

Transformer Block

Pre-Norm Decoder Block: RMSNorm + GQA + RoPE + SwiGLU

The fundamental repeated unit of every modern LLM.
Wires together RMSNorm, grouped-query attention with RoPE,
and a SwiGLU FFN into the pre-norm architecture used by
Llama, Mistral, and all modern open-weight models.

This demo covers:

1. Full forward pass walkthrough with shape tracing
2. Parameter distribution for Llama 2/3 and Mistral configs
3. SwiGLU gating mechanism visualization
4. Residual connection gradient highway analysis
5. FLOPs breakdown with analytical crossover points
6. Causal masking and RoPE position sensitivity

Random seed: 42

Number of visualizations: 6

Generated by `demo.py`

Examples: 6

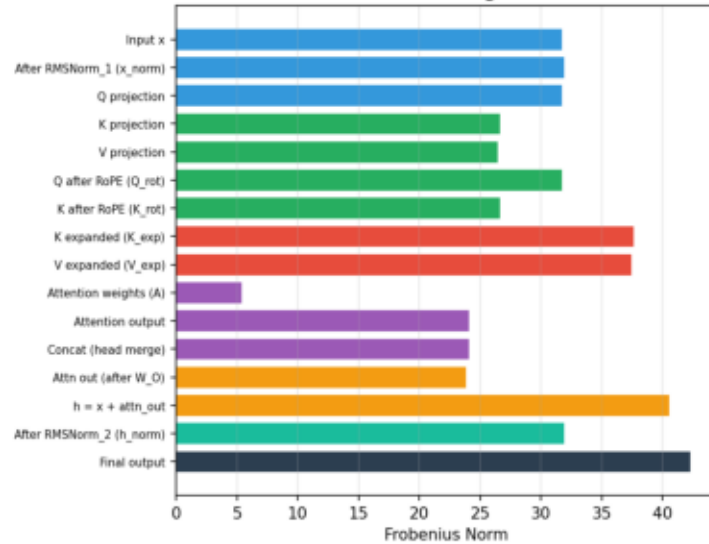
Summary of Findings

1. Forward Pass: Output shape matches input (B, L, d_model). Pre-norm architecture normalizes inputs to sublayers (RMS ~ 1.0) while the residual stream remains unnormalized and grows with depth.
2. Parameter Distribution: FFN (SwiGLU) dominates per-block parameters: $\sim 67\%$ with MHA (Llama 2 7B), $\sim 80\%+$ with GQA (70B, Llama 3, Mistral). GQA reduces attention params (Llama 2 70B saves $\sim 44\%$ vs MHA by using 8 KV heads for 64 Q heads). Norms are negligible ($< 0.01\%$).
3. SwiGLU Gating: The gate signal $\text{SiLU}(x @ W_{\text{gate}})$ selectively suppresses features. SiLU is smooth (no dead neurons unlike ReLU). The gating mechanism enables learned feature selection. 3 matrices vs 2, but $d_{\text{ff}} = 8/3 * d$ compensates for the extra parameters.
4. Residual Connections: The 'gradient highway' ensures gradients never vanish regardless of depth. Pre-norm gives $d(\text{output})/d(x) = I + \dots$. The identity term persists through all layers. With $1/\sqrt{N}$ weight scaling, norms grow moderately and gradients remain stable.
5. FLOPs: Attention core is $O(L^2)$ while FFN and projections are $O(L)$. For Llama 3 8B, attention core surpasses FFN at $L \sim 21,296$ tokens. Formula: $L_{\text{cross}} = 6*d*d_{\text{ff}} / ((4*d_k+5)*h)$. At short sequences, FFN dominates; at long sequences, attention core dominates.
6. Causal Masking: Position i 's output depends only on positions $0..i$. Verified by showing that changing future tokens has zero effect on past outputs. RoPE makes identical tokens position-aware through rotation, even without any trained weights.

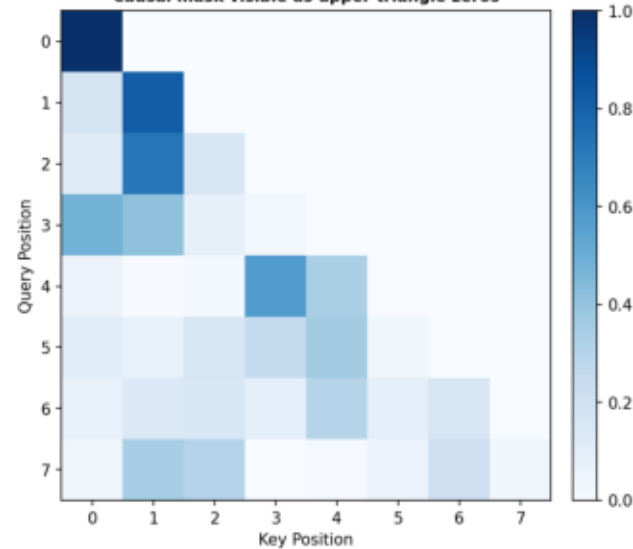
Example 1: Full Forward Pass Walkthrough

