What Makes a Layer Important?
A Cross-Architecture Analysis of Attention Entropy

Anonymous ACL submission

Abstract

Can a simple, cheap metric like attention entropy tell us which transformer layers we can safely skip? This question matters for any approach that does dynamic or conditional computation at inference time. Rather than speculate, we run the experiment—on both GPT-2 Small (124M, causal attention) and BERT-base (110M, bidirectional attention) across five NLP tasks, with exhaustive single-layer ablation to establish ground-truth importance scores. Across both architectures, the answer is consistently negative: attention entropy shows no reliable predictive relationship with layer importance. On GPT-2, the Pearson correlation between average attention entropy and ablation-induced perplexity increase is $r = 0.37$ ($p = 0.23$), falling to $r = -0.15$ ($p = 0.66$) when the outlier first layer is excluded. On BERT, the correlation is negligible: $r = 0.02$ ($p = 0.95$). We caution that all correlation analyses involve only 12 data points (one per layer), which limits statistical power; these results should be interpreted as suggestive evidence rather than definitive proof. We also evaluate two alternative signals. Hidden state norm shows a moderate rank correlation with importance on GPT-2 (Spearman $\rho = 0.58$, $p = 0.06$) but is near-constant and uninformative on BERT due to layer normalization. Gradient magnitude, a more expensive but still tractable signal, achieves Pearson $r = 0.95$ on BERT, but this is almost entirely driven by a single extreme layer (Spearman $\rho = 0.11$, $p = 0.73$). Our findings suggest that practical layer-skipping strategies require richer, possibly multi-signal approaches rather than relying on any single cheap metric. We release all data, code, and 100 prompts.

1 Introduction

Say you have a transformer model running in production, and you'd like to make it faster. One tempting idea: skip layers whose attention patterns look "unfocused." The intuition is appealing—a layer with sharp, concentrated attention must be doing something important, while a layer with flat, diffuse attention is probably just passing information through. And the beauty of it is that attention entropy is practically free to compute at inference time. If this heuristic works, dynamic transformer inference becomes much simpler.

The idea is not new. Attention entropy has been used as a descriptive tool in a long line of work on transformer interpretability (Clark et al., 2019; Kovaleva et al., 2019; Voita et al., 2019), and implicitly or explicitly motivates several proposals for dynamic computation (Schuster et al., 2022; Goyal et al., 2020). But here is what hasn't been done: a direct, quantitative test of whether attention entropy actually predicts which layers a model needs to keep.

Our contribution is that test. We subject two architecturally different transformers—GPT-2 Small (Radford et al., 2019) with causal attention and BERT-base (Devlin et al., 2019) with bidirectional attention—to the same three-part measurement. First, we compute per-layer attention entropy across five NLP tasks: summarization, natural language inference, question answering, machine translation, and text classification, with 20 prompts per task. Second, we establish ground-truth importance via single-layer ablation: remove each layer, measure the perplexity hit, and treat that as the true importance score. Third, we compare entropy against two alternative signals: the L2 norm of hidden states and the gradient magnitude at each layer.

A quick summary of what we found. Attention entropy is strikingly consistent across tasks on GPT-2 (mean pairwise $r = 0.99$) and somewhat less so on BERT (mean $r = 0.95$). This confirms that entropy largely reflects model-level structure. But when we compare entropy to actual layer importance, the correlation vanishes. For GPT-2 without the first layer, $r = -0.15$ ($p = 0.66$). For BERT,

$r = 0.02$ ($p = 0.95$). In plain terms: knowing a layer's attention entropy provides no reliable signal about whether the model needs it. We stress that these correlations involve only 12 data points per model, so the results should be read as indicative rather than conclusive—but the near-zero values are consistent across two architecturally distinct models. Hidden state norm does moderately better on GPT-2 (Spearman $\rho = 0.58$) but is essentially constant on BERT due to internal layer normalization, rendering it useless. Gradient magnitude achieves a strong Pearson correlation on BERT ($r = 0.95$) but this is entirely driven by a single extreme layer; the Spearman rank correlation is only $\rho = 0.11$. Across both models and all three signals, no single cheap metric reliably tracks layer importance.

