W_K, W_V is probably all you need: On the necessity of the Query, Key and Value weight triplet in the decoder-only Transformer architecture*

Marko Karbevski† and Antonij Mijoski‡[1]

[1]Institut de Recherche Mathématique Avancée (IRMA), Université de Strasbourg

October 2025

Abstract

The Query, Key, Value weight triplet is a building block of current attention mechanisms in state-of-the-art LLMs. We theoretically investigate whether this triplet can be reduced, proving under simplifying assumptions that the Query weights are redundant, thereby reducing the number of non-embedding/lm-head parameters by over 8%. We validate the theory on full-complexity GPT-3 small architectures (with layer normalization, skip connections, and weight decay) trained from scratch, demonstrating that the reduced model achieves comparable validation loss to standard baselines. These findings motivate the investigation of the Query weight redundancy at scale.

*Code & Checkpoints to be made available at: https://github.com/MarkoKarbevski/Wqkv_necessity

†marko.karbevski@gmail.com. Work initiated while at the HTEC Group [KH24].

‡antonij.mijoski@unistra.fr, antonijmijo@gmail.com

Contents

1 Introduction

Training and deploying transformer-based language models [VSP+17] remains computationally expensive [SZM+23], motivating architectural optimizations [TDBM22] including quantization [XXQ+23, MWM+24], efficient attention [CLD+20, WLK+20], weight sharing [Sha19, ALTdJ+23, LCG+20], and component simplification [HH24, Hei24, BKS+25, ZCH+25]. Recent work has shown that entire architectural components: layer normalization [Hei24, BKS+25], specific skip connections [HH24] can be removed with minimal performance impact, suggesting current architectures may be overparameterized.

We investigate redundancy within the attention mechanism itself: is the Query-Key-Value weight triplet necessary? We prove theoretically that under simplifying assumptions, the Query projection W_Q can be eliminated (i.e. replaced by $W_Q := I_d$) through basis transformations. Empirically, we validate that GPT-style models [BMR+20] trained from scratch with $W_Q = I_d$ achieve comparable validation loss to standard baselines, removing 25% of attention parameters per layer (8.3% of transformer block parameters).

1.1 Contributions

Theoretical results. We investigate query weight redundancy in settings complementary to recent theoretical work by Graef [Gra24] and in the spirit of [HH24].

- **Query elimination with skip connections (Theorems 4.1, 4.2):** For multi-layer transformers without layer normalization but with skip connections around attention blocks, we prove that W_Q can be set to the identity while appropriately modifying the remaining parameters to preserve all input-output mappings. Under weight-sharing, the result extends to transformers with skip connections around both attention and MLPs.
- **Partial results with layer normalization (Section 8.1):** We derive sufficient conditions for basis transformations to commute with layer normalization (Lemma 8.1, Theorem 8.2). These conditions are restrictive and may not hold in practice, explaining why our experiments require hyperparameter adjustments.
- **ReLU MLP Skip connection characterization (Theorem 8.3):** We characterize when residual connections can be absorbed into ReLU MLPs used in the transformers, solving the functional equation $W_2 \operatorname{ReLU}(W_1 x) + x = V_2 \operatorname{ReLU}(V_1 x)$ in V_1, V_2 for fixed W_1, W_2.

Theoretical scope. We theoretically investigate transformers with different levels of simplification around layer normalization and skip connections. Our two main results handle either skip connections only around attention, or all skip connections but with weight-sharing between blocks. While this differs from standard architectures that include both attention and MLP skip connections plus normalization, we remark that such simplifications of the model have proven fruitful in the theoretical analysis of the transformer architecture [GLPR24, IHL+24, YBR+19, LCG+20]. Moreover, recent empirical work [Hei24, BKS+25, ZCH+25] demonstrates that layer normalization can be removed at inference or replaced with simpler alternatives with minimal performance degradation, suggesting our simplified setting is realistic.

Crucially, our theoretical results provide sufficient conditions for query weight elimination, not necessary ones. This distinction motivates our empirical investigation: even when the architectural simplifications underlying our theorems are violated, query weight redundancy may persist. The experiments test precisely this possibility.

Empirical validation. We train GPT-3-Small-scale models (124M-163M parameters) from scratch on OpenWebText. Models with $W_Q = I_d$ achieve validation loss within $< 1\%$ of baselines when adjusting attention scaling ($1/\sqrt{d_k} \to 1/(2\sqrt{d_k})$) and weight decay ($0.1 \to 2^{-5}$).

Beyond full pretraining, we test whether gradient descent can discover approximate solutions to the basis transformation problem in the MLP+skip layer. Using randomly initialized Gaussian target networks—a deliberately general setting—trained GELU MLPs achieve 4-5% relative error across dimensions 256-768, substantially outperforming optimal linear baselines at 9-10%. This indicates that such compositions can be approximated up to a certain difference.

We do not test at scales beyond the 163M parameter models, leaving generalization to larger models and architectures as future work.

Limitations. A gap remains between theory and practice. We derive sufficient conditions for basis transformations to commute with layer normalization (Section 8.1), but these conditions are restrictive and may not hold during training.

Our experiments have additional limitations: we train only at modest scale (< 200M parameters), lack ablations isolating architectural effects from hyperparameter effects, and do not report confidence intervals across multiple random seeds. These constraints limit statistical confidence and leave generalization to billion-parameter models as important future work. Nonetheless, the consistency between our theoretical predictions and empirical observations, that models with $W_Q = I_d$ can match baseline performance when appropriately tuned provides evidence for Query weight redundancy even in realistic architectures with layer normalization.

Organization. Section 2 positions our work relative to recent simplification efforts. Section 3 establishes notation and the Reparametrization Lemma. Section 4 presents theoretical analysis under progressively relaxed assumptions, with Table 1.1 summarizing which architectural features each result handles. Section 5 provides empirical validation. Section 6 discusses limitations and future directions.

Result Type	Section	Normalization	Skip Connections	Multihead	Multi-layer	Weight Decay
Theoretical	4.1	✓	✓	×	✓	✓ post-training × during training
	4.2	×	Around Attention only	✓	✓	×
	4.3	×	✓	✓	Shared weights only	×
	8.1	✓	*Id*	*Id*	*Id*	×
Experimental	5.1	*Id*	✓	*Id*	*Id*	×
	5.2	✓	✓	✓	✓	✓

Table 1: Coverage of transformer components in theoretical analysis versus experimental validation. ✓ stands for a feature that's covered in the section, × means it's being ignored (i.e. a simplification is made w.r.t. that feature) and *Id* signifies results that are independent of a given feature

2 Related Work

Architectural simplification in prior work. Recent research challenges the necessity of standard transformer components. Graef [Gra24] formally proves that in skipless transformers without normalization, both W_Q and W_O can be eliminated simultaneously, but does not provide empirical validation. By contrast, He & Hofmann [HH24] empirically study simplified parallel attention & MLP blocks (building on Zhao et al. [ZXZ+19]), focusing on training speed and signal propagation in controlled settings such as CodeParrot and masked language modeling with BERT. Our work differs from both: we analyze decoder-only transformers with skip connections, where we eliminate only W_Q while retaining W_O, since skip connections require propagating basis transformations through residual pathways and W_O provides this mechanism. Beyond theoretical analysis, we validate our results by training GPT-style autoregressive models from scratch on OpenWebText, demonstrating that Query weight elimination transfers to practical large-scale language modeling with layer normalization and full architectural complexity.

Pan et al. [PPXS24] analyze the query-key interaction in Vision Transformers by decomposing $W_Q^\top W_K$ via SVD, revealing that singular vector pairs correspond to semantically interpretable attention patterns. Their analysis focuses on encoder-only Vision Transformers where bidirectional attention enables rich contextual interactions. Our work exploits the same mathematical property: that attention depends on $W_Q^\top W_K$ rather than W_Q separately but for a different purpose and architecture: we eliminate W_Q entirely in decoder-only language models by setting it to identity and propagating basis transformations through causal attention layers with skip connections.

Moreover, we provide the first combined theoretical and empirical study of Query weight elimination. We train models from scratch to validate our theoretical predictions, demonstrating that models with $W_Q = I_d$ achieve comparable performance to baselines when appropriately tuned.

We note that analyzing simplified transformer variants: without normalization, with restricted skip connections, or in weight-shared regimes, is standard practice in theoretical transformer analysis [GLPR24, IHL+24, YBR+19, LCG+20]. Such simplifications have proven fruitful for understanding neural architectures. Our empirical validation bridges theory and practice: we demonstrate that the insights extend to realistic architectures with layer normalization, achieving comparable performance with appropriate hyperparameter adjustments.

Recent work on removing layer normalization at inference [Hei24, BKS+25] or replacing it with simpler alternatives [ZCH+25] further supports the relevance of our simplified theoretical setting.

Efficient attention mechanisms. Numerous works reduce attention complexity through approximation [CLD+20, WLK+20] or sparsity. Grouped-Query Attention [ALTdJ+23] and Multi-Query Attention [Sha19] reduce parameters by sharing Key-Value projections across Query heads. Our approach is orthogonal and combinable: Query elimination applies to standard, GQA, and MQA architectures

alike, enabling multiplicative savings. While GQA's practical gains at scale are well-established, our 200M-parameter experiments provide initial evidence for Query redundancy, with larger-scale validation as important future work.

Weight sharing and tying. ALBERT [LCG+20] shares all parameters across layers, achieving 18× reduction with modest performance cost, demonstrating that layer-specific parameters may be inessential for many tasks. Press and Wolf [PW17] tie embedding and language model head (output projection) weights, exploiting the observation that both map between vocabulary and embedding spaces. This technique, adopted in GPT-2 [RWC+19] and many subsequent models, reduces parameters while often improving performance through regularization. Our work identifies finer-grained redundancy: even within a single attention layer, the Query weight matrix is redundant given Key and Value projections, suggesting opportunities for parameter reduction beyond layer-level or embedding-level sharing.

Theoretical foundations. Our analysis adopts methodological approaches similar to those used in theoretical studies of transformers. Yun et al. [YBR+19] establish universal approximation properties for sequence-to-sequence functions, while Ildiz et al. [IHL+24] reveal connections between self-attention and Markov dynamics in generative transformers. Geshkovski et al. [GLPR23, GLPR24] provide mean-field analyses revealing clustering phenomena in self-attention dynamics. Like these works, we analyze simplified architectural variants to obtain tractable results—a standard methodology in theoretical machine learning that has proven fruitful for understanding neural architectures. Our work focuses on a complementary question: parameter redundancy in the attention mechanism through basis transformations.