Transformer Block: Full Forward Pass Walkthrough

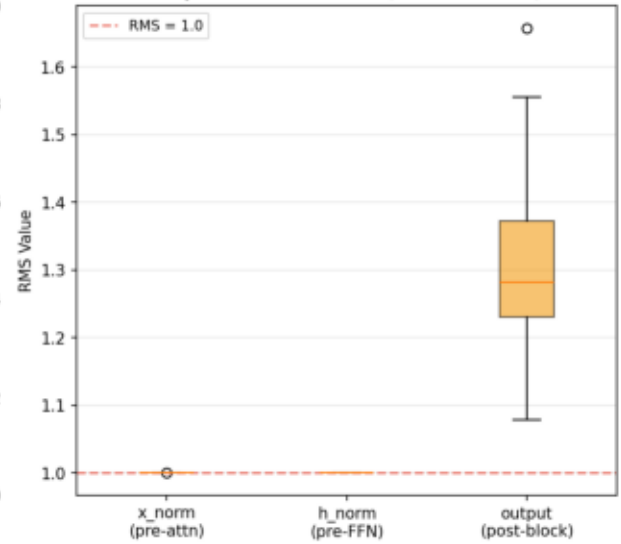
Tensor Norms Through the Block



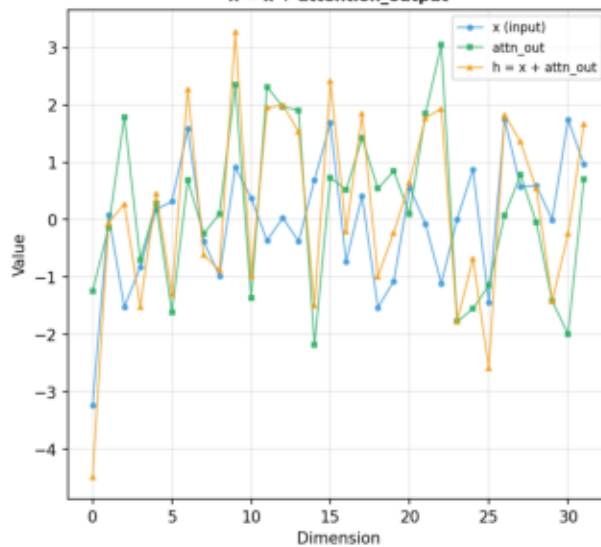
Attention Weights (B=0, Head=0)
Causal mask visible as upper triangle zeros



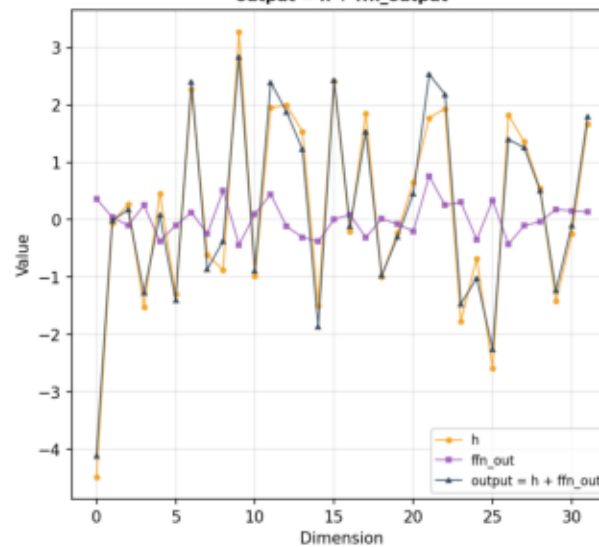
Pre-Norm: Normalized Inputs to Sublayers
Output is NOT normalized (residual stream)



Residual Connection 1 (First 32 dims)
 $h = x + \text{attention_output}$



Residual Connection 2 (First 32 dims)
 $\text{output} = h + \text{ffn_output}$



PRE-NORM DECODER BLOCK

```
d_model = 64
num_heads = 4 (h_kv = 2)
d_ff = 172
d_k = 16
```

Data Flow:

```
x -> RMSNorm 1 -> Q,K,V proj
    -> RoPE(Q,K) -> GQA -> W_O
    -> + x (residual 1) = h
h -> RMSNorm 2 -> SwiGLU FFN
    -> + h (residual 2) = output
```

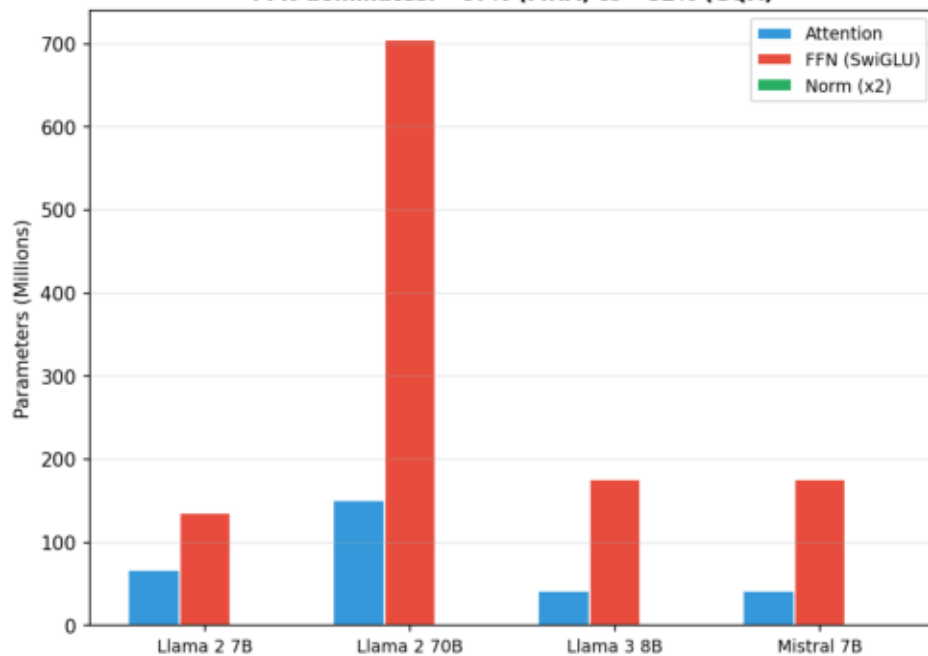
Key observations:

- x_norm RMS ~ 1.000 (normalized)
- h_norm RMS ~ 1.000 (normalized)
- output RMS ~ 1.315 (not normalized)
- Residual stream grows unboundedly (this is by design in pre-norm)

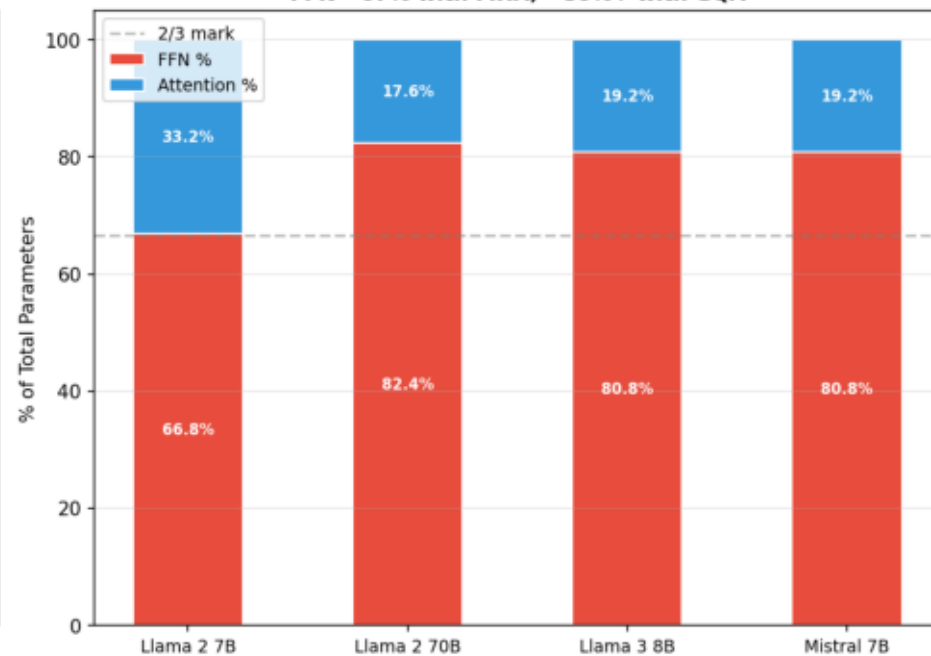
Example 2: Parameter Distribution Analysis

Transformer Block: Parameter Distribution Across Real LLM Configs

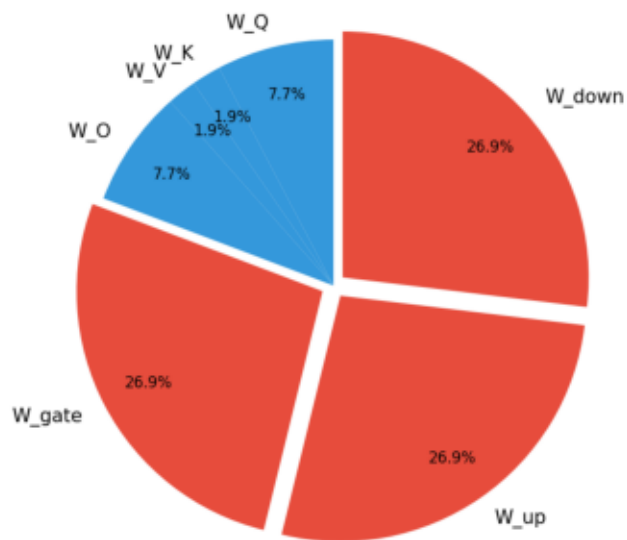
Parameter Breakdown per Block
FFN dominates: ~67% (MHA) to ~82% (GQA)



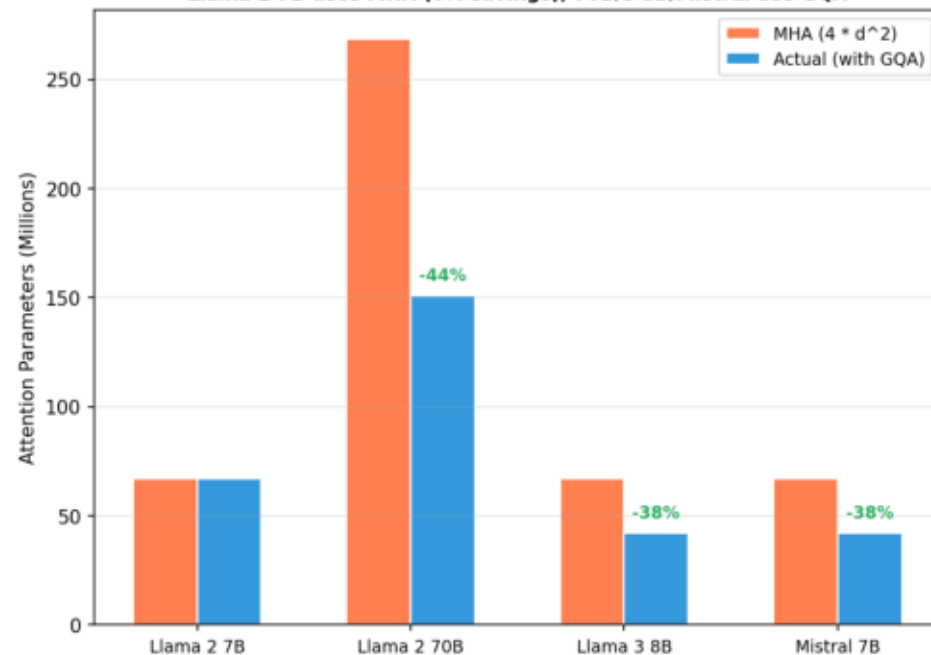
Attention vs FFN Share
FFN ~67% with MHA, ~80%+ with GQA