We do not propose a new method. This is a measurement study. We think the community will benefit from knowing that a popular heuristic fails when tested rigorously, and from understanding the shape of the actual problem.

2 Related Work

2.1 Attention analysis

Since the introduction of the transformer (Vaswani et al., 2017), researchers have analyzed what attention patterns encode. Clark et al. (2019) showed individual BERT heads correspond to syntactic relations. Kovaleva et al. (2019) catalogued common attention patterns. Vig and Belinkov (2019) visualized how attention changes across layers. Ghader and Monz (2017) examined attention in machine translation. Our work differs in that we ask a predictive rather than descriptive question: does attention tell us what to compute?

2.2 Dynamic computation

A growing literature on adaptive inference explores ways to make transformers cheaper at runtime. Early exiting (Schuster et al., 2022; Xin et al., 2020) lets confident predictions leave the model before reaching the final layer. Token pruning (Goyal et al., 2020; Kim & Cho, 2021) removes tokens mid-network. LayerDrop (Fan et al., 2020) trains models to tolerate layer removal. Deja Vu (Liu et al., 2023) uses contextual sparsity. These methods all face a common challenge: deciding what to compute and what to skip. Our work examines one candidate decision rule.

2.3 Layer importance

Layer importance has been studied through probing (Tenney et al., 2019; Jawahar et al., 2019), ablation (Sajjad et al., 2023), and attribution (Hao et al., 2021; Sundararajan et al., 2017). The closest work to ours is Sajjad et al. (2023), who use single-layer dropping on BERT-family models and show that middle layers are most redundant. However, their study treats ablation-based importance as a final answer—a way to identify which layers to drop for model compression—and does not ask whether any cheap inference-time signal can predict the ablation ranking without running ablation itself. This predictive question is the central contribution of our work. A positive answer would enable dynamic, input-conditioned layer skipping at no offline cost; a negative answer (as we find) tells the community that the compression and efficient inference literatures should not assume that low-cost attention-based proxies can substitute for more expensive analysis. In addition, while Sajjad et al. evaluate only BERT-family encoders, we include GPT-2 as a causal decoder, allowing us to compare how layer importance patterns and proxy signal behavior differ across the two dominant attention paradigms.

3 Experimental Setup

3.1 Models

We use two architecturally distinct models to test whether our findings generalize across attention mechanisms. GPT-2 Small (Radford et al., 2019) uses causal (unidirectional) self-attention with 12 layers, 12 heads, and a hidden dimension of 768 (124M parameters). BERT-base uncased (Devlin et al., 2019) uses bidirectional self-attention, also with 12 layers, 12 heads, and a hidden dimension of 768 (110M parameters). Both models run entirely on CPU to ensure reproducibility without GPU requirements.

3.2 Tasks and prompts

We select five tasks spanning diverse NLP requirements: summarization (CNN/DailyMail; Hermann et al., 2015), natural language inference (MultiNLI; Williams et al., 2018), question answering (SQuAD v2.0; Rajpurkar et al., 2018), machine translation (WMT14 En-De; Bojar et al., 2014), and text classification (AG News; Zhang et al., 2015). For each task, we manually selected 20 prompts from the respective datasets, varying in topic, length, and

complexity. All 100 prompts are released with our code.