Parameter-efficient fine-tuning and compression. LoRA [HSW+21] and related methods [HGJ+19, DPHZ24] achieve efficiency through low-rank adaptation during fine-tuning. Post-training compression via pruning [MFW23] and quantization [XXQ+23, MWM+24] are also orthogonal to our approach. Query elimination applies at the architectural level, benefiting both pre-training and inference, and can be combined with these techniques for compound efficiency gains.

3 Preliminaries

We establish notation for multi-head attention and state the Reparametrization Lemma underlying our theoretical analysis. We introduce block-wise Hadamard products for cleaner expressions and notational convenience.

3.1 Notation for the multi-head attention

To cleanly express our results, we introduce block-matrix notation that makes the per-head structure of the multi-head attention explicit.

Block decomposition. Fix n (sequence length), d_{model} (embedding dimension), and h (number of attention heads) with $h|d_{\text{model}}$. Let $d_k = d_{\text{model}}/h$ denote the dimension per head. We decompose matrices into head-wise blocks:

- For $W, U \in \mathbb{R}^{n \times d_{\text{model}}}$, write $W = (W_1|\cdots|W_h)$ and $U = (U_1|\cdots|U_h)$ where each $W_i, U_i \in \mathbb{R}^{n \times d_k}$.
- For $V \in \mathbb{R}^{d_{\text{model}} \times n}$, write $V = \begin{pmatrix} V_1 \\ \vdots \\ V_h \end{pmatrix}$ where each $V_i \in \mathbb{R}^{d_k \times n}$.

We recall that, in this notation, the standard matrix multiplication of W and V can be written as $WV = \sum_{i=1}^{h} W_i V_i$.

Block-wise operations. To express per-head operations, we define the *block-wise Hadamard product*:

$$W \boxdot V = (W_1 V_1 | \cdots | W_h V_h) \in \mathbb{R}^{n \times (hn)},$$

which multiplies corresponding head blocks and concatenates results. The transposed variant is

$$W \boxdot^t U = (W_1 U_1 | \cdots | W_h U_h) \in \mathbb{R}^{n \times (hn)}.$$

These are variants of Hadamard block products introduced in [HMN91].

The *causal block softmax* applies softmax and causal masking independently to each head:

$$\text{CausalSoftmax}^{\boxdot}(A_1 | \cdots | A_h) = \big(\text{softmax}(M + A_1) \,|\, \cdots \,|\, \text{softmax}(M + A_h)\big),$$

where $A_i \in \mathbb{R}^{n \times n}$ are attention logits and $M \in \mathbb{R}^{n \times n}$ is the causal mask with $M_{ij} = 0$ if $i \geq j$, otherwise $M_{ij} = -\infty$.

Multi-head attention in block notation. Multi-head causal self-attention is:

$$\mathcal{S}_c(X, W_Q, W_K, W_V) = \text{CausalSoftmax}^{\boxdot}\left(\frac{1}{\sqrt{d_k}}(XW_Q) \boxdot (XW_K)^T\right) \boxdot^t (XW_V),$$
$$\text{Attn}(X, W_Q, W_K, W_V, W_O) = \mathcal{S}_c(X, W_Q, W_K, W_V) \cdot W_O,$$

where $X \in \mathbb{R}^{n \times d_{\text{model}}}$ and $W_Q, W_K, W_V, W_O \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$. This notation makes explicit that attention scores are computed independently per head, while W_O mixes information across heads.

3.2 The Reparametrization Lemma

A key observation enabling our results is that the attention mechanism depends on X *only through* the projections XW_Q, XW_K, and XW_V. Formally, we can write

$$\mathcal{S}_c(X, W_Q, W_K, W_V) = g(XW_Q, XW_K, XW_V)$$

for some function g. This structural property allows us to absorb the Query projection through a change of basis:

Lemma 3.1 (Reparametrization Lemma)**.** *Let $n, d \in \mathbb{N}$ and let Ω be any set. Consider a function*

$$f : Mat(n, d) \times Mat(d, d)^3 \to \Omega$$

that depends on its first argument only through XW_Q, XW_K, XW_V. That is, suppose there exists a function $g : Mat(n, d)^3 \to \Omega$ such that

$$f(X, W_Q, W_K, W_V) = g(XW_Q, XW_K, XW_V) \quad \textit{for all } X \in Mat(n, d).$$

Then for any invertible $W_Q \in GL(d)$ and any $W_K, W_V \in Mat(d, d)$, there exist matrices $\Theta, \widetilde{W}_K, \widetilde{W}_V \in Mat(d, d)$ such that

$$f(X, W_Q, W_K, W_V) = f(X\Theta, I_d, \widetilde{W}_K, \widetilde{W}_V) \quad \textit{for all } X \in Mat(n, d).$$

Moreover, the choice $\Theta = W_Q$, $\widetilde{W}_K = W_Q^{-1} W_K$, $\widetilde{W}_V = W_Q^{-1} W_V$ satisfies this property.

Proof. Let $\Theta = W_Q$, $\widetilde{W}_K = W_Q^{-1} W_K$, and $\widetilde{W}_V = W_Q^{-1} W_V$. Then for any $X \in \text{Mat}(n, d)$:

$$\begin{aligned} f(X\Theta, I_d, \widetilde{W}_K, \widetilde{W}_V) &= g(X\Theta \cdot I_d,\ X\Theta \cdot \widetilde{W}_K,\ X\Theta \cdot \widetilde{W}_V) \\ &= g(XW_Q,\ XW_Q \cdot W_Q^{-1} W_K,\ XW_Q \cdot W_Q^{-1} W_V) \\ &= g(XW_Q,\ XW_K,\ XW_V) \\ &= f(X, W_Q, W_K, W_V). \end{aligned}$$

□

4 Theoretical analysis of the parameter reduction

In the previous notation, we have that

$$\begin{aligned}\mathrm{Block}(X) &= X + \mathrm{Att} \circ \mathrm{Ln}_1(X) + \mathrm{MLP} \circ \mathrm{Ln}_2(X + \mathrm{Att} \circ \mathrm{Ln}_1(X)) &(1)\\ &= (\mathrm{Id} + \mathrm{MLP} \circ \mathrm{Ln}_2) \circ (\mathrm{Id} + \mathrm{Att} \circ \mathrm{Ln}_1)(X) &(2)\end{aligned}$$

Let us now provide the theoretical investigation of the proposed improvement under different assumptions. The results are ordered by the subjective difficulty of deriving them. We remind the reader that such simplifications of the model have proven fruitful in the theoretical analysis of the transformer architecture [GLPR23, IHL+24, YBR+19, LCG+20].

4.1 On some degrees of freedom in the Single Head and Multi-Head Attention

4.1.1 The Single Head case

We begin with single-head attention, the conceptually simplest case. Remarkably, this setting admits a clean result that holds regardless of normalization or skip connections: attention scores depend on queries and keys only through their product $W_Q W_K^T$, suggesting that separate parameterization via (W_Q, W_K) may be redundant. Unlike the multi-head case analyzed in subsequent sections, the single-head reparametrization is local to the attention mechanism and does not require propagating basis transformations through other architectural components.

In the single head case, the attention mechanism reads:

$$\mathrm{Att}(X, W_Q, W_K, W_V, W_O) = \mathrm{softmax}\left(\frac{XW_Q W_K^T X^T}{c}\right) XW_V W_O$$

where $X \in \mathbb{R}^{n \times d}$ is the input sequence, and $W_Q, W_K, W_V, W_O \in \mathbb{R}^{d \times d}$.

The following proposition confirms this intuition and serves as a warm-up for the multi-head and multi-layer cases analyzed in subsequent sections.

Proposition 4.1. *Under the assumption of a single-head attention in a given block, the Query-Key weight pair $(W_Q, W_K) \in Mat(d,d) \times Mat(d,d)$ can be replaced by a single matrix $W \in Mat(d,d)$ without loss of information. More generally, for all X, W_Q, W_V, W_O of appropriate shape we have:*

$$Att(X, W_Q, W_K, W_V, W_O) = Att(X, Id, W_K W_Q^t, W_V, W_O)$$

Proof. Define $W = W_Q W_K^T$. Then:

$$XW_Q W_K^T X^T = XWX^T$$

Since the attention computation depends only on the product $W_Q W_K^T$, we can directly parametrize this product as a single matrix W. This preserves the exact same attention scores for any input sequence X. □

Remark 1. *We remark that in the single-head regime, the matrix W_O is also redundant.*

Weight decay during training may prevent the optimizer from naturally discovering this reduced form, as it penalizes $\|W_Q\|^2 + \|W_K\|^2$ rather than $\|W_Q W_K^T\|^2$. That said, an already-trained such model can have its Query and Key weights merged post-training.

The result extends naturally in the multihead setting, though in a weaker variant, and with that, we will begin our

4.1.2 Extension to the Multi Head case

Proposition 4.2. *Let $D = diag(D_1, \cdots, D_h) \in GL(d_{model})$ be any block-diagonal invertible matrix with blocks $D_i \in Mat(d_k, d_k)$. Then:*

$$\mathcal{S}_c(X, W_Q, W_K, W_V) = \mathcal{S}_c\left(X, W_Q D, W_K \left(D^t\right)^{-1}, W_V\right).$$

The result does imply under-utilized degrees of freedom in the (W_Q, W_K) pair, hence our choice of elimination of the Query weights rather than the Value weights. There might be similar degrees of freedom in the case of W_V when paired with W_O, but we leave such explorations to future work.

4.2 Multihead Attention with Skip Connection around the Attention & without Normalization

Having established Query weight redundancy for single-head attention, we now address the practically relevant case: multi-head attention with skip connections. This setting introduces a fundamental challenge: basis transformations can no longer be absorbed locally within a single attention layer, but must propagate through the network's residual pathways.

We focus on the skip connection around the attention layer, treating this as the primary residual pathway while assuming the MLP operates without its own skip connection.