Parameter Pie Chart: Llama 3 8B
Total = 218.1M per block

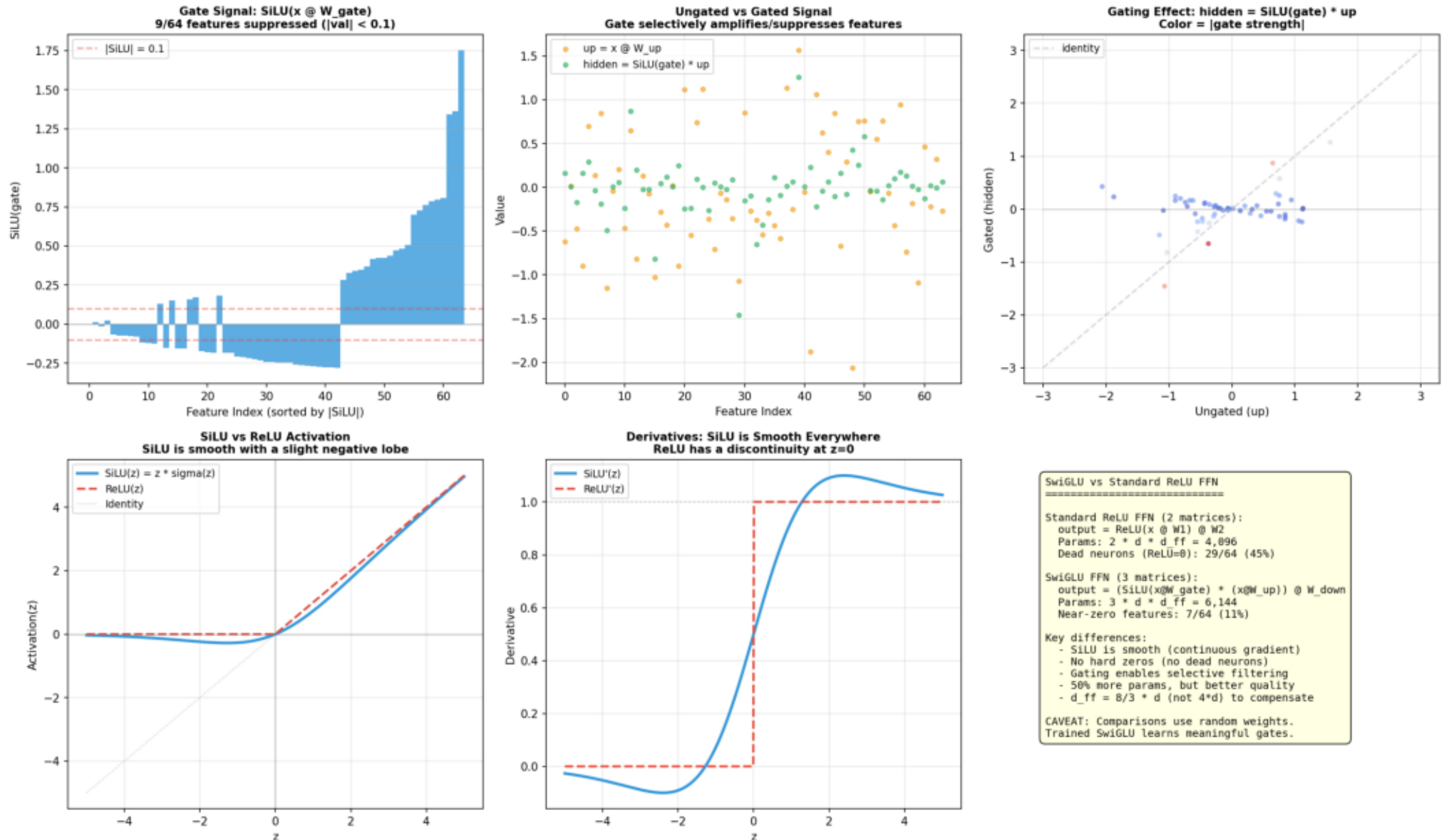


GQA Reduces Attention Parameters
Llama 2 7B uses MHA (0% savings); 70B/3 8B/Mistral use GQA



Example 3: SwiGLU Gating Visualization

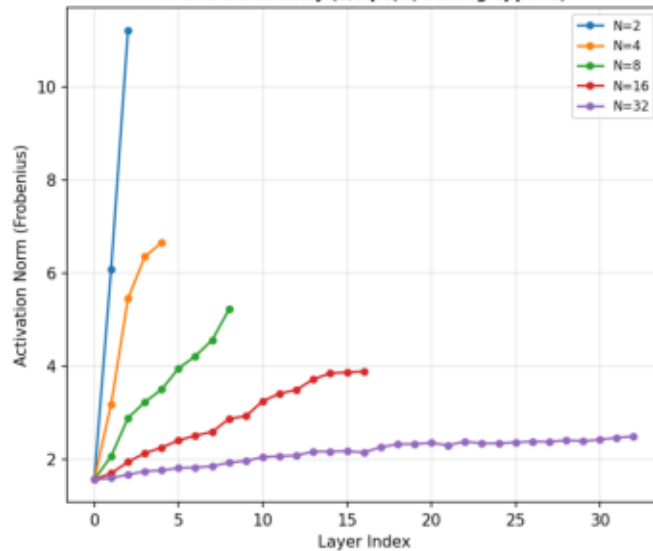
SwiGLU Gating Mechanism: Smooth, Selective Feature Filtering