3.3 Entropy computation

For each layer l and head h, we compute attention entropy $\mathrm{H}(1, \mathrm{h}) = -\Sigma_{\mathrm{i}}\, \mathrm{a_i} \log(\mathrm{a_i})$ on the attention weights α_i over all tokens attended to. Per-layer entropy is the mean over all 12 heads, averaged across all 20 prompts. For GPT-2, we compute entropy on the last token of each sequence. This choice is motivated by two considerations. First, in a causal (left-to-right) model, the last token has attended over the longest available context and therefore produces the most informative attention distribution; earlier tokens have shorter visible histories, making their entropy values structurally lower and less representative of the layer's general behavior. Second, for language generation tasks the last token's representation directly determines the next-token prediction, so its attention pattern is most relevant to the model's functional output. For BERT, we use the [CLS] token's attention pattern. The [CLS] token is designed to aggregate sequence-level information and is the canonical query for classification and sequence-level tasks. While per-token averaging is a defensible alternative, it introduces noise from tokens with highly variable local attention patterns (e.g., punctuation or stop words), and the [CLS] representation is the standard summary token for downstream use. We acknowledge that neither choice is uniquely optimal, and in principle averaging over all tokens could capture a more comprehensive view of each layer's behavior; we chose single-token summaries to keep the entropy computation lightweight and reproducible, consistent with the study's focus on cheap inference-time signals.

3.4 Ablation protocol

To establish ground-truth importance, we replace each layer individually with an identity function that passes hidden states through unchanged. For GPT-2, we load GPT2Model with the lm_{head} weight matrix for autoregressive perplexity. For BERT, we load BertModel with the MLM prediction head for masked language model perplexity. The ablation impact $\Delta\mathrm{PPL}(\mathrm{l}) = \mathrm{PPL}_{\mathrm{ablated}}(\mathrm{l}) - \mathrm{PPL}_{\mathrm{baseline}}$ serves as our importance score. Higher values mean the layer matters more; negative values indicate that removing the layer improves performance.

3.5 Alternative signals

We evaluate two alternatives to entropy. Hidden state norm $\mathrm{N}(1) = \|\mathrm{H_l}\|_2$ is the L2 norm of each layer's output, averaged over tokens and prompts. Gradient norm $G(l)$ captures the magnitude of the gradient flowing through each layer with respect to the final output, computed via backward hooks on a summed hidden state loss. Neither signal requires labeled data, though gradient norm requires a backward pass.

4 Results

4.1 Entropy is consistent across tasks, but less so for bidirectional attention

Figure 1 shows per-layer attention entropy for all five tasks on both models. On GPT-2, the curves are nearly indistinguishable: the mean pairwise Pearson correlation across all 10 task pairs is $r = 0.99$, with every pair exceeding $r = 0.98$. This high consistency confirms that attention entropy is primarily determined by the model architecture rather than the specific input.

On BERT, the story is different. While entropy patterns still correlate across tasks, the relationship is notably weaker: mean pairwise $r = 0.95$. This gap makes intuitive sense. Bidirectional attention allows every token to attend to every other token regardless of position, making the attention pattern inherently more sensitive to the specific linguistic content of the input. Causal attention, by contrast, imposes a rigid positional prior that dominates the entropy pattern.

4.2 Layer importance shows drastically different patterns across architectures

Figure 2 shows ablation results for both models. The baseline perplexity is 36 for GPT-2 (autoregressive) and 5787 for BERT (MLM). These baselines are not directly comparable since they measure different objectives, but the relative patterns within each model are informative.

GPT-2 follows a clean U-shaped curve. Layer 0, which processes raw token embeddings, is by far the most important: removing it spikes PPL by +6423. Middle layers (1-9) are partially redundant, with removal costs ranging from 2.7 to 6.6. The final layers 10 and 11 regain importance ($\Delta = +14$ and $+16$ respectively), consistent with their role in output integration.

