widening, and the conclusions about the split do not rest on them.

I A second model size

At 0.35B parameters and 1B tokens the gate gains +0.030 and the substituted model keeps +0.019 [−0.004, +0.041] of it future-free. Its future-free share, 62% [52%, 72%], sits below the 91% [80%, 103%] measured at a similar number of tokens per parameter at 0.16B, so the two sizes do not line up on that axis, and we fit no size law.

J Second corpus and the boundary convention

We rescored every checkpoint behind Table 2 on a 25M-token slice of WikiText-103 that no model trained on. With end-of-text tokens only between articles, every sign and ordering holds: the mask's future-free part exceeds the pooled gate's by +0.054 [+0.042, +0.066], and the pooled gate's fall

of the future-free part between 1B and 3.21B is $+0.016$, with a Welch interval that includes zero. The comparison depends on where end-of-text tokens sit. Closing every line with one, at about $10\times$ the density of the training split, puts every model whose channel reads its own tokens far behind causal while the substituted models do not trail, and turns the pooled fall to -0.022. Splitting each window at its first end-of-text token locates the effect: before that token the 3.21B pooled gate gains $+0.030$ over causal, and after it the gain is -0.206. The gates treat the token as a signal that an unrelated document follows, so the evaluation's boundary convention is part of any comparison of them. On the training corpus the pooled gate's future-dependent part is not confined to such boundaries. Splitting each FineWeb-Edu position's loss by whether its context holds an end-of-text token, the part in contexts without one is 89% as large at 1B tokens, and 86% as large at 3.21B, as in contexts with one.

K FIDELITY ANCHORS AND THE BOOST TERM

Neither gate is verified against a released artifact: CASTLE released no code, and no absolute published loss is matched. Our anchors are CASTLE's parameter counts, matched exactly at both sizes, and one published delta: at 0.35B parameters our SiLU-form gain over causal, $+0.028$, lies inside CASTLE's published bracket for that size. At 0.16B, extrapolating our measured decay of the gain to CASTLE's training horizon gives about twice the published delta. On BLiMP (Warstadt et al., 2020) the causal models score 73.8% at 1B tokens and 76.6% at the Chinchilla ratio, inside the band published for their model class, and the gate scores what its loss predicts.

CASTLE's SiLU allows the future to raise a logit by up to 0.279; our main gate uses softplus, which only suppresses. The suppression-only gate gains $+0.008$ [$+0.001$, $+0.016$] over the SiLU form at 0.16B and 1B tokens. It also scores slightly lower loss at 3.21B tokens (3.114 against 3.116) and at 0.35B (3.397 against 3.400), so the boost is not what the gain is made of. At context length 2048 the gate's gain is $+0.025$ [$+0.016$, $+0.034$], no larger than at 1024.