This architectural choice is partially supported by complementary evidence across domains. In vision transformers, removing attention skip connections causes catastrophic training failure, while removing MLP skip connections yields only modest degradation [JSML25]. For language models, [HH24] show that parallel attention-MLP architectures sharing a single skip connection match standard sequential designs, suggesting architectural flexibility though not establishing clear hierarchy of importance.

We temporarily set aside layer normalization. This simplification is supported by evidence that normalization can be removed at inference with minimal degradation [BKS+25, Hei24], and that deep transformers can be trained without normalization when using signal-propagation-preserving attention modifications [HMZ+23]. As shown in Section 8.1, incorporating normalization introduces technical complications obscuring the core mechanism. We therefore establish our result in this clearer setting first, then investigate conditions for extension through normalization layers.

With these simplifications, we state our main result:

Theorem 4.1 (Attention-Skip-Only Query Weight Elimination)**.** *Let E, E_P, W_{LM} be the embedding, positional embedding and Language Modelling Head weights respectively, and let $W^i_{MLP_u}, W^i_{MLP_d}$ and $W^i_Q, W^i_K, W^i_V, W^i_O$ be the MLP and attention weights in the i-th layer for $0 < i \leq L$ in the L-layer decoder-only transformer without normalization and with a skip-connection only around the attention:*

$$T_{E,E_P,W_{LM},(W^i_{MLP_u},W^i_{MLP_d},W^i_Q,W^i_K,W^i_V,W^i_O)_{0<i\leq L}}$$

Then there exist modified weights $\widetilde{E}, \widetilde{E}_P, \widetilde{W}_{LM}$ and $(\widetilde{W}^i_{MLP_u}, \widetilde{W}^i_{MLP_d}, \widetilde{W}^i_K, \widetilde{W}^i_V, \widetilde{W}^i_O)_{0<i\leq L}$ such that for all token sequences $x = (x_0, \ldots, x_k)$ of permissible length:

$$T_{E,E_P,(W^i_{MLP_u},W^i_{MLP_d},W^i_Q,W^i_K,W^i_V,W^i_O)_{0<i\leq L}}(x) = T_{\widetilde{E},\widetilde{E}_P,\widetilde{W}_{LM},(\widetilde{W}^i_{MLP_u},\widetilde{W}^i_{MLP_d},Id,\widetilde{W}^i_K,\widetilde{W}^i_V,\widetilde{W}^i_O)_{0<i\leq L}}(x).$$

4.3 Weight-Shared Multi-Layer Transformers with skip connections and no normalization

Weight sharing across transformer layers appears in both practical and theoretical contexts. Practically, ALBERT [LCG+20] employs full parameter sharing to achieve up to 18× parameter reduction compared to BERT while maintaining competitive performance, demonstrating that layer-specific parameters may be inessential for many tasks. Theoretically, weight sharing enables multiple analytical frameworks: deep equilibrium models [BKK19] study transformers as fixed-point solvers of a single repeated block, while mean-field approaches [GLPR24] analyze the continuous-time limit as $L \to \infty$, revealing clustering phenomena and long-time behavior of token representations in such a case.

Theorem 4.2 provides structural evidence that, in weight-shared architectures, the query matrix can be eliminated without loss of generality. While established here in a simplified setting, this canonical form offers analytical indications for simplifying fixed-point and continuous-time limits analysis.

Having established query elimination for a single layer, we now consider the weight-shared regime where all layers use identical parameters.

Theorem 4.2 (Weight-Shared Query Elimination)**.** *Consider an L-layer transformer without layer normalization where all layers share the same block parameters with $W_Q \in GL(d)$. Then the complete transformer is functionally equivalent to a weight-shared transformer in which the query weight matrix is replaced by Id.*

Remark 2. *We note that in the case of tied LM head and embedding weights ($W_{LM} = E^T$), the single-layer reduced model must untie these weights to maintain functional equivalence. Specifically,*

the reduced model requires $\widetilde{W}_{LM} = W_Q^{-1}E^T$ *and* $\widetilde{E} = EW_Q$*, which satisfy* $\widetilde{W}_{LM} \neq \widetilde{E}^T$ *unless* W_Q *is orthogonal.*

We validate our theoretical results by training decoder-only language models from scratch with $W_Q = I_d$ and comparing them to standard baselines. Our experiments confirm that Query weights are indeed redundant: models trained without them achieve comparable performance to standard models while using fewer parameters.

5 Experiments

5.1 On the approximation of the basis transfer around the MLP with skip connection

Our theoretical analysis establishes Query weight elimination for architectures with skip connections around attention layers (Theorem 4.1) and for weight-shared architectures (Theorem 4.2). However, standard transformers include skip connections around both attention and MLP blocks. The basis transformation approach requires transforming coordinates at layer boundaries, which in the presence of skip connections around MLPs leads to the functional equation:

$$\Omega_1 \circ (\mathrm{MLP} + \mathrm{Id}) \circ \Omega_2^{-1} \approx \mathrm{MLP}' + \mathrm{Id} \tag{3}$$

where Ω_i are the basis change matrices. We refer the reader to Theorem 8.3 for the case of exact solutions in the case of the ReLU activation.

Experimental setup. To empirically investigate whether standard gradient-based optimization can discover good approximate solutions to this composition problem, we conduct a controlled experiment. We generate synthetic target functions that mirror the structure arising from basis transformations through skip-connected blocks:

$$Y_{\text{target}}(X) = W_2 \cdot \mathrm{GELU}(W_1 \cdot X) + Z \cdot X - X \tag{4}$$

where $W_1 \in \mathbb{R}^{4h \times h}$, $W_2 \in \mathbb{R}^{h \times 4h}$, and $Z \in \mathbb{R}^{h \times h}$ are randomly initialized and normalized by $1/\sqrt{4h}$, $1/\sqrt{4h}$, and $1/\sqrt{h}$ respectively.

We train a two-layer GELU MLP of the same shape, with a skip connection to approximate this target:

$$Y_{\text{model}}(X) = W_2' \cdot \mathrm{GELU}(W_1' \cdot X) + X \tag{5}$$

minimizing the L2 relative squared error objective. We train for 20,000 steps with batch size 65,536 using AdamW (learning rate 5×10^{-3} with cosine decay, weight decay 10^{-1}, gradient clipping at norm 3.0). We test across embedding dimensions $h \in \{256, 512, 768\}$ and compare against an optimal linear baseline matrix A^* fitted via streaming ridge regression (regularization 10^{-6}) on 500,000 samples.

Results. Figure 5.1 shows the distribution of per-sample relative L2 errors on 32,768 held-out test samples. The trained GELU MLP achieves mean relative errors of 4.0-5.0% across all dimensions tested, substantially outperforming the linear baseline (9.0-10.0%).

Figure 2 provides complementary evidence through directional alignment: the trained MLP achieves mean cosine similarities around 0.999 between predicted and target outputs, while the linear baseline achieves ≈ 0.995.

Implications for Query weight elimination. These results demonstrate that:

1. **Approximate solutions are learnable**: While our theory characterizes exact solutions only, standard gradient descent reliably discovers high-quality approximate solutions to the composition problem $\Omega_1 \circ (\mathrm{MLP} + \mathrm{Id}) \circ \Omega_2^{-1} \approx \mathrm{MLP}' + \mathrm{Id}$ with GELU activations.

2. **Nonlinearity is essential**: The trained nonlinear MLP outperforms the optimal linear baseline by a factor of $2\times$ in relative error, confirming that the composition of basis changes with skip connections cannot be adequately approximated by linear transformations alone.

3. **Dimension-independent quality**: Consistent performance across $h \in \{256, 512, 768\}$ suggests the approximation quality does not degrade with dimension, supporting viability at larger scales.

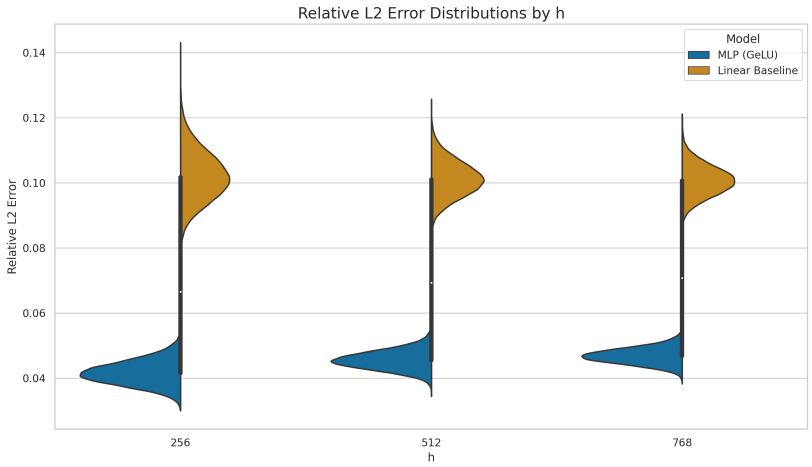

Figure 1: Per-sample relative L2 error distributions for approximating basis-transformed skip-connected MLPs. The trained GELU MLP (blue) achieves 4–5% relative error across dimensions $h \in \{256, 512, 768\}$, substantially outperforming the optimal linear baseline (orange, 9–10%).

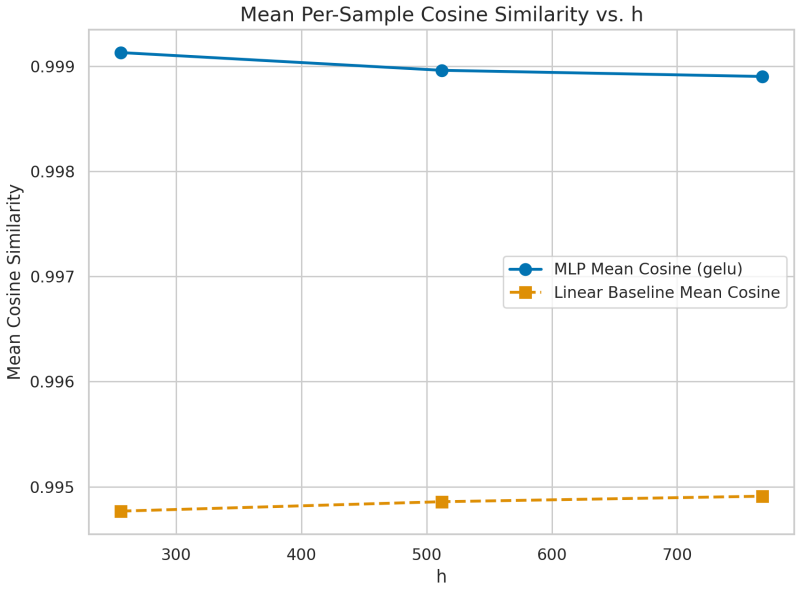

Figure 2: Mean per-sample cosine similarity between predicted and target outputs. The trained MLP achieves 0.999 cosine similarity ($\approx 2.5°$ angular error) while the linear baseline achieves ≈ 0.995 ($\approx 5.7°$ angular error), demonstrating near-perfect directional alignment.