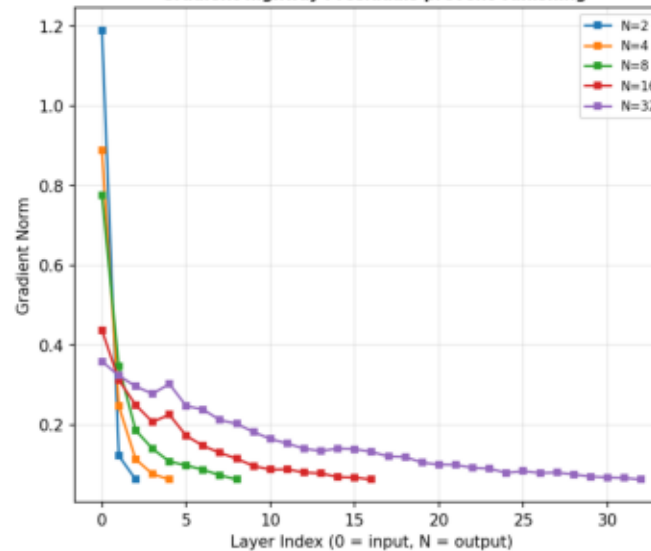
Example 4: Residual Connection Gradient Highway

Residual Connection Analysis: Gradient Highway Effect in Pre-Norm Transformer

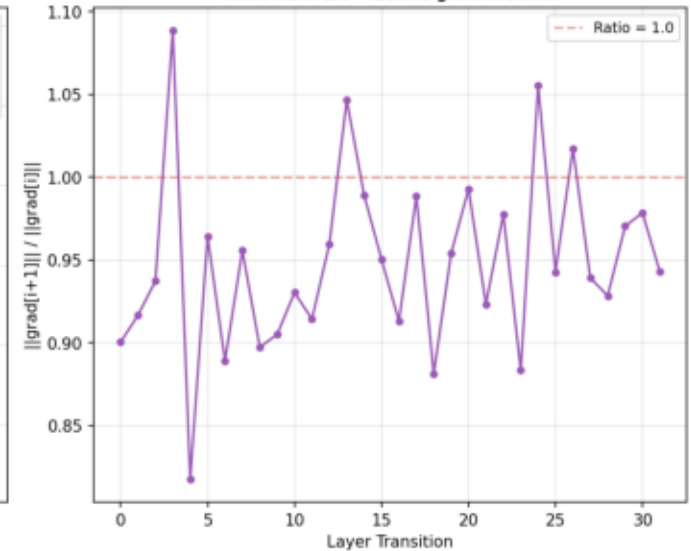
Forward: Activation Norms Through Layers
Grows moderately ($1/\sqrt{N}$ scaling applied)



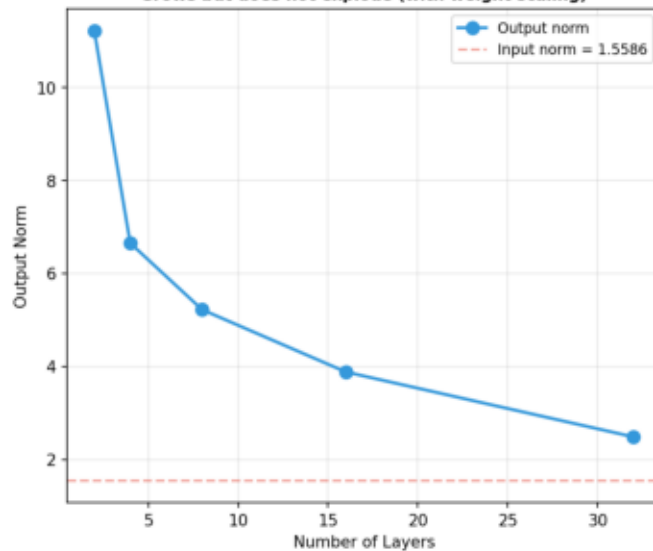
Backward: Gradient Norms Through Layers
'Gradient highway': residuals prevent vanishing



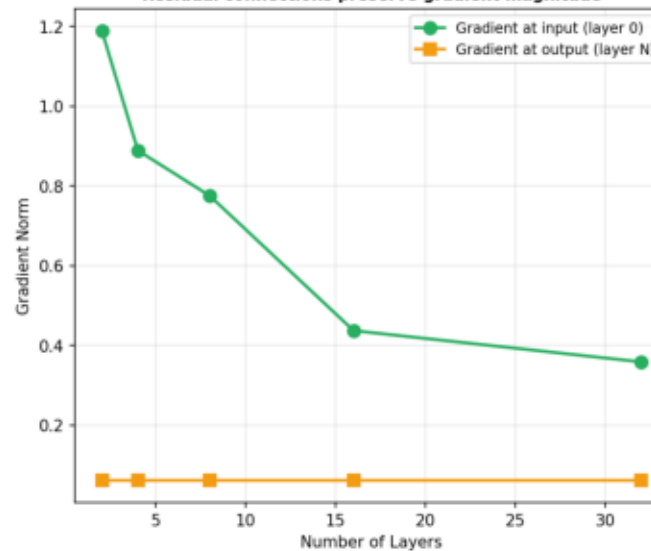
Gradient Ratio Between Adjacent Layers (N=32)
Ratio near 1.0 = stable gradient flow