BERT paints a fundamentally different picture. Layer 11, closest to the MLM output head, is ex-

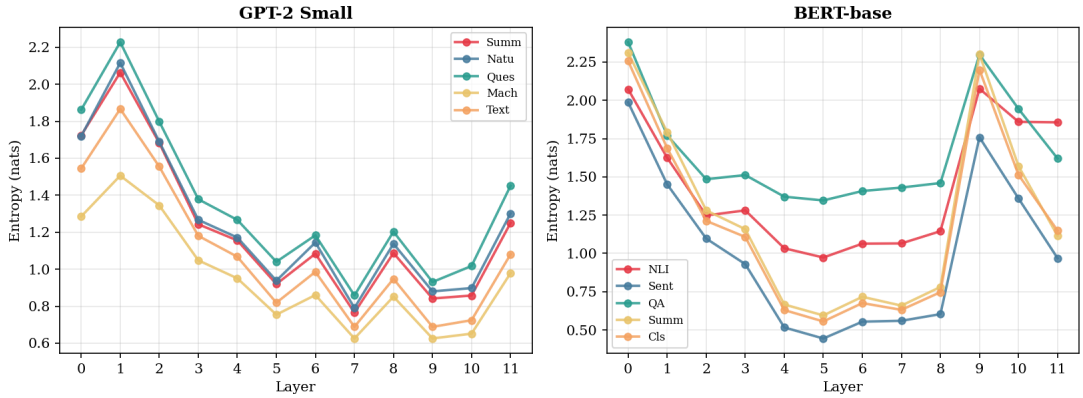

Figure 1: Per-layer attention entropy GPT-2 vs BERT.

treme: removing it explodes PPL to over 1.2e+06 ($\Delta > +10^6$). But several other layers actually hurt performance when present: removing layers 1, 3, 4, 6, and 10 each reduces PPL (negative Δ). The largest improvement comes from removing layer 3, which drops PPL by 2151 (37% relative). This is a striking finding: some BERT layers are not just redundant but actively harmful for MLM prediction on these prompts. We note that this observation likely reflects the interaction between our specific evaluation prompts and the MLM objective, rather than a universal property of BERT layers.

4.3 Entropy shows near-zero correlation with layer importance on both architectures

Now the central question: does entropy predict importance? Figure 3 plots the relationship.

On GPT-2, with all layers included, the Pearson correlation between average entropy and ablation impact is $r = 0.37$ ($p = 0.23$). This weak correlation is almost entirely driven by layer 0, which happens to have both moderately high entropy and enormous ablation impact. Excluding layer 0, the correlation collapses to $r = -0.15$ ($p = 0.66$)—essentially zero, and slightly negative, meaning higher entropy actually trends toward lower importance (the opposite of the intuitive story).

BERT is even worse. The overall correlation is $r = 0.02$ ($p = 0.95$). Even after removing layer 11—the extreme outlier—the correlation remains negligible ($r = 0.08$, $p = 0.81$). On both architectures, in both attention paradigms, attention entropy shows no consistent signal about a layer's importance. We acknowledge that correlation analyses over 12 data points are inherently low-powered, and the large p-values already reflect this. The consistent pattern across two architecturally different models—both yielding near-zero Spearman rank correlations—strengthens our confidence that the negative result is genuine, but we encourage replication on larger models before treating it as definitive. Future work should complement these point estimates with bootstrap confidence intervals to better characterize uncertainty.

4.4 Alternative signals offer limited improvement

Given entropy's failure, we evaluate two alternative cheap signals. Figure 4 (left) shows gradient norm vs ablation impact on BERT. The Pearson correlation is a deceptively strong $r = 0.95$ (p ≈ 0). However, this is entirely driven by layer 11, which has both the highest gradient norm (138) and the largest ablation impact. The Spearman rank correlation, which is robust to outliers, is merely $\rho = 0.11$ ($p = 0.73$). Gradient norm's high Pearson r is a statistical artifact of the final layer's dominance, not evidence of a genuine relationship across all layers. We also computed gradient norm for GPT-2 but omit its scatter plot for space; the pattern parallels BERT—a high Pearson r dominated by the first-layer outlier (layer 0), with Spearman ρ falling to near zero once that outlier is excluded—confirming the artifact is not architecture-specific.

