Gradient Boosting within a Single Attention Layer*

Saleh Sargolzaei[1]

University of Windsor, Windsor, ON, Canada
sargolz@uwindsor.ca

Abstract. Transformer attention computes a single softmax-weighted average over values—a one-pass estimate that cannot correct its own errors. We introduce *gradient-boosted attention*, which applies the principle of gradient boosting *within* a single attention layer: a second attention pass, with its own learned projections, attends to the prediction error of the first and applies a gated correction. Under a squared reconstruction objective, the construction maps onto Friedman's gradient boosting machine, with each attention pass as a base learner and the per-dimension gate as the shrinkage parameter. We show that a single Hopfield-style update erases all query information orthogonal to the stored-pattern subspace, that further iteration under local contraction can collapse distinct queries to the same fixed point, and that separate projections for the correction pass recover residual information inaccessible to the shared-projection approach of Tukey's twicing. On 10M-token subsets of WikiText-103 and OpenWebText, gradient-boosted attention improves test perplexity by 6.0% and 5.6% over standard attention, outperforming both Twicing Attention and a parameter-matched wider baseline on both benchmarks, with two rounds capturing most of the benefit. The mechanism requires the additive residual structure of Pre-LN transformers: under Post-LN, the same architecture degrades perplexity by 9.6%.

Keywords: Attention mechanism · Gradient boosting · Hopfield networks · Transformers

1 Introduction

The attention mechanism [21] computes a softmax-weighted combination of value vectors conditioned on query–key similarities: the query is compared against all keys once, a probability distribution is formed, and values are averaged accordingly. When this single-pass estimate is poor—because the query is ambiguous, the relevant keys are diluted among distractors, or the softmax assigns weight to incompatible values—there is no within-layer mechanism to detect or correct the error.

A parallel limitation has long been understood in classical machine learning: a single regression tree or kernel smoother produces a biased estimate, and

* Code available at https://github.com/salehsargolzaee/boosted-attention

the bias can be reduced by fitting a second model to the *residual* of the first. This is the insight behind gradient boosting [8], which builds an additive model $F = f_0 + \eta_1 f_1 + \eta_2 f_2 + \cdots$ by sequentially fitting each f_m to the negative gradient of the loss; with strong base learners, two or three rounds often suffice.

We apply this principle within a single attention layer. Round 0 produces an initial estimate via standard attention; the residual—the difference between the input and this estimate—is then passed through a second attention round with its own learned $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$ projections, all three operating on the residual, with a per-dimension learned gate controlling how much of the correction to apply. Because the second round derives keys and values from the residual, it can operate in a different subspace than the first and recover information shared-kernel corrections cannot amplify, giving a drop-in replacement for standard attention that adds one extra set of projections and a small gating network (approximately 18% additional parameters).

Why not simply iterate attention? A natural alternative is to iterate the same operation: feed the output back as the query and repeat, which amounts to running the modern Hopfield network [17] toward its fixed point. We show (Proposition 1) that this systematically destroys query information: under the local contraction conditions of Ramsauer et al. [17], queries in the same region converge to the same fixed point, determined by the stored patterns and temperature alone. Iteration is therefore appropriate for content-addressable memory but harmful for the transformer's task, where the query carries information that must survive to the output, and our negative results—including training with Deep Equilibrium Models (DEQ) [3]—confirm that it failed under every training procedure we tested (Section 5). Gradient-boosted attention avoids this by feeding a *different* signal (the residual) through *different* projections rather than re-processing the same state through the same function. The closest prior work is Twicing Attention [1], which smooths the residual $\mathbf{V} - \mathbf{AV}$ with the *same* matrix $\mathbf{A}$ to yield $(2\mathbf{A} - \mathbf{A}^2)\mathbf{V}$, without learned gating or separate projections; Section 7 discusses it and other related work in detail.

Contributions.

1. We introduce gradient-boosted attention, a multi-round mechanism in which each round corrects the prediction error of previous rounds using separate learned projections and a per-dimension gate, mapping directly onto Friedman's Multiple Additive Regression Trees (MART) framework.
2. We identify a fundamental limitation of iterative retrieval: query information orthogonal to the stored patterns is erased after one step, and further iteration collapses distinct queries to the same fixed point (Proposition 1). We establish a formal correspondence to gradient boosting under a reconstruction objective (Proposition 2), and show that shared-projection corrections like Twicing are confined to the row span of the original value matrix and are not fitted to the residual, while separate projections escape both constraints (Proposition 3).
3. We show that Post-LN placement disrupts the additive structure gradient boosting requires, since the rank-$(d-2)$ LayerNorm Jacobian [22] couples

dimensions and rescales corrections by a data-dependent factor, breaking element-wise gate control (Proposition 4).
4. On 10M-token subsets of WikiText-103 and OpenWebText, gradient-boosted attention improves test perplexity by 6.0% and 5.6% over standard attention, outperforming Twicing Attention and a parameter-matched wider baseline on both, with two rounds capturing most of the benefit; under Post-LN the same architecture instead degrades perplexity by 9.6% while the parameter-matched baseline improves under both placements.

2 Background

2.1 Gradient Boosting

Gradient boosting [8] constructs an additive model $F_M(x) = \sum_{m=0}^{M} \eta_m f_m(x)$ by sequential residual fitting: starting from $F_0(x) = f_0(x)$, each subsequent base learner f_m is fit to the negative gradient of the loss,

$$r_m = -\frac{\partial L(y, F)}{\partial F}\bigg|_{F=F_{m-1}(x)}, \qquad f_m \approx r_m, \qquad F_m = F_{m-1} + \eta_m f_m, \quad (1)$$

where $\eta_m \in (0, 1]$ is a shrinkage parameter regularizing the update. For the squared loss $L = \frac{1}{2}\|y - F\|^2$ the negative gradient is simply the residual $r_m = y - F_{m-1}(x)$. A key empirical finding is that with strong base learners two to three rounds often capture most of the achievable improvement, with rapidly diminishing returns thereafter.