These findings complement our theoretical results: while Theorem 4.1 guarantees exact Query elimination when skip connections are restricted to attention layers, this experiment provides empirical evidence that Query elimination remains practical when skip connections are present throughout the architecture, as in standard transformers. The key insight is that MLPs can implicitly learn the necessary basis adaptations through gradient-based optimization, even when those adaptations do not admit closed-form solutions. This bridges the gap between our theoretical guarantees (exact solutions, ReLU, restricted skip connections) and practical architectures (approximate solutions, GELU, full skip connections).

5.2 Full pretraining of GPT style models

5.2.1 Architecture and Training Configuration

Model architecture. We follow the GPT-2 and GPT-3-small architecture [YBR+19, BMR+20] with 12 layers, 12 attention heads, embedding dimension $d_{\text{model}} = 768$ (head dimension $d_k = 64$), and MLP hidden dimension $4d_{\text{model}} = 3072$ with GELU activations [HG16]. We use layer normalization [BKH16], a context length of 1024 tokens, and the GPT-2 BPE tokenizer (vocabulary size 50,257). We remove bias terms and the dropout, but retain all other architectural components. Our implementation builds on NanoGPT [Kar], modified to support $W_Q = I_d$ in the reduced variant while maintaining identical architecture otherwise.

Training. We train on OpenWebText [GC19], an open-source reproduction of GPT-2's WebText dataset containing approximately 8 million documents. Optimization uses AdamW [LH19] with $\beta_1 = 0.9$, $\beta_2 = 0.95$, learning rate warmup, cosine decay schedule, and gradient clipping at norm 1.0 following standard practice [RWC+19]. All models are implemented in PyTorch 2.7 with CUDA 12.8 and trained on a single NVIDIA RTX 5090 GPU. Both baseline and reduced models use PyTorch's scaled dot-product attention with Flash Attention [DFE+22, Dao24] optimizations, ensuring fair comparison of architectural modifications alone. We use the HuggingFace Datasets library [L+21] for data loading and track experiments with Weights & Biases [Bie20]. Code and trained checkpoints will be made available upon publication.

5.2.2 Model Variants and Practical Adjustments

We compare two primary configurations, each with two weight-tying variants:

Standard baseline: Full (W_Q, W_K, W_V, W_O) attention with the standard weight decay equal to 0.1.

Reduced ($W_Q = I_d$): Modified (I_d, W_K, W_V, W_O) attention with adjusted weight decay to 2^{-5}.

For each configuration, we test both tied weights (embedding matrix shared with LM head, as in GPT-2 [RWC+19]) and untied weights (independent embedding and LM head matrices).

Practical adjustments. While Theorems 4.1 and 4.2 establish formal equivalence in simplified models, practical implementation requires two adjustments to accommodate the interaction between our basis transformation and standard training procedures.

- **Attention scaling:** We scale attention logits by $\frac{1}{2\sqrt{d_k}}$ instead of the standard $\frac{1}{\sqrt{d_k}}$. This compensates for the different magnitude of pre-softmax activations when queries are computed as $Q = X$ versus $Q = XW_Q$ in the original parameterization, as well as the smaller weight decay applied to W_K.
- **Weight decay:** We reduce weight decay from 0.1 to $2^{-5} = 0.03125$ for all parameters in the reduced model. This adjustment accounts for two factors: (i) the basis transformation involves matrix inverses, which can have substantially larger entries than the original weights, making the transformed parameters more sensitive to decay; (ii) as shown in Theorem 8.2, the MLP may now need to approximate a composition involving the basis change, effectively requiring the network to learn a more complex function with the same capacity.

These modifications bring forward a fundamental trade-off between the architectural and the explicit regularization. The elimination of W_Q constitutes a form of architectural regularization: by reducing parameters from $(W_Q, W_K) \in \mathbb{R}^{d\times d_k} \times \mathbb{R}^{d\times d_k}$ to $W_K \in \mathbb{R}^{d\times d_k}$, we constrain the model's capacity and reduce its propensity to overfit. However, to preserve the expressivity necessary to solve the task, this architectural constraint necessitates a corresponding reduction in explicit L_2 regularization.

Maintaining the original weight decay of 0.1 in the reduced model would constitute over-regularization: the combination of fewer parameters and strong weight decay would excessively constrain the optimization, preventing the model from achieving the capacity required for the task. The reduction to decay $2^{-5} \approx 0.03$ is thus not merely a hyperparameter adjustment but a necessary rebalancing to maintain effective model capacity.

This finding connects to fundamental principles of statistical learning: the optimal amount of regularization depends on model complexity [Bar98, NBMS17]. Our result makes this principle explicit for transformer architectures: architectural choices and explicit regularization must be jointly

calibrated, and eliminating redundant parameters requires recalibrating decay rates to preserve the regularization-capacity balance.

5.2.3 Results

Table 2 summarizes parameter counts and validation performance. Figure 3 shows training dynamics for tied weights, and Figure 4 for untied weights.

	Standard (tied)	Reduced (tied, $W_Q = \text{Id}$)	Standard (untied)	Reduced (untied, $W_Q = \text{Id}$)
Trainable W_Q	✓	×	✓	×
Trainable W_O	✓	✓	✓	✓
Total #Params	124M	**117M**	163M	**156M**
Attn+MLP #Params	86M	**79M**	86M	**79M**
#Layers	12	12	12	12
#Heads	12	12	12	12
Embedding Dim	12×64	12×64	12×64	12×64
Scale: Attention	$\frac{1}{\sqrt{64}}$	$\frac{1}{2\sqrt{64}}$	$\frac{1}{\sqrt{64}}$	$\frac{1}{2\sqrt{64}}$
WD: Attention	10^{-1}	2^{-5}	10^{-1}	2^{-5}
WD: MLP	10^{-1}	2^{-5}	10^{-1}	2^{-5}
WD: Embedding	10^{-1}	2^{-5}	10^{-1}	2^{-5}
Val Loss	2.9	**2.88**	2.87	**2.86**

Table 2: Parameter counts and validation performance for standard and reduced models. Both tied and untied weight configurations are tested. The reduced models achieve comparable or better validation loss despite having fewer parameters.

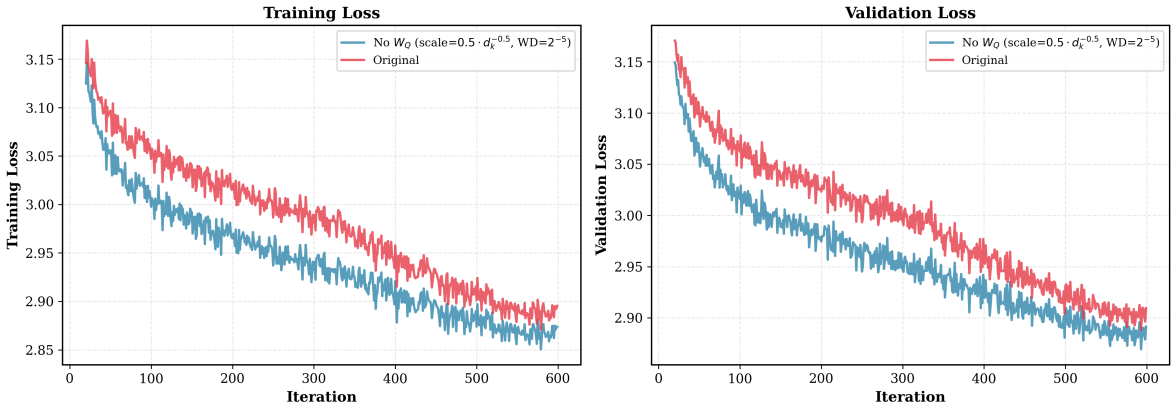

Figure 3: Training and validation loss for **tied Embedding/LMHead weights configuration**. The reduced model (No W_Q, red) closely tracks the standard baseline (blue) throughout training, achieving comparable final performance with fewer parameters.

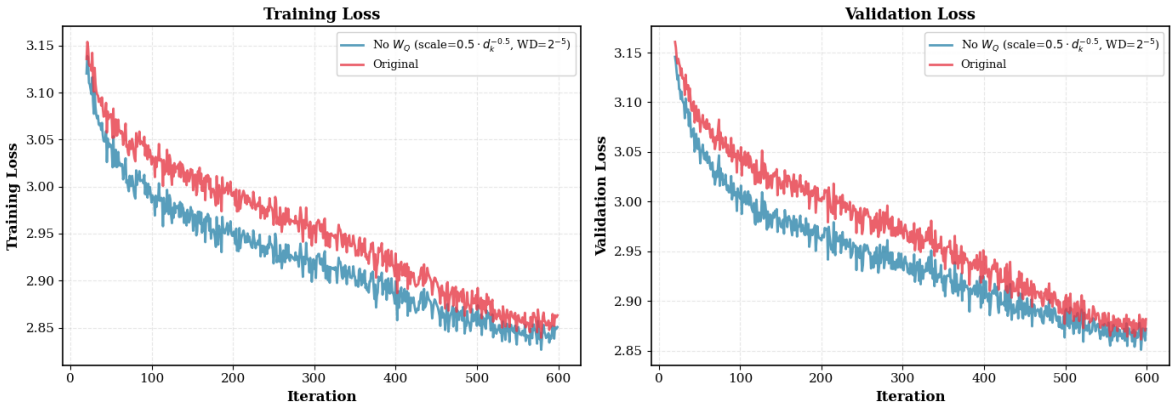

Figure 4: Training and validation loss for **untied weights** configuration. Both models converge smoothly, with the reduced variant (No W_Q, blue) achieving slightly better final validation loss than the standard model (red).