Output Norm vs Depth
Grows but does not explode (with weight scaling)



Gradient at Input vs Output Layer
Residual connections preserve gradient magnitude



RESIDUAL CONNECTIONS AS GRADIENT HIGHWAYS

Pre-norm architecture:
 $\text{output} = x + \text{sublayer}(\text{Norm}(x))$

Gradient at residual:
$$\frac{d(\text{output})}{d(x)} = I + \frac{d(\text{sublayer})}{d(\text{Norm}(x))} * \frac{d(\text{Norm})}{d(x)}$$

The identity term I ensures:
 $\| \text{grad}_x \| \geq \| \text{grad}_{\text{output}} \| - \| \text{sublayer grad} \|$

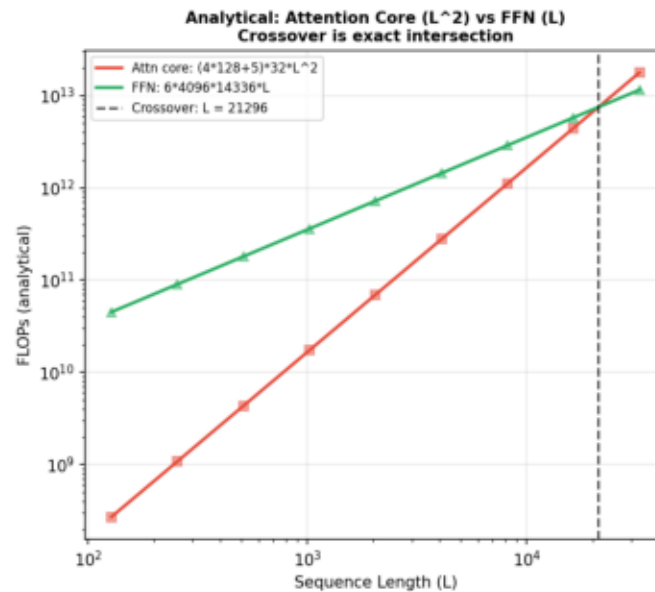
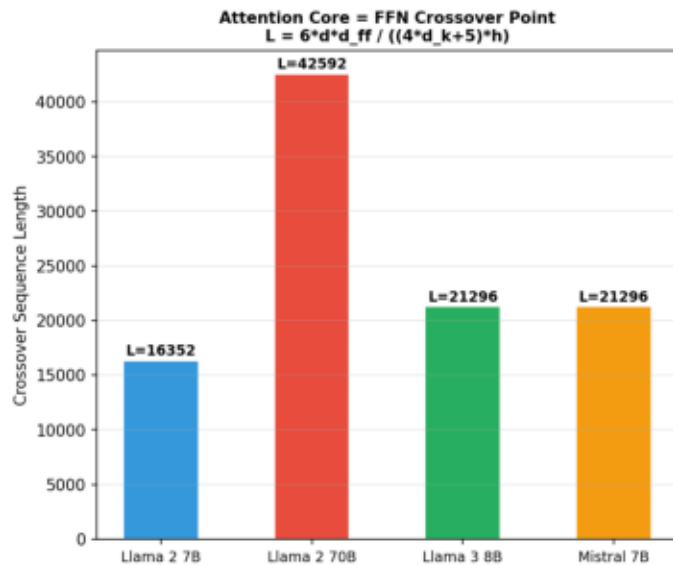
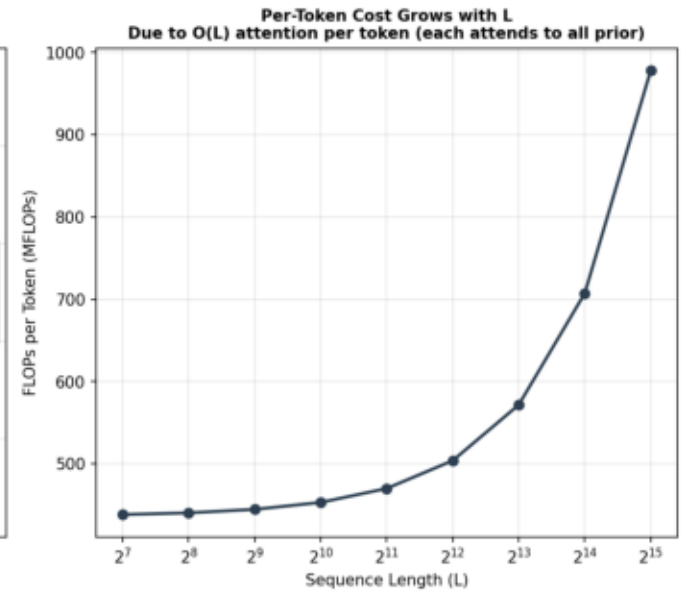
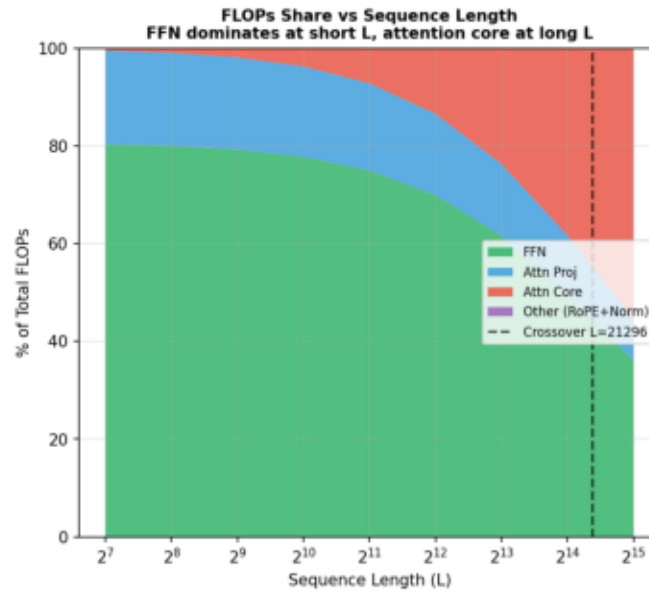
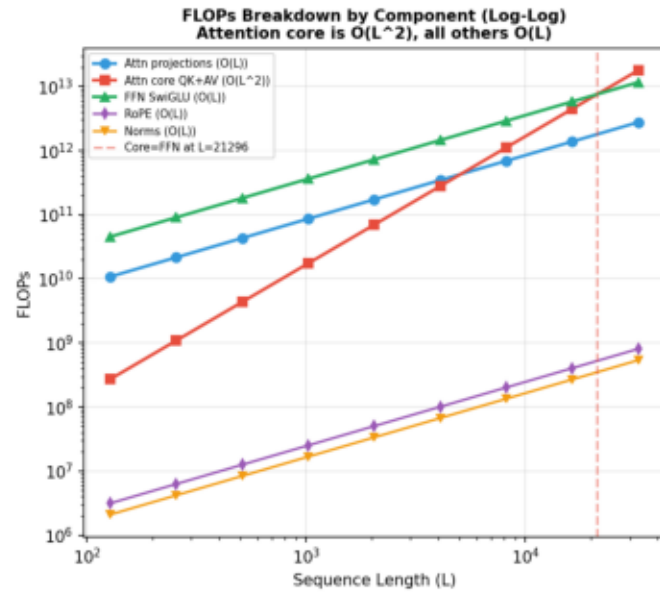
Through N blocks:
grad includes $I^N = I$
-> NEVER vanishes (gradient highway)