2.2 Attention as Hopfield Retrieval

Ramsauer et al. [17] showed that transformer attention is mathematically equivalent to one step of the modern continuous Hopfield network. Given stored patterns $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbb{R}^{d \times N}$ and a query $\boldsymbol{\xi} \in \mathbb{R}^d$, the update rule is

$$T(\boldsymbol{\xi}) = \mathbf{X}\,\mathrm{softmax}(\beta \mathbf{X}^\top \boldsymbol{\xi}), \quad (2)$$

where $\beta > 0$ is the inverse temperature; setting $\beta = 1/\sqrt{d_h}$ (the per-head dimension) and allowing separate key/value projections recovers standard attention. The update is the negative gradient step $\boldsymbol{\xi}_{\text{new}} = -\nabla_{\boldsymbol{\xi}} E(\boldsymbol{\xi}) + \boldsymbol{\xi} = T(\boldsymbol{\xi})$ of the energy $E(\boldsymbol{\xi}) = -\beta^{-1} \log \sum_i \exp(\beta\, \mathbf{x}_i^\top \boldsymbol{\xi}) + \frac{1}{2}\|\boldsymbol{\xi}\|^2 + \text{const}$, and under sufficient pattern separation relative to β the iterates $T^t(\boldsymbol{\xi})$ converge exponentially to fixed points of three types: global averages over all patterns, metastable states averaging over subsets, and near-single-pattern retrieval.

3 Method

3.1 Gradient-Boosted Attention

Let $\mathbf{x} \in \mathbb{R}^{B \times T \times d}$ denote the input to an attention layer, and let n_h denote the number of attention heads with per-head dimension $d_h = d/n_h$. We define M

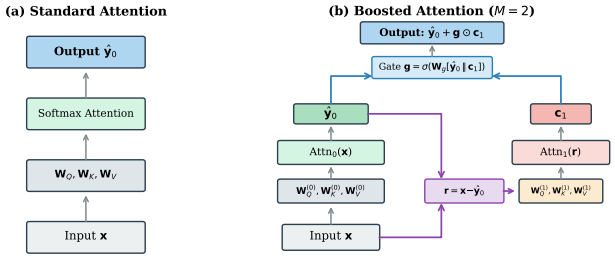

Fig. 1. (a) Standard attention computes a single softmax-weighted average. (b) Gradient-boosted attention (M=2) adds a second pass where the prediction residual $\mathbf{r} = \mathbf{x} - \hat{\mathbf{y}}_0$ is projected through separate $\mathbf{W}_Q^{(1)}, \mathbf{W}_K^{(1)}, \mathbf{W}_V^{(1)}$ to produce queries, keys, and values. A learned gate $\mathbf{g}$ controls the per-dimension correction magnitude.

Algorithm 1 Gradient-Boosted Attention Forward Pass

Input: $\mathbf{x} \in \mathbb{R}^{B \times T \times d}$, number of rounds M
$\hat{\mathbf{y}}_0 \leftarrow \text{Attn}_0(\mathbf{x})$ {Round 0: initial estimate}
$F \leftarrow \hat{\mathbf{y}}_0$
for $m = 1$ to $M - 1$ **do**
  $\mathbf{r}_m \leftarrow \mathbf{x} - F$ {Prediction error (negative gradient for L_2 loss)}
  $\mathbf{c}_m \leftarrow \text{Attn}_m(\mathbf{r}_m)$ {Q, K, V all from residual via separate projections}
  $\mathbf{g}_m \leftarrow \sigma\left(\mathbf{W}_g^{(m)}[F \,\|\, \mathbf{c}_m]\right) \in [0,1]^d$ {Per-dimension shrinkage gate}
  $F \leftarrow F + \mathbf{g}_m \odot \mathbf{c}_m$ {Gated correction}
end for
Return: $\mathbf{W}_{\text{out}} \cdot F$

attention rounds, each with its own learned projections $\mathbf{W}_Q^{(m)}, \mathbf{W}_K^{(m)}, \mathbf{W}_V^{(m)} \in \mathbb{R}^{d_h \times d}$ (per head):

$$\text{Attn}_m(\mathbf{z}) = \text{softmax}\left(\frac{\mathbf{W}_Q^{(m)}\mathbf{z} \cdot (\mathbf{W}_K^{(m)}\mathbf{z})^\top}{\sqrt{d_h}}\right)\mathbf{W}_V^{(m)}\mathbf{z}. \tag{3}$$

All three projections are applied to the same input $\mathbf{z}$: the original input $\mathbf{x}$ for round 0, and the current residual $\mathbf{r}_m$ for rounds $m \geq 1$, so each correction round operates in its own subspace of the residual, with keys and values reflecting what has *not yet been predicted* rather than the original context. Algorithm 1 summarizes the forward pass.

The gate as a learned shrinkage parameter. In Friedman's framework the shrinkage parameter η_m is a scalar regularizing each boosting step. Our gate $\mathbf{g}_m = \sigma(\mathbf{W}_g^{(m)}[F \| \mathbf{c}_m]) \in [0,1]^d$ generalizes this in two ways: it is per-dimension, allowing selective correction along different feature directions, and input-dependent, letting the correction magnitude vary with the current prediction and the proposed correction; collapsed to a scalar it reduces exactly to η_m. Our ablations (Section 6.3) show all gate variants improve over the single-round baseline, suggesting the residual-attention mechanism is the primary ingredient.

Table 1. Correspondence between gradient boosting (MART) and gradient-boosted attention.

Component	MART [8]	Ours
Initial estimate	$f_0(x)$ (e.g., mean of targets)	$\hat{\mathbf{y}}_0 = \mathrm{Attn}_0(\mathbf{x})$
Pseudo-residual	$r_m = y - F_{m-1}(x)$	$\mathbf{r}_m = \mathbf{x} - F_{m-1}$
Base learner	f_m fit to r_m	$\mathbf{c}_m = \mathrm{Attn}_m(\mathbf{r}_m)$
Shrinkage	Scalar $\eta_m \in (0, 1]$	Gate $\mathbf{g}_m = \sigma(\mathbf{W}_g^{(m)}[\cdot]) \in [0, 1]^d$
Update	$F_m = F_{m-1} + \eta_m f_m$	$F_m = F_{m-1} + \mathbf{g}_m \odot \mathbf{c}_m$

Computational cost. With M rounds the attention computation is approximately M times that of standard attention, plus a small gating network; in practice $M = 2$ adds approximately 18% total parameters, whereas Twicing Attention [1] adds zero parameters by reusing the same attention matrix. Our additional cost buys separate projections for the correction pass, which Proposition 3 shows can recover information inaccessible to shared-kernel correction, and Section 6 shows the improvement cannot be replicated by simply widening the standard model to match the parameter count.

3.2 Connection to MART

Table 1 makes the correspondence explicit: under the squared reconstruction objective $L = \frac{1}{2}\|\mathbf{x} - F\|^2$, Algorithm 1 instantiates gradient boosting exactly, with the residual as the negative gradient, each attention round as a base learner, and the gate as the shrinkage parameter. This is an internal objective governing the within-layer residual computation; the model is trained end-to-end with the task loss (e.g., cross-entropy for language modeling).