Our main findings are the following:

1. Query weights are redundant. Models trained with $W_Q = I_d$ achieve validation loss competitive with or better than standard baselines (Table 2). The tied-weights variant reaches 2.88 versus a 2.90 baseline; the untied variant reaches 2.86 versus a 2.87 baseline. This confirms our theoretical prediction that W_Q can be eliminated without fundamental loss of model capacity.

2. Training dynamics are stable. Figures 3 and 4 show that training curves for reduced models closely track standard models throughout training. Both training and validation losses converge smoothly without instabilities. The slight performance improvement in reduced models may be due to the modified weight decay acting as a beneficial regularization, though more extensive experiments across random seeds would be needed to confirm statistical significance.

3. Untied weights perform better. Both standard and reduced models achieve lower validation loss with untied weights (2.88 vs 2.9 for standard; 2.86 vs 2.87 for reduced). This is consistent with recent trends in large language models [LT24, GZA+23] which typically do not tie embedding and LM head weights.

Parameter efficiency. The reduced architecture removes 7M parameters in both tied and untied configurations. When measured against only attention and MLP parameters (excluding embeddings and LM head), the reduction is 7M out of 86M, representing an $1/12 \approx 8.3\%$ decrease in transformer block parameters. This reduction applies to most modern decoder-only transformer architecture and can be combined multiplicatively with other compression techniques.

6 Discussion

Our theoretical analysis identifies redundancy in the Query-Key-Value weight structure. While any of the three matrices can theoretically be eliminated under our assumptions, we focus experimentally on W_Q due to its favorable propagation properties through W_O (Section 4.1).

The reduction mechanism relies solely on linear projection structure and therefore extends broadly. For Rotary Position Embeddings [SLP+21], the fixed rotation commutes with our basis transformation. For Grouped-Query Attention [ALTdJ+23, Sha19], transformations act independently per Query head group. For Mixture-of-Experts [JJNH91, SMM+17, LLX+21, FZS21, DHD+22, Mis23, OAA+25, Tea25], elimination applies per-expert since gating operates outside attention. The analysis extends to encoder-only architectures with potentially larger relative savings due to smaller output projections.

Our framework (Sections 3.2 and 4) and MLP composition results (Section 5.1) may explain the effectiveness of recent architectural simplifications by He and Hofmann [HH24] to the work of Zhao et al. [ZXZ+19] in the parallel attention variant. This connection merits formal investigation as future work.

Beyond Query weights, recent work has removed layer normalization [Hei24, BKS+25, ZCH+25], eliminated skip connections [HH24, HMZ+23], and reduced head counts [ALTdJ+23]. Our 25% per-

layer attention parameter reduction (8.3% of transformer block parameters) adds evidence that standard architectures contain overparameterization.

Limitations. Our experiments are limited to models under 200M parameters. Validation at billion-parameter scale is necessary to establish whether efficiency gains scale favorably, particularly given concentration phenomena in high dimensions [Ver18]. The full theory-practice gap regarding layer normalization remains unresolved. Our sufficient conditions (Section 8.1) for basis transformations to commute with normalization are restrictive and likely violated during training, explaining the required hyperparameter adjustments (attention scaling, weight decay). Finally, multiple random seeds are needed to establish statistical significance of the architectural modification.

7 Conclusion

We prove that the Query weight matrix W_Q in multi-head attention can be eliminated through basis transformations under simplified architectural assumptions, reducing attention parameters by 25% per layer. Empirical validation at 100-200M parameter scale confirms that models trained with $W_Q = \mathrm{Id}$ match baseline validation loss when hyperparameters are appropriately adjusted, providing evidence that the redundancy persists in realistic settings despite theoretical gaps.

The results extend naturally to Grouped-Query Attention, Mixture-of-Experts, and RoPE (except Section 4.1), suggesting the redundancy is architectural rather than configuration-specific. More broadly, the work demonstrates that transformers can achieve strong performance despite identifiable overparameterization, raising fundamental questions about which architectural components are necessary for expressivity versus artifacts of design history. Formalizing the connection between our Query elimination framework and other recent simplifications [HH24] represents important future work toward a unified theory of transformer redundancy.

8 Appendix

8.1 LayerNorm & Change of Basis via the (skip +) MLP

We recall that recent empirical studies [Hei24, BKS+25] suggest that Layer Normalization can often be omitted at least at inference time with little to no degradation in performance. While such evidence points to its potential disposability, we also investigate from the complementary perspective of asking what structural constraints arise if LayerNorm is in fact kept as-is. In particular, we investigate how the presence of LayerNorm interacts with surrounding linear and nonlinear components, and what reparameterizations would be required to preserve the change of basis.

Concretely, we ask when there exist a modified multilayer perceptron MLP' and a diagonal matrix D' such that

$$D'\, L_\varepsilon\big(x + MLP'(x)\big) \;\approx\; \Theta D\, L_\varepsilon\big(x + MLP(x)\big), \tag{6}$$

where L_ε denotes the ε-regularized LayerNorm operator, D is the original learned diagonal scaling associated with the normalization, and W_Q is the subsequent linear projection. In order to provide the sufficient condition, we will rely on the following Lemma:

Lemma 8.1. *Let $\varepsilon > 0$ and define $L_\varepsilon : \mathbb{R}^d \to \mathbb{R}^d$ by*

$$L_\varepsilon(x) \;=\; \frac{x - \mu(x)\mathbf{1}}{\sigma_\varepsilon(x)}, \tag{7}$$

where $\mu(x) = \frac{1}{d}\sum_{i=1}^d x_i$ and $\sigma_\varepsilon(x) = \sqrt{\frac{1}{d}\sum_{i=1}^d (x_i - \mu(x))^2 + \varepsilon}$. Let $A \in GL(d)$ be fixed. Then there exists a function $f : \mathbb{R}^d \to \mathbb{R}^d$ and a diagonal matrix $D \in \mathbb{R}^{d\times d}$ such that

$$L_\varepsilon(f(x)) \;=\; (DA)\, L_\varepsilon(x), \quad \forall x \in \mathbb{R}^d. \tag{8}$$

Moreover, for any real function (i.e. scalar field) $h : \mathbb{R}^n \to \mathbb{R}$, $f + h\mathbf{1}$ also satisfies the equation 8.

Proof. Let $H = \{z \in \mathbb{R}^d \mid \mathbf{1}^T z = 0\}$ be the zero-mean subspace. The normalization function L_ε maps every vector $x \in \mathbb{R}^d$ to a vector in H since $\mu(L_\varepsilon(x)) = 0$.

Step 1: Constructing the Equivariant Matrix M_0

First, we require the linear transformation $M = DA$ to preserve the zero-mean subspace H, i.e., $M(H) \subset H$. Let

$$\tilde{v} = (A^T)^{-1}\mathbf{1}, \quad v = \frac{\tilde{v}}{\|\tilde{v}\|}.$$

We define the diagonal matrix $D_\lambda = \lambda \cdot \mathrm{Diag}(v)$. For any $z \in H$, we check the mean of $D_\lambda A z$:

$$\mathbf{1}^T(D_\lambda A)z = (\mathbf{1}^T D_\lambda)(Az) = \lambda v^T Az = \lambda(A^T v)^T z = \frac{\lambda}{\|\tilde{v}\|}(\underbrace{A^T\tilde{v}}_{=\mathbf{1}})^T z = \frac{\lambda}{\|\tilde{v}\|}\mathbf{1}^T z = 0.$$

Thus, $(D_\lambda A)(H) \subset H$ for any λ.

Next, we ensure the matrix $M = D_\lambda A$ maps the domain of L_ε on H back into itself. Let $Q \in \mathbb{R}^{d\times(d-1)}$ be an orthonormal basis matrix for H. The operator norm of the restriction of M to H is $\left\|M_{|H}\right\|_2 = \|Q^T D_\lambda A Q\|_2$. We choose λ_0 such that this norm is exactly 1:

$$\lambda_0 = \left\|Q^T \,\mathrm{Diag}(v) A Q\right\|_2^{-1}.$$

Let $M_0 = D_{\lambda_0} A$. With this choice, M_0 is a λ-scaled affine transformation that preserves H and satisfies $\left\|M_{0|H}\right\|_2 = 1$.

Step 2: Defining the Domain and Unique Inverse

Let $\sigma_0(x) = \sqrt{\frac{1}{d}\sum_{i=1}^d (x_i - \mu(x))^2}$ be the standard deviation. We have:

$$\|L_\varepsilon(x)\| = \frac{\|x - \mu(x)\mathbf{1}\|}{\sigma_\varepsilon(x)} = \frac{\sigma_0(x)\sqrt{d}}{\sqrt{\sigma_0(x)^2 + \varepsilon}}.$$

Since $\frac{\sigma_0(x)^2}{\sigma_0(x)^2+\varepsilon} < 1$, the image of L_ε is the open ball:

$$B = \mathrm{Im}(L_\varepsilon) = \{z \in H \mid \|z\| < \sqrt{d}\}.$$

Since $\left\|M_{0|H}\right\|_2 = 1$ and $M_0(H) \subset H$, we have $M_0(B) \subset B$. The set B is the domain of the composition $M_0 L_\varepsilon(\cdot)$.

For any $z \in B$, the set of pre-images $L_\varepsilon^{-1}(z) = \{x \in \mathbb{R}^d \mid L_\varepsilon(x) = z\}$ is an equivalence class defined by the mean:

$$L_\varepsilon^{-1}(z) = \{c + \gamma\mathbf{1} \mid \gamma \in \mathbb{R}\},$$

where c is the unique vector satisfying $L_\varepsilon(c) = z$ and $\mu(c) = 0$. We define the unique inverse function $L_\varepsilon^{-1} : B \to H$ by choosing the representative with zero mean (i.e., we set $\mu(c) = 0$ or $c \in H$).