Hidden state norm on GPT-2 (Figure 4, right) does better than entropy: Spearman $\rho = 0.58$ ($p = 0.06$) without layer 0. This is a borderline result—suggestive but not reliable. Norm increases roughly monotonically with depth, which aligns with the "deeper layers are more important" half of the U-shaped importance curve but misses the early-layer spike. On BERT, hidden state norm is essentially constant (values range from 0.5 to 0.8) due to internal layer normalization, making

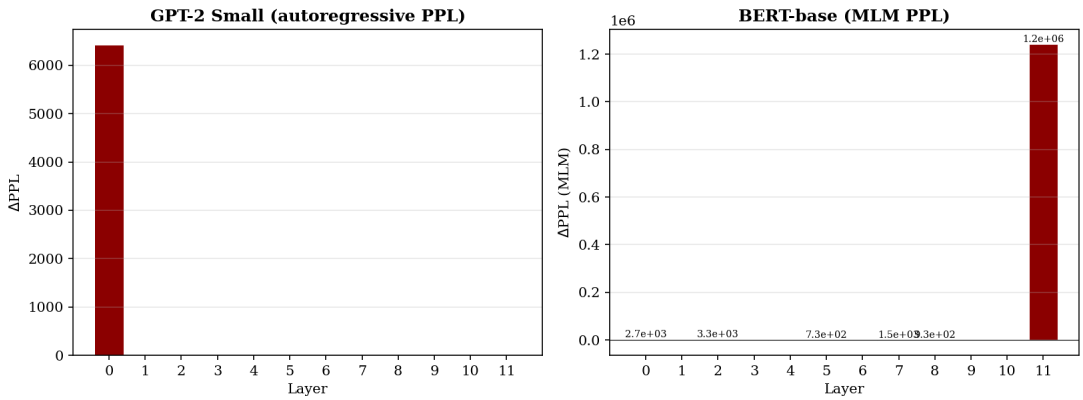

Figure 2: Single-layer ablation impact—GPT-2 vs BERT.

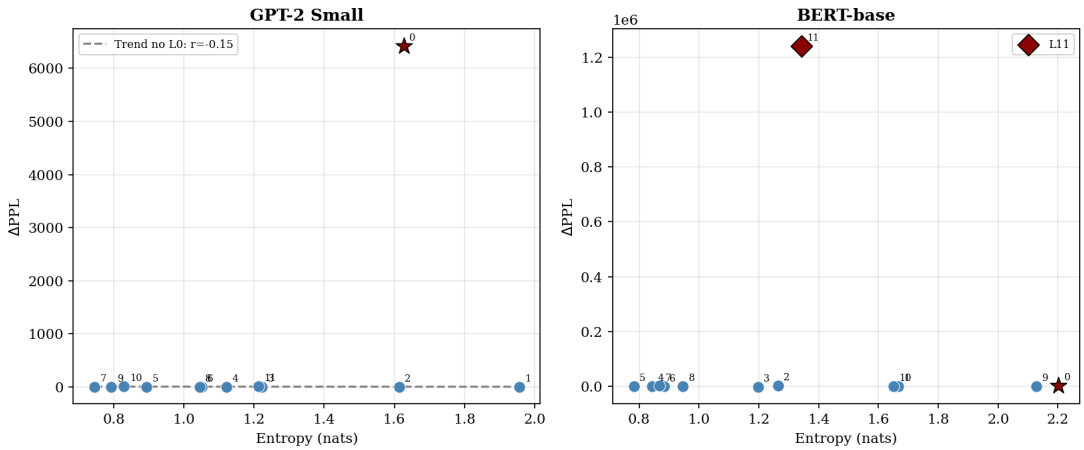

Figure 3: Attention entroy vs ablation impact.

it completely uninformative. The divergence between the two models is explained by architectural differences in layer normalization placement: GPT-2 uses pre-layer normalization (LN applied before each sublayer), which allows residual stream norms to grow with depth; BERT uses post-layer normalization (LN applied after each sublayer addition), which actively constrains output norms to remain near-constant across layers. This architectural detail entirely explains why norm is a viable (if imperfect) signal on one model and useless on the other.