4 Theoretical Analysis

4.1 Iterating Retrieval Erases Query Information

The most natural way to "boost" attention would be to iterate the same operation: compute $T(\boldsymbol{\xi})$, then $T(T(\boldsymbol{\xi}))$, and so on, converging to the Hopfield fixed point. We state the result in the Hopfield setting (Eq. 2), where the analysis is cleanest; part (a) generalizes directly to standard attention with separate projections, since the softmax always produces convex weights, so the output lies in $\mathrm{conv}(\mathbf{V})$ regardless of how $\mathbf{Q}$, $\mathbf{K}$, $\mathbf{V}$ are computed, and Section 5 confirms that iteration fails in practice even with learned $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$.

Proposition 1 (One-step projection and information loss). *Let $\mathbf{X} \in \mathbb{R}^{d \times N}$ be a matrix of stored patterns and $T(\boldsymbol{\xi}) = \mathbf{X}\,\mathrm{softmax}(\beta \mathbf{X}^\top \boldsymbol{\xi})$ the Hopfield update (2). Then:*

(a) For every $\boldsymbol{\xi}$, $T(\boldsymbol{\xi}) \in \mathrm{conv}(\mathbf{X}) \subseteq \mathrm{col}(\mathbf{X})$.

(b) If $\boldsymbol{\xi} = \boldsymbol{\xi}_\parallel + \boldsymbol{\xi}_\perp$ with $\boldsymbol{\xi}_\parallel \in \mathrm{col}(\mathbf{X})$ and $\boldsymbol{\xi}_\perp \perp \mathrm{col}(\mathbf{X})$, then $T(\boldsymbol{\xi}) = T(\boldsymbol{\xi}_\parallel)$. Hence all information in the component orthogonal to $\mathrm{col}(\mathbf{X})$ is erased after a single step.

(c) Any fixed point $\boldsymbol{\xi}^*$ satisfies $\boldsymbol{\xi}^* = \mathbf{X}\,\mathrm{softmax}(\beta \mathbf{X}^\top \boldsymbol{\xi}^*)$, so every fixed point lies in $\mathrm{conv}(\mathbf{X})$.

Proof. (a) Since softmax returns nonnegative weights summing to one, $T(\boldsymbol{\xi}) = \sum_{i=1}^{N} \alpha_i \mathbf{x}_i$ with $\alpha_i \geq 0, \sum_i \alpha_i = 1$, a convex combination of the columns of $\mathbf{X}$. (b) The update depends on $\boldsymbol{\xi}$ only through the score vector $\mathbf{X}^\top \boldsymbol{\xi}$; since $\mathbf{X}^\top \boldsymbol{\xi}_\perp = 0$, we have $\mathbf{X}^\top \boldsymbol{\xi} = \mathbf{X}^\top \boldsymbol{\xi}_\parallel$, so $T(\boldsymbol{\xi}) = T(\boldsymbol{\xi}_\parallel)$. (c) Setting $\boldsymbol{\xi} = \boldsymbol{\xi}^*$ in (a) gives $\boldsymbol{\xi}^* = T(\boldsymbol{\xi}^*) \in \mathrm{conv}(\mathbf{X})$. □

Remark 1. After one step the state is restricted to $\mathrm{conv}(\mathbf{X})$ and depends on the query only through $\mathbf{X}^\top \boldsymbol{\xi}$, so any query information outside $\mathrm{col}(\mathbf{X})$ is irrecoverably discarded. Further iteration compounds this: Ramsauer et al. [17] showed (their Lemma A7) that under sufficient pattern separation T is a local contraction within a sphere around each stored pattern, with a unique fixed point per sphere by the Banach fixed-point theorem. Since these fixed points are determined by $\mathbf{X}$ and β alone, distinct queries within the same contraction region converge to the same output. This is consistent with Smart et al. [19], who showed that a single attention step can realize Bayes-optimal denoising behavior outperforming both exact retrieval of a stored pattern and convergence to a potentially spurious local minimum, and with Section 5, where iterating trained attention to convergence degrades retrieval accuracy to near chance while the one-step output retains useful query information.

4.2 Gradient-Boosted Attention as Gradient Boosting

Proposition 2 (MART equivalence). *Consider the squared prediction loss* $L(\mathbf{x}, F) = \frac{1}{2}\|\mathbf{x} - F\|^2$*, where* $\mathbf{x}$ *is the input and* F *is the cumulative prediction from attention rounds* $0, \ldots, m-1$*. Then the negative functional gradient of* L *with respect to* F *is* $-\nabla_F L(\mathbf{x}, F) = \mathbf{x} - F = \mathbf{r}_m$*, which is exactly the residual computed in line 5 of Algorithm 1. Each correction round* $\mathrm{Attn}_m(\mathbf{r}_m)$ *fits a base learner (attention with separate projections) to this negative gradient, and the gated update* $F_m = F_{m-1} + \mathbf{g}_m \odot \mathrm{Attn}_m(\mathbf{r}_m)$ *is a shrinkage-regularized gradient boosting step.*

Proof. The gradient is immediate from the loss definition. The structural correspondence between Algorithm 1 and the boosting update (1) is exact when the loss is squared and the gate $\mathbf{g}_m$ plays the role of η_m. The only difference is that η_m in MART is a scalar optimized via line search, while $\mathbf{g}_m$ is a per-dimension function of the current state, learned jointly with the rest of the model by backpropagation. □

Cheng et al. [5] proved that the L-layer transformer implements L steps of functional gradient descent in a reproducing kernel Hilbert space, but did not connect this to the classical boosting literature or to Friedman's MART; our construction makes the connection explicit and operates at a finer granularity, boosting within a single layer rather than across layers.

4.3 Why Separate Projections Are Necessary

Abdullaev and Nguyen [1] proposed Twicing Attention, which smooths the residual $\mathbf{V} - \mathbf{AV}$ using the *same* attention matrix $\mathbf{A}$, yielding the output $(2\mathbf{A} - \mathbf{A}^2)\mathbf{V}$. We identify two structural limitations of this shared-projection approach that gradient-boosted attention overcomes.