Solving $z = L_\varepsilon(c) = \frac{c}{\sqrt{\|c\|^2/d+\varepsilon}}$ for c (where $c \in H$, so $\mu(c) = 0$):

$$\|z\| = \frac{\|c\|}{\sqrt{\|c\|^2/d + \varepsilon}}.$$

Let $r = \|z\|$ and $s = \|c\|$. Solving $r = \frac{s\sqrt{d}}{\sqrt{s^2+\varepsilon}}$ for s yields $s = \frac{r\sqrt{\varepsilon}}{\sqrt{d-r^2}}$. The unique zero-mean pre-image is $c = \frac{s}{r}z$. Thus, we define the unique inverse on H:

$$L_\varepsilon^{-1}(z) = \sqrt{\frac{d\varepsilon}{d - \|z\|^2}}z. \quad \text{(Note: } L_\varepsilon^{-1}(z) \in H\text{)}$$

Step 3: Defining the Function f

We start by explicitly defining $f : \mathbb{R}^n \to H$ and then we extend it to a class of functions $\mathbf{f} \subset \left\{\tilde{f} : \mathbb{R}^n \to \mathbb{R}^n\right\}$. In the case of H our function is given by:

$$f : x \in H \mapsto L_\varepsilon^{-1}\left(M_0 L_\varepsilon(x)\right).$$

Since $L_\varepsilon(x)$ is invariant to the mean, $L_\varepsilon(x) = L_\varepsilon\left(x - c\mathbf{1}\right)$ for any scalar constant c.

We can therefore define $\tilde{f}$ as the class of $\mathbb{R}^d \to \mathbb{R}^d$ functions:

$$\tilde{f} = \{L_\varepsilon^{-1}\big(M_0 L_\varepsilon(.)\big) + h(.)\mathbf{1} | h \text{ is a } \mathbb{R}^n \to \mathbb{R} \text{ function}\}$$

Here, $L_\varepsilon(x), M_0 L_\varepsilon(x) \in B \subset H \hookrightarrow \mathbb{R}^n$ and $L_\varepsilon^{-1}(\dots) \in H$ (zero mean) and we remark that h need not be continuous or even measurable.

Finally, we verify the identity:

$$\begin{aligned} L_\varepsilon(f(x)) &= L_\varepsilon\Big(L_\varepsilon^{-1}\big(M_0 L_\varepsilon(x)\big) + h(x)\mathbf{1}\Big) \\ &= L_\varepsilon\Big(L_\varepsilon^{-1}\big(M_0 L_\varepsilon(x)\big)\Big) \quad \text{(Since } L_\varepsilon(\cdot) \text{ ignores the mean)} \\ &= M_0 L_\varepsilon(x) \\ &= (DA)\, L_\varepsilon(x). \end{aligned}$$

This holds for all $x \in \mathbb{R}^d$. □

Remark 3 (An informal discussion on the approximate linearity of f_M in high dimensions)**.** *Lemma 8.1 establishes a conjugacy relation through a carefully constructed matrix $M_0 = D_{\lambda_0} A$, where the diagonal scaling D_{λ_0} is specifically chosen to ensure M_0 maps the image ball B into itself. While this construction is essential for the conjugacy result, motivates a geometric question: does the composition $f_M(x) = L_\varepsilon^{-1}(ML_\varepsilon(x))$ exhibit approximate linearity for more general matrices M in high dimensions? Setting aside the lemma's specific construction, consider $f_M(x) = \sqrt{\frac{d\varepsilon}{\|x\|^2 + d\varepsilon - \|Mx\|^2}}\, Mx$, which is exactly linear when M is an isometry. For random matrices M with i.i.d. $\mathcal{N}(0, 1/d)$ entries and typical unit vectors x, concentration of measure [Ver18] yields $\|Mx\|^2 \approx \|x\|^2$ with high probability, making the coefficient approximately constant across typical inputs. This concentration effect, combined with the smoothing regime where $d\varepsilon$ is non-negligible relative to $\|x\|^2$, causes the square root term to be close to 1. Consequently, in high dimensions, f_M exhibits approximate linearity for typical M even when it is not an exact isometry, with the approximation quality improving as d increases.*

Theorem 8.2 (Basis Transformation Through LayerNorm)**.** *Let $\Theta \in GL(d)$ be a basis transformation, $MLP : \mathbb{R}^d \to \mathbb{R}^d$ a multilayer perceptron, and $D \in \mathbb{R}^{d\times d}$ a diagonal matrix with positive entries representing the learned LayerNorm scaling.*

If there exist a modified MLP $MLP' : \mathbb{R}^d \to \mathbb{R}^d$ and diagonal matrix $D' \in \mathbb{R}^{d\times d}$ such that

$$D' L_\varepsilon(x + MLP'(x)) = \Theta D L_\varepsilon(x + MLP(x)) \quad \forall x \in \mathbb{R}^d, \tag{9}$$

then by Lemma 8.1, D' and MLP' must satisfy:

$$\begin{aligned} D' &= \tilde{D}^{-1} \\ MLP'(x) &= L_\varepsilon^{-1}\Big(M_0\, L_\varepsilon(x + MLP(x))\Big) - x + h(x)\mathbf{1} \end{aligned} \tag{10}$$

where $\tilde{D} = Diag(v)\lambda_0$, $M_0 = Diag(v)\lambda_0 \cdot \Theta D$, and

$$L_\varepsilon^{-1}(z) = \sqrt{\frac{d\varepsilon}{d - \|z\|^2}}\, z, \tag{11}$$

with v and λ_0 as constructed in Lemma 8.1 for $A = \Theta D$ and $h : \mathbb{R}^n \to \mathbb{R}$ is any real function.

Note that h actually adds a single degree of freedom to the MLP' and can be used to pass on information to the skip connection.

Remark 4. *The argument above can be distilled down to the case of RMSNorm [ZS19], which has become the normalization of choice in many modern LLMs [LT24, Dee24, Tea25]. Furthermore, a similar reasoning applies to the recently proposed the Dynamic Tanh alternative to normalization [ZCH+25]. Indeed, consider the map*

$$DyT_{\gamma,\alpha,\beta} : \mathbb{R}^d \to \mathbb{R}^d, \quad (x_1, \dots, x_d) \mapsto \mathrm{diag}(\gamma)\big(\tanh(\alpha_1 x_1), \dots, \tanh(\alpha_d x_d)\big) + \beta.$$

This transformation is bijective onto its image whenever $\gamma_i \neq 0$ and $\alpha_i \neq 0$ for all i, a condition that is reasonable due to the stochasticity of the optimization. Hence, the structural result established above extends beyond LayerNorm to both RMSNorm and Dynamic Tanh.

Remark 5. *This corollary reveals that preserving a basis transformation through LayerNorm requires MLP' to approximate a highly nonlinear function involving the composition of normalization, the original MLP, and denormalization. Since standard MLPs of the form $W_2\sigma(W_1x)$ cannot exactly represent this composition, exact Query weight elimination with LayerNorm is not achievable using standard architectural components. This theoretical obstruction motivates either removing LayerNorm or accepting approximate equivalence with adjusted hyperparameters, as we pursue in our experiments.*

8.2 When Can Skip Connections Be Absorbed into ReLU MLPs?

The theorem below provides a precise characterization of when a skip connection appended to a single-hidden-layer ReLU MLP can be exactly absorbed into an equivalent residual-free ReLU mapping. This question connects two complementary lines of inquiry: the algebraic and reparameterization perspective on identity and residual blocks [HZRS16], and geometric analyses of ReLU networks that relate equality of piecewise-linear maps to alignment of their kink hyperplanes and neuron parameters [ABMM18, MPCB14].

Theorem 8.3 (Residual-free representability of skip-ReLU-MLPs)**.** *Let $h \geq 2$ and set $m = 4h$. Let*

$$W_1 \in \mathbb{R}^{m\times h}, \qquad W_2 \in \mathbb{R}^{h\times m}$$

be given matrices with $\operatorname{rank} W_1 = \operatorname{rank} W_2 = h$. *Denote* ReLU *coordinatewise. There exist matrices*

$$V_1 \in \mathbb{R}^{m\times h}, \qquad V_2 \in \mathbb{R}^{h\times m}$$

such that the identity

$$W_2\ \mathrm{ReLU}(W_1x) + x\ =\ V_2\ \mathrm{ReLU}(V_1x) \qquad \textit{holds for all } x \in \mathbb{R}^h$$

if and only if there exists an index set $J \subset \{1,\dots,m\}$ with $|J| \geq h$ for which

$$W_2[:, J]\, W_1[J, :] = -I_h.$$

Proof. We prove both directions.

(Sufficiency). Suppose $J \subset \{1,\dots,m\}$ satisfies $|J| \geq h$ and

$$W_2[:, J]\, W_1[J, :] = -I_h.$$

Let $\Pi = \operatorname{diag}(\mathbf{1}_J) \in \{0,1\}^{m\times m}$ be the coordinate projector onto J, and set the sign diagonal $D := I - 2\Pi \in \{\pm1\}^{m\times m}$. Define

$$V_1 := DW_1, \qquad V_2 := W_2.$$

For any $y \in \mathbb{R}^m$ write $\mathrm{ReLU}(y) = \frac{1}{2}(y + |y|)$. Since D has diagonal entries ±1 we have $|V_1x| = |W_1x|$ for all x. Hence

$$\begin{aligned} V_2\,\mathrm{ReLU}(V_1x) &= \tfrac{1}{2}V_2\big(V_1x + |V_1x|\big) = \tfrac{1}{2}\big(V_2V_1x + V_2|W_1x|\big), \\ W_2\,\mathrm{ReLU}(W_1x) + x &= \tfrac{1}{2}\big(W_2W_1x + W_2|W_1x| + 2x\big). \end{aligned}$$

Using $D = I - 2\Pi$ and $W_2\Pi W_1 = -I_h$ we obtain

$$V_2V_1 = W_2DW_1 = W_2(I - 2\Pi)W_1 = W_2W_1 - 2(W_2\Pi W_1) = W_2W_1 + 2I_h.$$

Substituting into the displayed expressions yields $V_2\,\mathrm{ReLU}(V_1x) = W_2\,\mathrm{ReLU}(W_1x) + x$ for all x. Thus the condition on J is sufficient.