4.5 Cross-task entropy stability differs by architecture

Figure 5 compares the distribution of cross-task entropy correlations for both models. GPT-2's correlations cluster tightly around $r = 0.99$ (range: [0.98, 1.00]), reflecting the structural dominance of causal masking. BERT's correlations span a much wider range (0.90-0.999), confirming that bidirectional attention allows task-specific patterns to emerge.

5 Discussion

Our results provide two complementary messages. The negative message is clear: neither attention entropy ($r \leq 0.15$ on both models) nor hidden state norm nor gradient magnitude reliably predicts layer importance. If you are building a system that decides which layers to skip, none of these cheap heuristics will serve as an adequate decision rule.

The positive message, such as it is, is that we now know the shape of the problem. Layer importance is not monotonic, not smooth, and not captured by any single low-level signal we tested. The fact that some BERT layers actively degrade MLM performance when present is particularly striking and suggests that the right approach may involve input-dependent predictions rather than static per-layer importance scores.

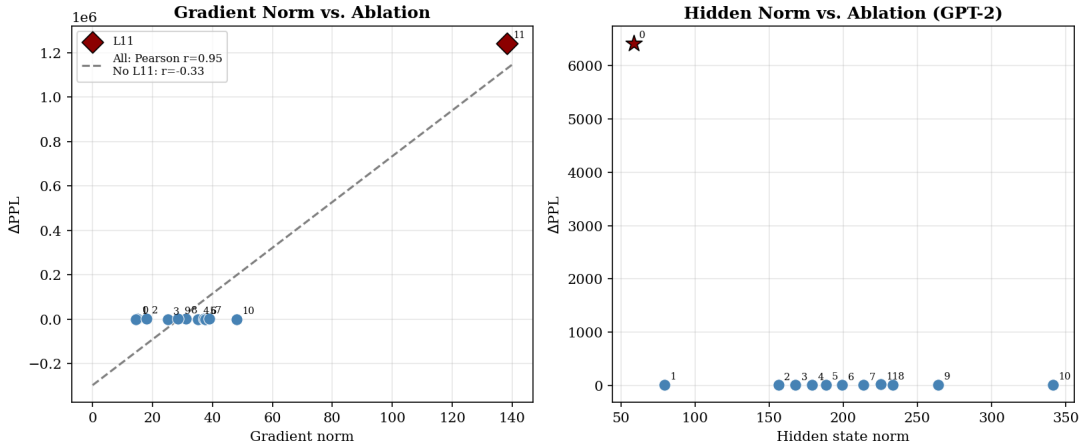

Figure 4: Alternative importance signals.

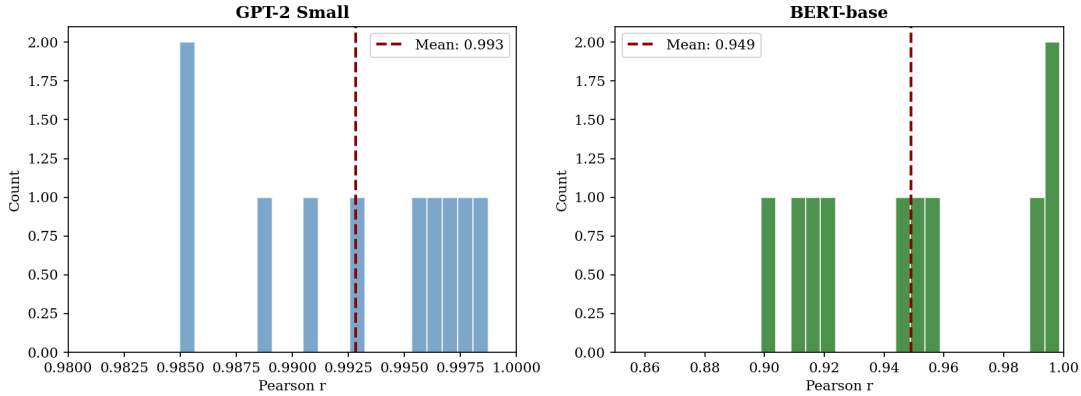

Figure 5: Cross-task entropy correlation distribution.