Proposition 3 (Limitations of shared-projection correction). *Let* $\mathbf{A} = \mathrm{softmax}(\mathbf{QK}^\top/\sqrt{d_h}) \in \mathbb{R}^{N \times N}$ *and* $\mathbf{V} = \mathbf{W}_V^{(0)}\mathbf{x} \in \mathbb{R}^{N \times d}$*.*

(a) Row-span confinement. *For any polynomial* p*, each row of* $p(\mathbf{A})\mathbf{V}$ *is a linear combination of the rows of* $\mathbf{V}$*. In particular, the Twicing output* $(2\mathbf{A} - \mathbf{A}^2)\mathbf{V}$*—and any higher-degree polynomial extension—is confined to* $\mathrm{rowspan}(\mathbf{V})$*, the subspace spanned by the original value vectors.*

(b) Fixed correction. *Twicing's correction* $\mathbf{A}(\mathbf{I}-\mathbf{A})\mathbf{V}$ *is uniquely determined by* $\mathbf{A}$ *and* $\mathbf{V}$ *with no free parameters: in the MART framework of Proposition 2, the weak learner is not fitted to the residual.*

(c) Gradient-boosted attention escapes both constraints. *The correction* $\mathbf{g} \odot \mathbf{A}'\mathbf{V}'$, *where* $\mathbf{A}' = \mathrm{softmax}(\mathbf{Q}'\mathbf{K}'^{\top}/\sqrt{d_h})$ *with* $\mathbf{Q}' = \mathbf{W}_Q^{(1)}\mathbf{r}$, $\mathbf{K}' = \mathbf{W}_K^{(1)}\mathbf{r}$, $\mathbf{V}' = \mathbf{W}_V^{(1)}\mathbf{r}$, *produces value vectors generically outside* $\mathrm{rowspan}(\mathbf{V})$, *and the projections are fitted to the residual by backpropagation.*

Proof. (a) $p(\mathbf{A})\mathbf{V} = \sum_k c_k \mathbf{A}^k\mathbf{V}$; each row of $\mathbf{AV}$ is $\sum_j A_{ij}\mathbf{v}_j$, a linear combination of value vectors, and by induction the same holds for $\mathbf{A}^k\mathbf{V}$, hence for any $p(\mathbf{A})\mathbf{V}$. (b) The output $(2\mathbf{A}-\mathbf{A}^2)\mathbf{V}$ is a deterministic function of $\mathbf{A}$ and $\mathbf{V}$; the correction introduces no learnable parameters. (c) Since $\mathbf{V}' = \mathbf{W}_V^{(1)}\mathbf{r}$ and $\mathbf{V} = \mathbf{W}_V^{(0)}\mathbf{x}$ use different projections of different inputs ($\mathbf{r} = \mathbf{x} - F_0 \neq \mathbf{x}$ in general), $\mathrm{rowspan}(\mathbf{V}')$ and $\mathrm{rowspan}(\mathbf{V})$ are generically distinct, and $\mathbf{W}_Q^{(1)}, \mathbf{W}_K^{(1)}, \mathbf{W}_V^{(1)}$ are trained jointly, fitting the weak learner to the residual signal. □

Abdullaev and Nguyen [1] exploit the spectral structure of part (a) positively, showing that $\tilde{p}(\lambda) = 2\lambda - \lambda^2$ slows eigencapacity decay from $O(n^{-1})$ to $O(n^{-1/2})$ across layers, mitigating representation collapse without additional parameters. Gradient-boosted attention addresses a complementary problem: rather than optimizing the spectral filter within $\mathrm{rowspan}(\mathbf{V})$, it introduces independent projections fitted to the prediction residual, producing value vectors that can represent directions the original $\mathbf{V}$ cannot express.

4.4 Normalization Placement and the Additive Structure

Gradient boosting builds an additive model: the cumulative prediction $F_M = F_0 + \sum_{m=1}^{M} \eta_m f_m$ must preserve each correction term without distortion. The LayerNorm Jacobian and its properties are well established [22]; we restate the relevant facts to make explicit their consequence for gradient-boosted attention.

Proposition 4 (Normalization placement and additive preservation). *Consider a transformer layer with sublayer output* $\mathbf{f} \in \mathbb{R}^d$ *(e.g., the output of gradient-boosted attention).*

(a) Under Pre-LN placement, $\mathbf{h}_{l+1} = \mathbf{h}_l + \mathbf{f}(\mathrm{LN}(\mathbf{h}_l))$, *the Jacobian of* $\mathbf{h}_{l+1}$ *with respect to* $\mathbf{f}$ *is the identity,* $\partial\mathbf{h}_{l+1}/\partial\mathbf{f} = \mathbf{I}$. *Each component of* $\mathbf{f}$*—including any gated correction* $\mathbf{g}_m \odot \mathbf{c}_m$*—contributes independently and without distortion to the residual stream.*

(b) Under Post-LN placement, $\mathbf{h}_{l+1} = \mathrm{LN}(\mathbf{h}_l + \mathbf{f}(\mathbf{h}_l))$. *Let* $\mathbf{u} = \mathbf{h}_l + \mathbf{f}$, $\mu = \frac{1}{d}\sum_j u_j$, $\sigma^2 = \frac{1}{d}\sum_j (u_j - \mu)^2$, *and* $\hat{\mathbf{u}} = (\mathbf{u} - \mu\mathbf{1})/\sigma$. *The Jacobian is the well-known LayerNorm Jacobian [22]:*

$$\frac{\partial \mathbf{h}_{l+1}}{\partial \mathbf{f}} = \frac{\mathrm{diag}(\boldsymbol{\gamma})}{\sigma}\left(\mathbf{I} - \frac{1}{d}\mathbf{1}\mathbf{1}^{\top} - \frac{1}{d}\hat{\mathbf{u}}\hat{\mathbf{u}}^{\top}\right), \tag{4}$$

where $\boldsymbol{\gamma} \in \mathbb{R}^d$ *is the learned scale parameter of LayerNorm. This matrix has rank* $d-2$*: it annihilates the constant vector* $\mathbf{1}$ *and the standardized activation* $\hat{\mathbf{u}}$, *and couples all surviving dimensions through the outer-product terms.*

Proof. (a) Since $\mathbf{h}_{l+1} = \mathbf{h}_l + \mathbf{f}$ and $\mathbf{h}_l$ does not depend on $\mathbf{f}$, the Jacobian is immediate. (b) The i-th component of $\mathrm{LN}(\mathbf{u})$ is $\gamma_i(u_i - \mu)/\sigma + b_i$ with b_i the learned bias; using $\partial\mu/\partial u_j = 1/d$ and $\partial\sigma/\partial u_j = (u_j - \mu)/(d\sigma)$ gives $\partial[\mathrm{LN}(\mathbf{u})]_i/\partial u_j = \frac{\gamma_i}{\sigma}(\delta_{ij} - \frac{1}{d} - \frac{\hat{u}_i\hat{u}_j}{d})$.