(Necessity). Suppose the identity $W_2\ \mathrm{ReLU}(W_1x) + x = V_2\ \mathrm{ReLU}(V_1x)$ holds for all $x \in \mathbb{R}^h$. Using $\mathrm{ReLU}(y) = \frac{1}{2}(y + |y|)$, the identity becomes:

$$W_2\,\tfrac{1}{2}(W_1x + |W_1x|) + x = V_2\,\tfrac{1}{2}(V_1x + |V_1x|)$$

Multiplying by 2 and separating the odd and even terms in x yields:

$$\big(W_2W_1 + 2I_h - V_2V_1\big)x = V_2|V_1x| - W_2|W_1x|$$

The left-hand side is an odd function of x, while the right-hand side is an even function. This implies both sides must be identically zero, giving two conditions:

$$V_2V_1 = W_2W_1 + 2I_h \tag{N1}$$

$$V_2|V_1x| = W_2|W_1x| \quad \text{for all } x \in \mathbb{R}^h \tag{N2}$$

The set of hyperplanes where the networks are non-differentiable (the "kinks") must be identical. This means $\mathcal{H}_W = \{x \mid (W_1x)_i = 0\} = \{x \mid (V_1x)_i = 0\} = \mathcal{H}_V$. This implies that the rows of V_1 must be a scaled permutation of the rows of W_1. We can express this relationship as $V_1 = DPW_1$ for some diagonal matrix $D = \text{diag}(d_1, \ldots, d_m)$ with $d_i \neq 0$ and some permutation matrix P.

Substitute $V_1 = DPW_1$ into (N2):

$$V_2|DPW_1x| = W_2|W_1x|$$

Let $D_a = \text{diag}(|d_1|, \ldots, |d_m|)$. Since a permutation only reorders elements, $|Py| = P|y|$. The equation becomes:

$$V_2D_aP|W_1x| = W_2|W_1x|$$

Since $h \geq 2$, the set of vectors $\{|W_1x| : x \in \mathbb{R}^h\}$ is rich enough to imply the matrix identity $V_2D_aP = W_2$. Solving for V_2 gives $V_2 = W_2P^{-1}D_a^{-1} = W_2P^TD_a^{-1}$.

Now substitute both $V_1 = DPW_1$ and $V_2 = W_2P^TD_a^{-1}$ into (N1):

$$(W_2P^TD_a^{-1})(DPW_1) = W_2W_1 + 2I_h$$

$$W_2P^T(D_a^{-1}D)PW_1 = W_2W_1 + 2I_h$$

Let $S = D_a^{-1}D$. This is a diagonal matrix with entries $S_{jj} = d_j/|d_j| = \text{sign}(d_j) \in \{\pm 1\}$. Let $D' = P^TSP$. This similarity transformation simply permutes the diagonal entries of S, so D' is also a diagonal matrix with entries ± 1. The equation simplifies to:

$$W_2D'W_1 = W_2W_1 + 2I_h$$

$$W_2(D' - I)W_1 = 2I_h$$

Let $J \subset \{1, \ldots, m\}$ be the index set where the diagonal entries of D' are -1. Then $D' - I = -2\Pi$, where $\Pi = \text{diag}(\mathbf{1}_J)$ is the coordinate projector onto J. Substituting this gives:

$$W_2(-2\Pi)W_1 = 2I_h \implies W_2\Pi W_1 = -I_h$$

The matrix $W_2\Pi W_1$ is precisely the product $W_2[:, J]\, W_1[J, :]$. Thus, we have $W_2[:, J]\, W_1[J, :] = -I_h$.

Finally, by Sylvester's rank inequality:

$$h = \text{rank}(W_2[:, J]\, W_1[J, :]) \leq \min(\text{rank}(W_2[:, J]), \text{rank}(W_1[J, :]))$$

Since $\text{rank}(W_1[J, :]) \leq |J|$, we must have $|J| \geq h$. This completes the proof of necessity. □

Corollary 8.3.1 (Uniqueness under invertible residual perturbations)**.** *Let $h \in \mathbb{N}$ with $h \geq 2$, and set $m = 4h$. Denote by $\sigma : \mathbb{R}^m \to \mathbb{R}^m$ the coordinatewise ReLU function.*

For matrices $W_1 \in \mathbb{R}^{m\times h}$, $W_2 \in \mathbb{R}^{h\times m}$, and $Z \in \mathbb{R}^{h\times h}$, define the map

$$\phi_{W_1,W_2,Z} : \mathbb{R}^h \to \mathbb{R}^h, \quad x \mapsto W_2\,\sigma(W_1x) + Zx.$$

Equip the space

$$\mathcal{W} := \mathbb{R}^{m\times h} \times \mathbb{R}^{h\times m} \times \mathbb{R}^{h\times h}$$

with its natural Lebesgue measure. Then for Lebesgue-almost every $(W_1, W_2, Z) \in \mathcal{W}$, the following holds:

If $Z' \in \mathbb{R}^{h\times h}$ satisfies $Z' \neq Z$ and $Z - Z'$ is invertible, then for every $(W_1', W_2') \in \mathbb{R}^{m\times h} \times \mathbb{R}^{h\times m}$,

$$\phi_{W_1,W_2,Z} \neq \phi_{W_1',W_2',Z'}.$$

8.3 Proofs of some results

Proof of Theorem 4.1. We first clarify the architecture. The complete model consists of an embedding layer, n transformer blocks, and a final LM Head (unembedding layer). For a token sequence $x = (x_0, \ldots, x_k)$, let $E[x]$ denote the matrix obtained by looking up the embeddings for each token, and let $E_P[0:k]$ denote the positional embeddings for positions 0 through k. The input to the first layer is $X_0 = E[x] + E_P[0:k]$.

In the original model, each block-layer B_i for $1 \leq i \leq L$ consists of an attention layer with skip connection followed by an MLP layer without skip connection:

$$B_i(X) = \phi((X + Att(X, W_Q^i, W_K^i, W_V^i, W_O^i))W_{MLP_u}^i)W_{MLP_d}^i,$$

where ϕ denotes the element-wise activation function (e.g., ReLU or GELU). The final output is computed as $X_n W_{LM}$ where W_{LM} is the unembedding matrix and X_n is the output after all n blocks.

We prove by induction on i that after processing i layers with appropriately modified weights, the output equals the output of the original model right-multiplied by Θ_i.

Base case ($i = 0$, input to first layer): Set $\Theta_1 = W_Q^1$ and define $\widetilde{E} = E\Theta_1$ and $\widetilde{E}_P = E_P\Theta_1$. For the token sequence x, the input to the first layer in the modified model is:

$$\widetilde{X}_0 = \widetilde{E}[x] + \widetilde{E}_P[0:k] = E[x]\Theta_1 + E_P[0:k]\Theta_1 = (E[x] + E_P[0:k])\Theta_1 = X_0\Theta_1,$$

where $X_0 = E[x] + E_P[0:k]$ is the input in the original model.

Inductive step ($1 \leq i \leq L$): Assume that after layer $i-1$, the output of the modified model equals $X_{i-1}\Theta_i$, where X_{i-1} denotes the output of the original model after layer $i-1$.

For the layer i, set $\Theta_k = W_Q^k$ for $k \in \{i, i+1\}$ and define the modified weights as follows:

$$\begin{aligned}
&\Theta_{n+1} = (\Theta_1^t)^{-1} \quad \text{if the LMHead and embedding weights are tied} \\
&\Theta_{n+1} = \mathrm{Id} \quad \text{otherwise} \\
&\widetilde{W}_K^i = \Theta_i^{-1} W_K^i, \\
&\widetilde{W}_V^i = \Theta_i^{-1} W_V^i, \\
&\widetilde{W}_O^i = W_O^i \Theta_i, \\
&\widetilde{W}_{MLP_u}^i = \Theta_i^{-1} W_{MLP_u}^i \\
&\widetilde{W}_{MLP_d}^i = W_{MLP_d}^i \Theta_{i+1}
\end{aligned}$$

With these modified weights, the output of layer i in the modified model is:

$$\begin{aligned}
&\phi\left(\left(X_{i-1}\Theta_i + Att(X_{i-1}\Theta_i, \mathrm{Id}, \widetilde{W}_K^i, \widetilde{W}_V^i, \widetilde{W}_O^i)\right)\widetilde{W}_{MLP_u}^i\right)\widetilde{W}_{MLP_d}^i \\
&= \phi\left(\left(X_{i-1}\Theta_i + \mathcal{S}_c(X_{i-1}\Theta_i, \mathrm{Id}, \widetilde{W}_K^i, \widetilde{W}_V^i)\widetilde{W}_O^i\right)\Theta_i^{-1}W_{MLP_u}^i\right)W_{MLP_d}^i\Theta_{i+1} \\
&= \phi\left(\left(X_{i-1}\Theta_i + \mathcal{S}_c(X_{i-1}\Theta_i, \mathrm{Id}, \Theta_i^{-1}W_K^i, \Theta_i^{-1}W_V^i)W_O^i\Theta_i\right)\Theta_i^{-1}W_{MLP_u}^i\right)W_{MLP_d}^i\Theta_{i+1} \\
&= \phi\left(\left(X_{i-1}\Theta_i + \mathcal{S}_c(X_{i-1}, W_Q^i, W_K^i, W_V^i)W_O^i\Theta_i\right)\Theta_i^{-1}W_{MLP_u}^i\right)W_{MLP_d}^i\Theta_{i+1} \\
&= \phi\left(\left(X_{i-1} + \mathcal{S}_c(X_{i-1}, W_Q^i, W_K^i, W_V^i)W_O^i\right)W_{MLP_u}^i\right)W_{MLP_d}^i\Theta_{i+1} \\
&= B_i(X_{i-1})\Theta_{i+1},
\end{aligned}$$

where the transition from line 3 to line 4 applies the Reparametrization Lemma to $\mathcal{S}_c$.

By induction, after all n layers, the output of the modified model is $X_n\Theta_{n+1}$, where X_n is the output of layer n in the original model.

Finally, for the LM Head weights: In the case where the LM Head weights W_{LM} and embedding weights E are not tied, we have $\Theta_{n+1} = \mathrm{Id}$, so the output is X_n. Setting $\widetilde{W}_{LM} = W_{LM}$ gives $X_n \cdot \widetilde{W}_{LM} = X_n \cdot W_{LM}$.