Post-norm (for comparison):
 $\text{output} = \text{Norm}(x + \text{sublayer}(x))$
grad must pass through Norm backward
-> no guaranteed I term
-> potential vanishing gradients

This is why ALL modern LLMs use pre-norm (GPT-2 onwards).

Example 5: FLOPs Breakdown and Crossover Analysis

Compute Distribution: FLOPs Breakdown and Attention/FFN Crossover



FLOPs FORMULAS (per block, fwd)

Config: Llama 3 8B
 $d=4096, h=32, h_{kv}=8$
 $d_k=128, d_{ff}=14336$

Attn projections (linear in L):
 Q: $2*B*L*d^2 = 2*L*4096^2$
 K: $2*B*L*d*h_{kv}*d_k = 2*L*4096*1024$
 V: same as K
 O: $2*B*L*d^2$

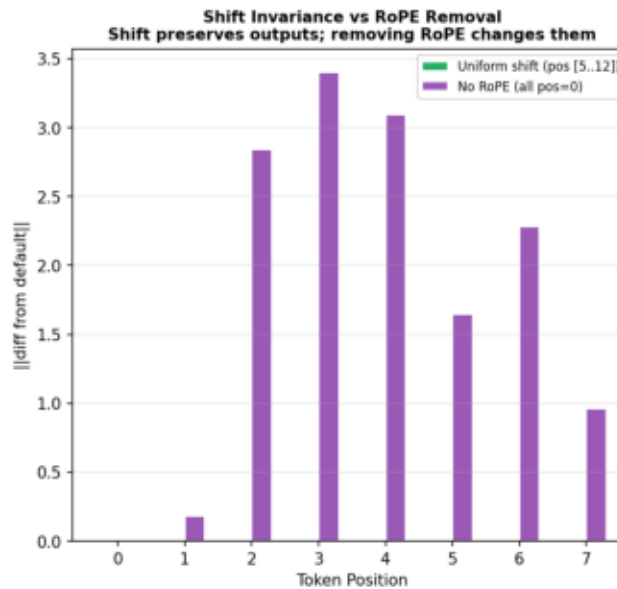
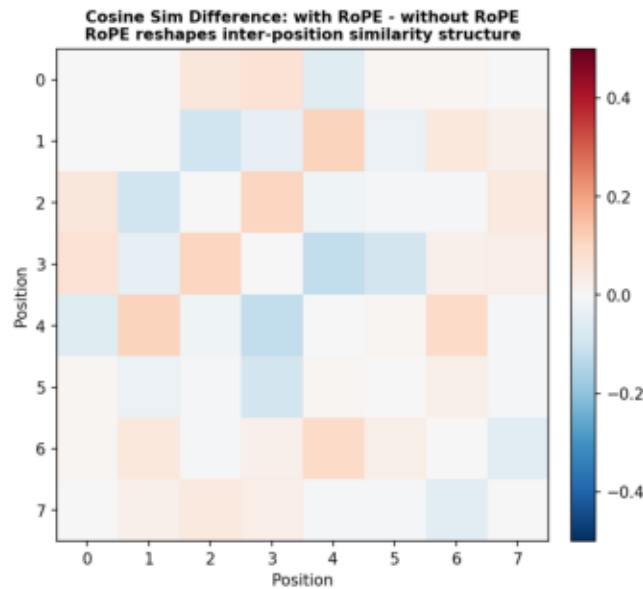
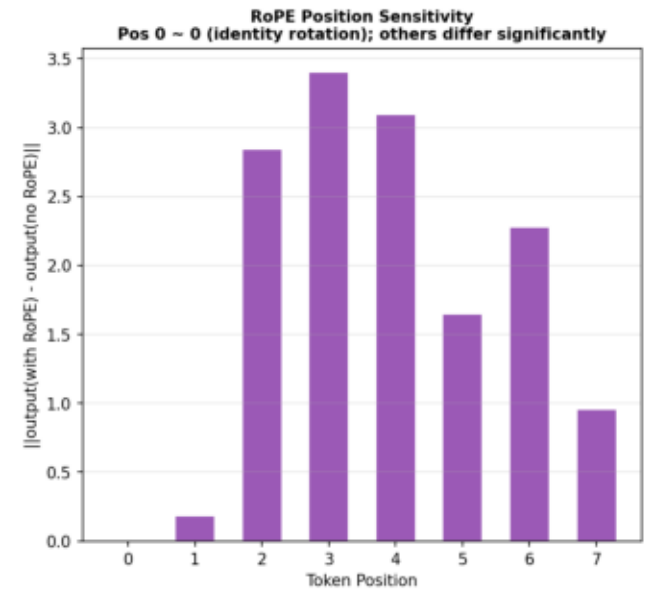
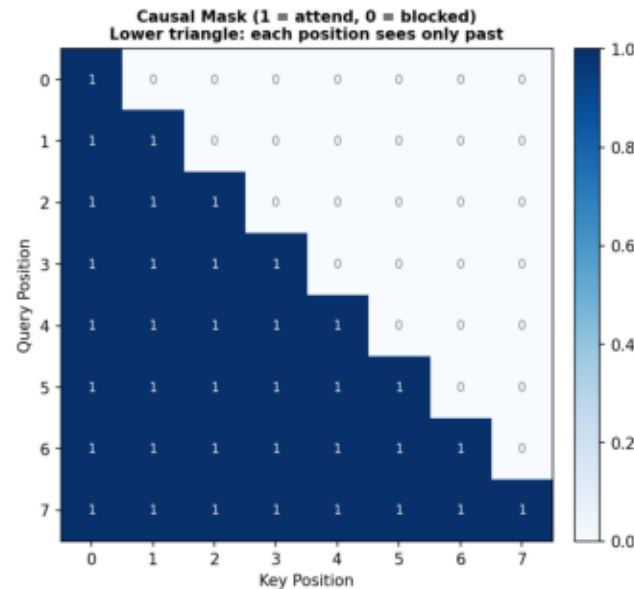
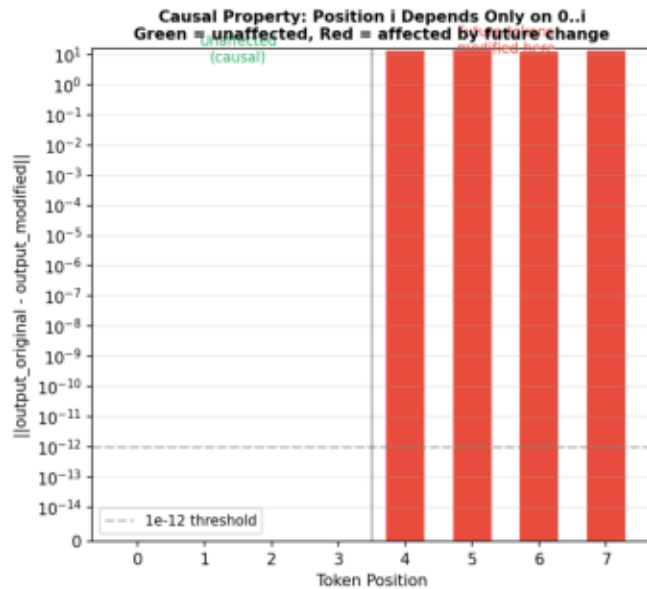
Attn core (QUADRATIC in L):
 QK^T: $2*B*h*L^2*d_k$
 softmax: $5*B*h*L^2$
 AV: $2*B*h*L^2*d_k$

FFN SwiGLU (linear in L):
 gate+up+down: $6*B*L*d*d_{ff}$

CROSSOVER (core = FFN):
 $L = 6*d*d_{ff} / ((4*d_k+5)*h)$
 $L = 21296$ tokens

Example 6: Causal Masking and Position Sensitivity

Causal Masking and Position Sensitivity: Autoregressive Behavior + RoPE



CAUSAL MASKING + RoPE

=====

Causal mask ensures:
output[1] depends only on input[0..1]
Verified: changing tokens at pos 4..7
does NOT affect output at pos 0..3

RoPE position awareness:
RoPE at pos 0 is identity (no rotation)
Other positions rotate Q/K, changing
attention score distribution.
Mean diff (RoPE vs no-RoPE): 1.8008

Shift invariance (relative pos. prop.):
Identical tokens + uniform shift
preserves all relative positions.
Mean diff: 2.77e-15 (~0)

Single token vs full sequence:
pos 0 diff = 4.33e-15
(Confirms pos 0 only sees itself)

CAVEAT: With random weights, position
effects are noise-like. Trained models
learn meaningful position-dependent
attention patterns.