Since $\mathbf{u} = \mathbf{h}_l + \mathbf{f}$ with $\partial u_j / \partial f_j = 1$, this gives (4). For the rank: $\hat{\mathbf{u}}$ is centered ($\mathbf{1}^\top \hat{\mathbf{u}} = 0$), so $\mathbf{1}$ and $\hat{\mathbf{u}}$ are linearly independent; both are null vectors of $\mathbf{I} - \frac{1}{d}\mathbf{1}\mathbf{1}^\top - \frac{1}{d}\hat{\mathbf{u}}\hat{\mathbf{u}}^\top$ (using $\|\hat{\mathbf{u}}\|^2 = d$), and any $\mathbf{v} \perp \{\mathbf{1}, \hat{\mathbf{u}}\}$ satisfies $(\mathbf{I} - \frac{1}{d}\mathbf{1}\mathbf{1}^\top - \frac{1}{d}\hat{\mathbf{u}}\hat{\mathbf{u}}^\top)\mathbf{v} = \mathbf{v}$, giving rank exactly $d - 2$. □

Remark 2. Under Pre-LN, Proposition 4(a) ensures the gated correction $\mathbf{g}_m \odot \mathbf{c}_m$ enters the residual stream unchanged, preserving the additive structure the MART correspondence requires; more broadly, the residual stream after L Pre-LN layers decomposes as $\mathbf{h}_L = \mathbf{h}_0 + \sum_l \mathbf{f}_l$, matching the additive form of gradient boosting across layers [5]. Under Post-LN, the rank-$(d-2)$ Jacobian projects out two degrees of freedom (the mean direction $\mathbf{1}$ and the variance-aligned direction $\hat{\mathbf{u}}$), rescales the remainder by the data-dependent factor $\boldsymbol{\gamma}/\sigma$, and couples dimensions through $\hat{\mathbf{u}}\hat{\mathbf{u}}^\top$, mixing corrections across dimensions and undermining the element-wise control the gate was learned to apply. Section 6.2 confirms this empirically. The requirement echoes the identity-mapping principle of He et al. [10]: in ResNets, placing normalization inside the residual branch—preserving a clean additive shortcut—improves both gradient flow and final accuracy.

5 Why Iterating Attention Fails: Negative Results

We summarize here the empirical evidence that motivated the gradient-boosted attention design; these results complement Proposition 1 by showing that the failure of iterative attention persists across the training methods and architectural modifications we tested.

Synthetic pattern retrieval task. These experiments and the ablations of Section 6.3 use a synthetic retrieval task: K unit-normalized patterns $\mathbf{p}_1, \ldots, \mathbf{p}_K \in \mathbb{R}^d$ are sampled uniformly from the unit sphere, and a query is formed by selecting $\mathbf{p}_j$ uniformly at random and adding isotropic Gaussian noise, $\tilde{\mathbf{x}} = \mathbf{p}_j + \boldsymbol{\varepsilon}$, $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$. Retrieval accuracy is the fraction of queries whose output is nearest (in Euclidean distance) to the generating pattern. All synthetic experiments train with mean squared error $\|\hat{\mathbf{x}} - \mathbf{p}_j\|^2$—consistent with the squared reconstruction objective underlying the MART correspondence—using Adam, learning rate 3×10^{-3}, batch size 512, and 150 epochs. The task admits a closed-form Bayes-optimal predictor [19], $f_{\text{opt}}(\tilde{\mathbf{x}}) = \sum_k \mathbf{p}_k e^{\langle \mathbf{p}_k, \tilde{\mathbf{x}} \rangle / \sigma^2} / \sum_k e^{\langle \mathbf{p}_k, \tilde{\mathbf{x}} \rangle / \sigma^2}$: softmax attention over the patterns with $\beta = 1/\sigma^2$ and identity projections, which we report as an analytical ceiling.

Iterating attention destroys accuracy. Across all six configurations in Table 2, both converged variants—trained with backpropagation through 20 unrolled iterations and with implicit differentiation via Deep Equilibrium Models [3]—achieve accuracy indistinguishable from random chance (e.g., 6.7% vs. 6.3% chance for K=16), while one-step attention reaches 26–71% depending on difficulty, approaching the analytical Bayes-optimal ceiling. This is consistent with Proposition 1 and the local-contraction analysis of Ramsauer et al. [17]: queries in the same contraction region converge to the same fixed point regardless of their initial position.

Table 2. Retrieval accuracy (%) on the synthetic pattern retrieval task. Both converged variants—trained by backpropagation through 20 iterations or by DEQ implicit differentiation—collapse to near chance, while one-step attention approaches the analytical Bayes-optimal ceiling.

	d=16, K=4		d=32, K=8		d=64, K=16	
	σ=0.5	σ=0.8	σ=0.5	σ=0.8	σ=0.5	σ=0.8
Random chance	25.0	25.0	12.5	12.5	6.3	6.3
Bayes optimal	80.7	61.9	68.8	45.3	58.1	32.6
One-step (trained)	71.1	55.0	53.1	36.4	46.5	26.1
Converged (trained, backprop)	25.2	25.2	12.1	12.1	6.7	6.7
Converged (trained, DEQ)	25.2	25.2	12.1	12.1	6.7	6.7

The failure is structural, not a training artifact. One might suspect vanishing gradients through many iteration steps cause the converged path to fail, but our two controls rule this out: backpropagation through the unrolled graph provides gradients that may vanish over 20 steps, while Deep Equilibrium Models compute exact gradients at the fixed point via the implicit function theorem, bypassing the iteration graph entirely, and both achieve identical random-chance accuracy. The information loss therefore follows from the contraction dynamics themselves, not from how gradients are computed. We further trained routing gates with five feature sets (both outputs, their difference, scalar divergence, attention entropy) to decide when to trust the converged path; all learned to always select the one-step output, matching the one-step baseline exactly, so the converged path carries no complementary signal.

