Attention Residuals: A Drop-In Fix for How Every LLM Stacks Its Layers

Kabui, Charles

2026-03-18

Read at ToKnow.ai

Moonshot AI's Kimi team identified a flaw baked into every modern LLM: residual connections add each layer's output with a fixed weight of one. As layers stack deeper, this uniform accumulation causes hidden states to grow without bound, diluting what each individual layer contributes. Their fix, Attention Residuals (AttnRes), replaces that fixed addition with softmax attention over all preceding layer outputs. Each layer learns input-dependent weights

that control how much it draws from earlier representations. The practical version, Block AttnRes, groups layers into blocks and only attends across block boundaries, cutting memory overhead to near zero. Integrated into the Kimi Linear architecture (48B total parameters, 3B active) and pre-trained on 1.4 trillion tokens, AttnRes improved GPQA-Diamond by 7.5 points (36.9 to 44.4), HumanEval by 3.1 points, and MATH by 3.6 points over the standard residual baseline. Scaling law experiments confirmed consistent gains across model sizes.

Residual connections have been essentially unchanged since ResNet in 2015. They were carried into Transformers as-is. AttnRes is a drop-in replacement that requires no changes to training pipelines or inference infrastructure. Block AttnRes matches the loss of a baseline trained with 1.25x more compute, which means existing models could get "free" performance just by swapping their residual wiring.

If this holds up across more architectures, it could become standard in next-generation LLMs. The fact that such a fundamental component went unquestioned for a decade, and that a relatively simple fix yields broad gains, suggests there may be more low-hanging architectural improvements hiding in plain sight.

Sources:

- Attention Residuals Paper (arXiv)
- GitHub Repository (MoonshotAI)
- ResNet: Deep Residual Learning (He et al., 2015)