In the case of tied weights, we have $\widetilde{W}_{LM} = \widetilde{E}^t = \Theta_1^t E^t$, so:

$$X_n\Theta_{n+1} \cdot \widetilde{W}_{LM} = X_n(\Theta_1^t)^{-1} \cdot \Theta_1^t W_{LM} = X_n W_{LM},$$

yielding exactly the same final output logits in both the reduced and the original model. □

Proof of Theorem 4.2. In order to prove the theorem we will rely on the following lemma:

Lemma 8.4 (Blocks are conjugate to reduced blocks)**.** *Let* $Block : \mathbb{R}^{l\times d} \to \mathbb{R}^{l\times d}$ *be a transformer block without layer normalization given by*

$$Block(X) = (Id + MLP) \circ (Id + Att)(X)$$

where Att uses weights (W_Q, W_K, W_V, W_O) *with* $W_Q \in GL(d)$*, and* $MLP(Y) = \varphi(YW_{up})W_{down}$ *for an element-wise activation* φ*.*

Then there exists $\Theta \in GL(d)$ *and a reduced block* $\widetilde{Block}$ *using weights* $(Id, \widetilde{W}_K, \widetilde{W}_V, \widetilde{W}_O, \widetilde{W}_{up}, \widetilde{W}_{down})$ *such that*

$$Block = \Theta^{-1} \circ \widetilde{Block} \circ \Theta.$$

In other words, for almost every choice of weights (in the Lebesgue sense), a transformer block without normalization is conjugate to a reduced block.

Proof. Set $\Theta = W_Q$, $\widetilde{W}_K = \Theta^{-1}W_K$, $\widetilde{W}_V = \Theta^{-1}W_V$, $\widetilde{W}_O = W_O\Theta$, $\widetilde{W}_{up} = \Theta^{-1}W_{up}$, and $\widetilde{W}_{down} = W_{down}\Theta$. □

Let us now continue with the proof of the result by considering the new variables as in the proof of the lemma, and let furthermore

$$\widetilde{E} = E\Theta, \quad \widetilde{E}_P = E_P\Theta, \quad \widetilde{W}_{LM} = \Theta^{-1}W_{LM}.$$

With these parameters, applying the lemma, each block satisfies

$$\text{Block} = \Theta^{-1} \circ \widetilde{\text{Block}} \circ \Theta.$$

Since all layers share the same block, the L-layer composition telescopes:

$$\text{Block}^{\circ L} = (\Theta^{-1} \circ \widetilde{\text{Block}} \circ \Theta)^{\circ L} = \Theta^{-1} \circ \left(\widetilde{\text{Block}}\right)^{\circ L} \circ \Theta.$$

The embedding and output head reparameterizations complete the proof. □

Acknowledgements

We would like to thank Nils Graef, Borjan Geshkovski and François Yvon for helpful discussions on this work. The first author would also like to thank Igor Ševo for discussions around the attention mechanism.

References

[ABMM18] Sanjeev Arora, Amitabh Basu, Poorya Mianjy, and Anirbit Mukherjee. Understanding deep neural networks with rectified linear units. In *International Conference on Machine Learning (ICML)*, pages 2810–2819, 2018. Available as arXiv:1611.01491 (extended versions).

[ALTdJ+23] Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. Gqa: Training generalized multi-query transformer models from multi-head checkpoints. *arXiv preprint arXiv:2305.13245*, 2023.

[Bar98] Peter L Bartlett. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Transactions on Information Theory*, 44(2):525–536, 1998.

[Bie20] Lukas Biewald. Weights & biases. https://www.wandb.com/, 2020. Software available from wandb.com.

[BKH16] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[BKK19] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. Deep equilibrium models. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[BKS+25] Luca Baroni, Galvin Khara, Joachim Schaeffer, Marat Subkhankulov, and Stefan Heimersheim. Transformers don't need layernorm at inference time: Scaling layernorm removal to gpt-2 xl and the implications for mechanistic interpretability. *arXiv preprint arXiv:2507.02559*, 2025.

[BMR+20] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[CLD+20] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

[Dao24] Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*, 2024.

[Dee24] DeepSeek-AI. Deepseek-v3: Scaling open-source language models with efficient training. *arXiv preprint arXiv:2412.19437*, 2024.

[DFE+22] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[DHD+22] Nan Du, Yanping Huang, Andrew M Dai, Shixiang Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, 2022.

[DPHZ24] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024.

[FZS21] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. In *International Conference on Learning Representations*, 2021.

[GC19] Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus, 2019.

[GLPR23] Borjan Geshkovski, Cyril Letrouit, Yury Polyanskiy, and Philippe Rigollet. The emergence of clusters in self-attention dynamics. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[GLPR24] Borjan Geshkovski, Cyril Letrouit, Yury Polyanskiy, and Philippe Rigollet. A mathematical perspective on transformers. In *Advances in Neural Information Processing Systems*, volume 36, 2024.

[Gra24] Nils Graef. Transformer tricks: Removing weights for skipless transformers. *arXiv preprint arXiv:2404.12362*, 2024.

[GZA+23] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. Textbooks are all you need, 2023.

[Hei24] Stefan Heimersheim. You can remove gpt2's layernorm by fine-tuning. *arXiv preprint arXiv:2409.13710*, 2024. Presented at the ATTRIB and Interpretable AI workshops, NeurIPS 2024.

[HG16] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[HGJ+19] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. *ArXiv*, abs/1902.00751, 2019.

[HH24] Bobby He and Thomas Hofmann. Simplifying transformer blocks. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, editors, *International Conference on Representation Learning*, volume 2024, pages 8882–8910, 2024.

[HMN91] Roger A. Horn, Roy Mathias, and Yoshihiro Nakamura. Inequalities for unitarily invariant norms and bilinear matrix products. *Linear and Multilinear Algebra*, 30(4):303–314, November 1991.

[HMZ+23] Bobby He, James Martens, Guodong Zhang, Aleksandar Botev, Andrew Brock, Samuel L Smith, and Yee Whye Teh. Deep transformers without shortcuts: Modifying self-attention for faithful signal propagation. In *International Conference on Learning Representations (ICLR)*, 2023.

[HSW+21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision (ECCV)*, pages 630–645, 2016.

[IHL+24] Muhammed Emrullah Ildiz, Yixiao Huang, Yingcong Li, Ankit Singh Rawat, and Samet Oymak. From self-attention to Markov models: Unveiling the dynamics of generative transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 20955–20982. PMLR, 2024.

[JJNH91] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

[JSML25] Yiping Ji, Hemanth Saratchandran, Peyman Moghadam, and Simon Lucey. Always skip attention, 2025.

[Kar] Andrej Karpathy. Nanogpt. https://github.com/karpathy/nanoGPT/blob/master/model.py.

[KH24] Marko Karbevski and HTEC Group. HTEC AI Open Day Skopje — Possible Modifications to the Attention Mechanism. YouTube video. Available at: https://www.youtube.com/watch?v=UjTHj-cFJOA, 2024. Accessed: October 19, 2025.

[L+21] Quentin Lhoest et al. Datasets: A community library for natural language processing. https://github.com/huggingface/datasets, 2021. Version 1.0.

[LCG+20] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In *International Conference on Learning Representations (ICLR)*, 2020.

[LH19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[LLX+21] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*, 2021.

[LT24] AI @ Meta Llama Team. The llama 3 herd of models. *ArXiv*, abs/2407.21783, 2024.

[MFW23] Xinyin Ma, Gongfan Fang, and Xinchao Wang. Llm-pruner: On the structural pruning of large language models. *Advances in neural information processing systems*, 36:21702–21720, 2023.

[Mis23] Mistral AI. Mixtral of experts. https://mistral.ai/news/mixtral-of-experts/, 2023.

[MPCB14] Guillermo F. Montúfar, Razvan Pascanu, Kyunghyun Cho, and Yoshua Bengio. On the number of linear regions of deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 27, pages 2924–2932, 2014. NeurIPS 2014.

[MWM+24] Shuming Ma, Hongyu Wang, Lingxiao Ma, Lei Wang, Wenhui Wang, Shaohan Huang, Li Dong, Ruiping Wang, Jilong Xue, and Furu Wei. The era of 1-bit llms: All large language models are in 1.58 bits. *arXiv preprint arXiv:2402.17764*, 2024.

[NBMS17] Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[OAA+25] OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Boaz Barak, et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

[PPXS24] Xu Pan, Aaron Philip, Ziqian Xie, and Odelia Schwartz. Dissecting query-key interaction in vision transformers. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

[PW17] Ofir Press and Lior Wolf. Using the output embedding to improve language models. In Mirella Lapata, Phil Blunsom, and Alexander Koller, editors, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163, Valencia, Spain, April 2017. Association for Computational Linguistics.

[RWC+19] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

[Sha19] Noam M. Shazeer. Fast transformer decoding: One write-head is all you need. *ArXiv*, abs/1911.02150, 2019.

[SLP+21] Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.

[SMM+17] Noam Shazeer, Azalia Mirhoseini, Piotr Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017.

[SZM+23] Siddharth Samsi, Dan Zhao, Joseph McDonald, Baolin Li, Adam Michaleas, Michael Jones, William Bergeron, Jeremy Kepner, Devesh Tiwari, and Vijay Gadepally. From words to watts: Benchmarking the energy costs of large language model inference. In *2023 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–9. IEEE, 2023.

[TDBM22] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *ACM Computing Surveys*, 55(6), 2022.

[Tea25] Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[Ver18] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc.

[WLK+20] Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

[XXQ+23] Lingling Xu, Haoran Xie, Si-Zhao Joe Qin, Xiaohui Tao, and Fu Lee Wang. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *arXiv preprint arXiv:2312.12148*, 2023.

[YBR+19] Chulhee Yun, Srinadh Bhojanapalli, Ankit Singh Rawat, Sashank J Reddi, and Sanjiv Kumar. Are transformers universal approximators of sequence-to-sequence functions? *arXiv preprint arXiv:1912.10077*, 2019.

[ZCH+25] Jiachen Zhu, Xinlei Chen, Kaiming He, Yann LeCun, and Zhuang Liu. Transformers without normalization. *arXiv preprint arXiv:2503.10622*, 2025. CVPR 2025.

[ZS19] Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019.

[ZXZ+19] Guangxiang Zhao, Jingjing Xu, Zhiyuan Zhang, Liangchen Luo, Zhengdong Lu, and Xu Sun. Muse: Parallel multi-scale attention for sequence to sequence learning. *arXiv preprint arXiv:1911.09483*, 2019.