6 Experiments

6.1 Language Modeling

Setup. We train small transformer language models from scratch and compare four configurations: (1) **Standard** causal multi-head attention, $d = 256$, 4 layers, 4 heads (7.4M parameters); (2) **Twicing** Attention [1], same architecture, applying $(2\mathbf{A} - \mathbf{A}^2)\mathbf{V}$ at no parameter cost (7.4M); (3) a **parameter-fair** standard model with $d = 288$ (8.8M); and (4) **gradient-boosted attention** ($M = 2$) with separate QKV projections per round and a per-dimension sigmoid gate (8.7M). All models use Pre-LN placement [22], byte-pair encoding (BPE) tokenization (16,384 types), sequence length 256, FFN hidden dimension $4d$, dropout 0.1, weight tying, AdamW with learning rate 3×10^{-4}, weight decay 0.01, cosine schedule with 1500 warmup steps, batch size 32, and gradient clipping at 1.0; the gate is Linear($2d \to d$) followed by a sigmoid. We evaluate on 10M-token subsets of WikiText-103 [14] (full corpus ∼103M tokens) and of OpenWebText [9], an open reproduction of WebText containing Reddit-upvoted web content, each with a separately trained BPE tokenizer, so perplexity is comparable only within a dataset. We train for 15 epochs and report test perplexity averaged over 2 seeds (42, 123). Training used one NVIDIA RTX 2000 Ada (16GB) for

Table 3. Test perplexity (lower is better); values are not comparable across datasets (different tokenizers). Under Pre-LN, gradient-boosted attention beats every baseline on both benchmarks, including a wider model with matched parameter count. Under Post-LN (WikiText-103, single seed) the ordering reverses for the boosted model only.

		WikiText-103		**OpenWebText**
Model	**Params**	**Pre-LN**	**Post-LN**	**Pre-LN**
Standard (d=256)	7.4M	72.2 ± 0.3	77.2	114.9 ± 0.5
Twicing (d=256)	7.4M	69.6 ± 0.1	—	110.7 ± 0.3
Standard (d=288, param-fair)	8.8M	69.0 ± 0.1	71.8	110.2 ± 0.5
Gradient-boosted (M=2)	8.7M	$\mathbf{67.9 \pm 0.1}$	84.6	$\mathbf{108.5 \pm 0.9}$
Δ Boosted vs. Standard		−6.0%	+9.6%	−5.6%
Δ Param-fair vs. Standard		−4.4%	−7.0%	−4.1%

WikiText-103 and one A100 80GB for OpenWebText and the Post-LN runs, at ∼30 and ∼10 minutes per run; synthetic ablations ran on Apple M1 (MPS).

Results. Gradient-boosted attention achieves 67.9 perplexity on WikiText-103 and 108.5 on OpenWebText (Table 3), the best on both—a 6.0% and 5.6% reduction over standard attention. Two comparisons isolate the source of the improvement. Twicing and our method both correct the initial estimate, but Twicing uses a fixed shared-kernel correction while ours uses separate projections per round with a learned gate; the consistent gap across both datasets (−1.7 and −2.2 points) validates Proposition 3, that independent projections fit residual structure a shared kernel cannot. A wider standard model (d=288, 8.8M parameters) slightly exceeds the boosted model in parameter count yet falls short on both benchmarks, isolating the residual-fitting architecture from additional capacity alone. Gradient-boosted attention also converges in fewer optimizer steps, reaching the parameter-fair baseline's final validation perplexity (69.6) by epoch 12 of 15 (20% fewer) and the standard model's (72.2) by epoch 10 (33% fewer), though each step costs more due to the second attention round.

Key/value source. An alternative design derives keys and values from the original input $\mathbf{x}$ rather than the residual, i.e. $\mathrm{softmax}(\mathbf{W}_Q^{(m)}\mathbf{r}_m(\mathbf{W}_K^{(m)}\mathbf{x})^\top/\sqrt{d_h})\,\mathbf{W}_V^{(m)}\mathbf{x}$, preserving the asymmetry between what the correction *attends to* (the residual) and what it *retrieves from* (the original context), analogous to cross-attention. On WikiText-103 under the same setup this variant reaches 69.3 test and 69.5 validation perplexity, against 67.9 and 68.1 for the residual-derived design—1.4 points worse. Residual-derived keys and values let the correction round operate entirely within the residual subspace, producing values tailored to the error signal, whereas deriving them from $\mathbf{x}$ confines both rounds to the same representational space, consistent with Proposition 3; our implementation supports both via a `kv_source` flag.

6.2 Normalization Ablation

Proposition 4 predicts that Post-LN placement should disrupt gradient-boosted attention by distorting the additive correction through the LayerNorm Jacobian.

Table 4. Ablations on the synthetic pattern retrieval task (d=64, K=16, σ=0.5); Bayes-optimal ceiling 58.1%. **Top:** effect of the number of boosting rounds. **Bottom:** effect of gate type, all at M=2 rounds.

Rounds (M)	1	2	3	4	5
Accuracy (%)	46.0	55.5	57.6	58.1	57.0
Δ from M=1	—	+9.5	+11.6	+12.1	+11.0
Gate type (M=2)	None	Scalar	MLP	Baseline (M=1)	
Accuracy (%)	55.4	56.5	55.6	48.2	
Δ from baseline	+7.2	+8.3	+7.4	—	

We retrained three models on WikiText-103 with Post-LN, keeping all other hyperparameters identical (single seed); results appear in the Post-LN column of Table 3. Gradient-boosted attention degrades perplexity from 77.2 to 84.6 (+9.6%), *reversing* the 6.0% Pre-LN improvement, while the parameter-fair baseline—at nearly identical parameter count (8.8M vs. 8.7M)—instead *improves* (77.2 $\rightarrow$ 71.8, -7.0%), ruling out the hypothesis that extra parameters are generically harder to train under Post-LN. The failure is thus specific to the boosting mechanism: Post-LN's rank-($d-2$) Jacobian (Proposition 4(b)) distorts the gated correction before it reaches the next layer, breaking the additive structure gradient boosting requires. This is consistent with He et al. [10], who found pre-activation placement—preserving a clean identity shortcut—essential for depth-dependent refinement in residual networks.

6.3 Ablation Studies

We conduct ablations on the synthetic pattern retrieval task ($d = 64$, $K = 16$ patterns, noise $\sigma = 0.5$) where the mechanism is most transparent.

Number of boosting rounds. The jump from 1 to 2 rounds (+9.5 points) accounts for the vast majority of the improvement, raising accuracy from 46.0% to 55.5% and closing most of the gap to the Bayes-optimal ceiling of 58.1% (Table 4, top); further rounds yield diminishing returns, with 4 rounds (58.1%) nearly matching the ceiling. This mirrors gradient boosting with strong base learners [8], where early rounds capture the dominant signal and later ones offer marginal gains.

Gate type. We compare three configurations at M=2: no gate ($\mathbf{g} = \mathbf{1}$, pure additive correction), scalar gate (one learned shrinkage value per round, matching the MART η_m), and per-dimension MLP gate. All three give similar accuracy, with even the no-gate variant improving substantially over the single-round baseline (Table 4, bottom). That a single scalar matches the per-dimension MLP gate suggests the separate projections already absorb per-dimension adjustment, leaving the gate to control only overall magnitude; the narrow spread confirms the residual-attention mechanism, not the gating, is the key ingredient.

Scaling with problem difficulty. The benefit grows with difficulty: across configurations from (d=32, K=8) to (d=128, K=32), the improvement over the single-round baseline increases with the number of patterns and with moderate

noise ($\sigma \leq 0.5$), reaching at σ=0.3 gains of +17.8 points for (d=64, K=16), +14.6 for (d=32, K=8), and +13.5 for (d=128, K=32)—precisely the regime where the initial pass makes systematic but correctable errors. At very high noise ($\sigma \geq 0.8$) even the base learner's errors become hard to correct and the gains diminish.

6.4 Analysis

Gate values across layers. The gate is an input-dependent MLP mapping the cumulative output and correction to a per-dimension mask in $[0, 1]$. Averaged over 50 test sequences, layer 0 is the most conservative (mean 0.33, std. 0.08), admitting only a modest fraction of the correction; layer 1 uses it most aggressively (mean 0.49) with the highest variance across dimensions ($\sigma = 0.22$), so some dimensions rely heavily on the correction while others suppress it; layers 2 and 3 fall in between (0.39, 0.36). Given the capacity the gate thus learns non-trivial dimension-specific shrinkage, the per-dimension generalization of the MART parameter of Section 3.2—though, as the ablation shows, a scalar gate achieves comparable accuracy because the separate projections absorb per-dimension adjustment when the gate cannot.

Convex hull escape. Proposition 1(a) maps every single-step output into $\text{conv}(\mathbf{V}^{(0)})$, while Proposition 3 predicts the gated correction can move it *outside* that hull, since $\mathbf{V}^{(1)} = \mathbf{W}_V^{(1)}\mathbf{r}$ spans a different subspace than $\mathbf{V}^{(0)} = \mathbf{W}_V^{(0)}\mathbf{x}$. For each head at position t we compute the exact ℓ_2 distance from the boosted output $\mathbf{o} \in \mathbb{R}^{d_h}$ to the hull of the causally visible round-0 values by solving $\min_{\boldsymbol{\alpha}} \|\mathbf{o} - \sum_{i=1}^{t} \alpha_i \mathbf{v}_i^{(0)}\|_2$ subject to $\alpha_i \geq 0$, $\sum_i \alpha_i = 1$, via Sequential Least-Squares Programming (`scipy.optimize.minimize`) from a uniform initialization, restricting to $t \leq 64$ and sampling 600 (position, head) pairs per layer from 30 test sequences ($d_h = 64$); any positive distance indicates escape. All 2,400 measured outputs lie strictly outside $\text{conv}(\mathbf{V}^{(0)})$ across the four layers (100% escape rate), and the ranking mirrors the gate analysis: layer 1, applying the strongest correction (mean gate 0.49), escapes farthest (mean distance 5.19), layer 0 the most conservative (gate 0.33) stays closest (0.88), and layers 2 and 3 fall in between (4.25, 3.41)—confirming that the correction round produces representations outside the original value space, as Proposition 3 predicts.

Attention entropy and example-level corrections. Attention entropy reveals a learned division of labor: round 0 is consistently *more diffuse* than standard attention (mean 3.34 vs. 2.71 nats) while round 1 is more focused (2.55 nats), so the initial round casts a wider net and delegates sharpening to the correction round. The effect is strongest in layer 1, where round 0 entropy exceeds standard by 27% while round 1 drops 45% below it, coinciding with the highest gate values—suggesting the model relies on the correction most when it can be made precise. Individual predictions agree: for "Ke" → "iser" (completing Keiser), round 1 concentrates on "Ke" and the nearby geographical context while standard attention predicts "ong," confused by the earlier substring "Yongsan"; for "iron @-@ h" → "ul" (completing "hull"), round 1 attends sharply to "iron," the hyphen, and "h," while standard attention predicts the wrong subword. Both come from the top of the per-token improvement distribution (> 4.8 nats) and are best-case rather than typical; the overall 4.3-point perplexity improvement is the average across all tokens.

7 Related Work

Hopfield networks and transformers as gradient descent. Ramsauer et al. [17] established the equivalence between attention and one step of the modern continuous

Hopfield network, identifying three fixed-point types, and Smart et al. [19] showed that a single trained attention step implements a gradient descent update on a context-aware dense associative memory landscape whose one-step estimate can be closer to the Bayes-optimal denoiser than the converged fixed point; we extend these by proving why convergence fails (Proposition 1) and proposing a constructive alternative. Cheng et al. [5] proved that each layer implements one step of functional gradient descent in the RKHS of the attention kernel without connecting this to boosting; Huang et al. [12] interpreted ResNet blocks as boosting stages and Siu [18] formalized the analogy; and Badirli et al. [2] used shallow networks as base learners in a boosting ensemble. Our work differs from all of these by applying boosting within a single attention operation, at finer granularity than the layer level.

Residual correction in attention. The closest prior work is Twicing Attention [1], which applies Tukey's twicing [20] within each attention layer, justified from nonparametric statistics where twicing reduces the bias of the Nadaraya–Watson estimator [15]. Our approach differs in three ways: (i) we apply separate learned projections to the *residual*, producing new queries, keys, *and* values that can span a different subspace than the original attention (Proposition 3); (ii) we include a learned gate that adapts the correction magnitude per-input and per-dimension; (iii) our framing as gradient boosting suggests natural extensions such as more rounds and adaptive halting. Whether the fixed correction $(2\mathbf{A} - \mathbf{A}^2)\mathbf{V}$ is optimal, or whether learned projections do better, is addressed directly by Section 6.

Attention variants and iterative computation. Differential Transformer [23] takes the difference of two parallel attention maps to cancel common-mode noise—a parallel subtractive mechanism orthogonal to our sequential error-corrective one, and combinable with it. Gated Attention [16] adds a post-attention sigmoid gate per head, introducing non-linearity and sparsity into a single pass, but computes no second pass and does not attend to residuals. Universal Transformers [6], Deep Equilibrium Models [3], and PonderNet [4] iterate the *same* function on the accumulated state—with shared parameters and adaptive halting, implicit differentiation at the fixed point, and learned stopping respectively—whereas gradient-boosted attention operates on the *residual* with *different* projections, a distinction Proposition 1 shows is critical.

Normalization placement and cross-layer residual methods. He et al. [10] showed that pre-activation placement in ResNets preserves a clean identity shortcut—the direct precursor of Pre-LN in transformers; Xiong et al. [22] proved that Post-LN produces gradient magnitudes growing with layer index at initialization while Pre-LN keeps them bounded; and Elhage et al. [7] formalized the residual stream as an additive accumulator to which each head and layer writes an independent contribution. That additive decomposition—preserved by Pre-LN and disrupted by Post-LN—is precisely the structure gradient boosting requires, and Proposition 4 connects these observations. Separately, DeepCrossAttention [11] replaces residual connections with depth-wise cross-attention and Attention Residuals [13] replaces fixed skip connections with learned softmax attention over all preceding layer outputs at 48B-parameter scale; both operate across layers, while ours operates within a single attention computation.

8 Discussion

Limitations. Our experiments use small models (7–9M parameters) on two benchmarks with 10M tokens each. While the parameter-fair comparison controls for capacity and

the second benchmark confirms generalization, we have not established whether the improvement persists at the 100M–1B scale where modern attention variants are usually evaluated [23]. The cost of two attention passes per layer (approximately $2\times$ the attention computation, comparable to Twicing Attention) may matter for latency-sensitive applications, though key-value computations could be shared to reduce overhead. The Post-LN ablation uses a single seed; the effect is large (+9.6%) and directionally consistent with Proposition 4, and the parameter-fair control rules out a generic difficulty with extra parameters under Post-LN, but multi-seed confirmation would strengthen the claim.

What scaling would show, and future work. The central question is whether the 1.6% relative improvement over a parameter-matched baseline holds, grows, or shrinks at larger scales. Gradient boosting helps most when individual learners are moderately strong—too weak and the residual is too noisy to fit, too strong and a single learner already captures most of the signal—so attention in small models may sit in the regime where boosting is maximally effective; whether this persists at scale is an empirical question. Natural extensions include using gradient-boosted attention as a drop-in replacement during fine-tuning of pretrained models, which requires no large-scale pretraining; combining it with Differential Transformer for joint noise cancellation and error correction; applying boosting only at the layers that benefit most, since the gate analysis shows some layers barely use the correction while others rely on it heavily; and analyzing whether the correction round develops interpretable specialization across heads and layers.

A structural principle across scales. Since Pre-LN layers implement boosting across depth [5] and our mechanism adds boosting within each layer, both levels require the same additive residual stream—which Pre-LN preserves and Post-LN disrupts at both. The historical shift from Post-LN [21] to Pre-LN, usually attributed to training stability [22], may thus also be read as enabling the additive refinement that gradient boosting formalizes. The architecture was itself motivated by the contrast between single-pass attention and iterative convergence, reminiscent of dual-process theories in cognitive science; the formal contribution, however, is the boosting connection, which supplies precise theoretical tools rather than informal analogy.

9 Conclusion

We have introduced gradient-boosted attention, which applies gradient boosting within a single attention layer: a second attention pass, with its own learned projections, attends to the prediction error of the first and applies a gated correction. The natural alternative—iterating the same operation—erases query information orthogonal to the stored-pattern subspace and can collapse distinct queries to the same fixed point, while the correspondence to Friedman's MART framework connects a growing body of work on transformers-as-gradient-descent to the classical boosting literature. Experiments on WikiText-103 and OpenWebText confirm that the mechanism outperforms standard attention, Twicing Attention, and a parameter-matched wider baseline on both benchmarks, and the normalization ablation reveals a structural requirement: it depends on Pre-LN's additive residual stream and degrades under Post-LN, as predicted.

Disclosure of Interests. The author has no competing interests to declare that are relevant to the content of this article.

References

1. Abdullaev, L., Nguyen, T.M.: Transformer meets twicing: Harnessing unattended residual information. In: ICLR (2025)
2. Badirli, S., Liu, X., Xing, Z., Bhatt, A., Cetin, A., Singh, M.: Gradient boosting neural networks: GrowNet (2020)
3. Bai, S., Kolter, J.Z., Koltun, V.: Deep equilibrium models. In: NeurIPS (2019)
4. Banino, A., Balaguer, J., Blundell, C.: PonderNet: Learning to ponder. arXiv preprint arXiv:2106.01345 (2021)
5. Cheng, X., Chen, Y., Sra, S.: Transformers implement functional gradient descent to learn non-linear functions in context. In: ICML. pp. 8002–8037 (2024)
6. Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J., Kaiser, Ł.: Universal transformers. In: ICLR (2019)
7. Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., et al.: A mathematical framework for transformer circuits. Transformer Circuits Thread (2021)
8. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Annals of Statistics **29**(5), 1189–1232 (2001)
9. Gokaslan, A., Cohen, V.: OpenWebText corpus (2019), `http://Skylion007.github.io/OpenWebTextCorpus`, accessed: 2026-06-23
10. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: ECCV. pp. 630–645 (2016)
11. Heddes, L., et al.: DeepCrossAttention: Supercharging transformer residual connections. arXiv preprint arXiv:2502.06785 (2025)
12. Huang, F., Ash, J., Langford, J., Schapire, R.: Learning deep ResNet blocks sequentially using boosting theory. In: ICML (2018)
13. Kimi Team: Attention residuals. arXiv preprint arXiv:2603.15031 (2026)
14. Merity, S., Xiong, C., Bradbury, J., Socher, R.: Pointer sentinel mixture models. arXiv preprint arXiv:1609.07843 (2016)
15. Newey, W.K., Hsieh, F., Robins, J.M.: Twicing kernels and a small bias property of semiparametric estimators. Econometrica **72**(3), 947–962 (2004)
16. Qiu, Z., et al.: Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. arXiv preprint arXiv:2505.06708 (2025)
17. Ramsauer, H., Schäfl, B., Lehner, J., Seidl, P., Widrich, M., Adler, T., Gruber, L., Holzleitner, M., Pavlović, M., Sandve, G.K., et al.: Hopfield networks is all you need. In: ICLR (2021)
18. Siu, C.: Residual networks behave like boosting algorithms. arXiv preprint arXiv:1909.11790 (2019)
19. Smart, M., Bietti, A., Sengupta, B.: In-context denoising with one-layer transformers: Connections between attention and associative memory retrieval. In: ICML. pp. 55950–55971 (2025)
20. Tukey, J.W.: Exploratory Data Analysis. Addison-Wesley (1977)
21. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS. vol. 30 (2017)
22. Xiong, R., Yang, Y., He, J., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., Liu, T.Y.: On layer normalization in the transformer architecture. In: ICML. pp. 10524–10533 (2020)
23. Ye, T., Dong, L., Xia, Y., Sun, Y., Zhu, Y., Huang, G., Wei, F.: Differential transformer. In: ICLR (2025)