5.1 Why entropy fails

We see three reasons. First, entropy measures attention dispersion, not computational contribution. A layer with diffuse attention might be performing crucial contextual integration through its MLP sublayer, while a layer with focused attention might be stuck in a local pattern. Entropy simply does not distinguish these cases. Second, residual connections (Elhage et al., 2021) allow information to bypass attention entirely, decoupling attention behavior from computational importance. Third, the importance curve is fundamentally non-monotonic (U-shaped for GPT-2, spikey for BERT), while entropy trends are approximately decreasing. A shape mismatch is fatal for correlation.

5.2 Toward better importance prediction

Our results suggest two concrete directions. The first is multi-signal fusion. No single signal—entropy, hidden state norm, or gradient magnitude—reliably tracks layer importance across both architectures. However, these signals capture partially orthogonal aspects of layer behavior: entropy reflects attention dispersion, norm reflects residual magnitude, and gradient reflects sensitivity to output loss. A learned combination of these signals, potentially trained on a small labeled set of ablation-measured importances, might capture the non-monotonic importance pattern in a way that no individual signal can. The second is input-dependent prediction. Our results show that layer importance is not static: the BERT finding that some layers are counter-productive for specific prompt types suggests that importance is at least partially conditional on the input. Rather than assigning a fixed per-layer score, a lightweight gating network—taking the current hidden state as input and predicting a skip probability—could make layer execution conditional in a way that adapts to each sequence. Early-exit literature (Schuster et al., 2022; Xin et al., 2020) pursues this intuition at the sequence level; a layer-level analog remains underexplored. A third direction, beyond the scope of this paper but important for any follow-up, is

grounding importance in downstream task performance rather than perplexity. Our ablation-based importance scores use language modeling loss as a proxy, which may not fully capture what a layer contributes to NLI, QA, or summarization. Future work should validate whether perplexity-based rankings correlate with task-specific rankings on GLUE or SuperGLUE benchmarks, which would either strengthen or qualify the generalizability of findings like ours.

5.3 Limitations

Several limitations of this study should be noted. First, scope of models. We only tested two models of similar size (12 layers, 120M parameters), both released in 2019. Larger or more recent models—such as Llama-2-7B (32 layers), Mistral-7B, or modern BERT variants like DeBERTa—may exhibit qualitatively different layer importance patterns; our findings should not be generalized beyond small-scale encoder and decoder models without additional empirical verification. Second, ablation methodology. Our ablation method replaces each layer with an identity function, which disrupts the residual stream statistics that the subsequent layer normalization was trained to operate on. This is particularly concerning for the BERT results, where several layers appear to actively degrade MLM performance when present—an observation that may partly reflect identity-replacement artifacts rather than true layer counter-productivity. Alternative ablation methods (weight zeroing, random noise injection, or Taylor-expansion-based importance estimation) could yield different importance rankings and should be explored in future work. Third, evaluation metric. We use perplexity (autoregressive for GPT-2, MLM for BERT) as our proxy for task performance. Perplexity is not equivalent to downstream task performance; a layer that is unimportant for language modeling loss may nonetheless be critical for, e.g., NLI or QA. Validating our ablation-based importance scores against downstream GLUE or SuperGLUE performance would substantially strengthen the ground-truth claim. Fourth, statistical power. All correlation analyses involve 12 data points (one per layer). This severely limits statistical power, and the reported p-values reflect this. Results should be interpreted as consistent indicative evidence rather than statistically conclusive findings; bootstrap resampling or permutation tests would provide more rigorous uncertainty estimates. Fifth, prompt scale and diversity. Our evaluation prompts (20 per task for entropy, 8 for ablation per task) are hand-selected and limited in number. While we release all prompts for reproducibility, the finding that some BERT layers improve performance when removed is likely sensitive to both the specific prompt selection and the MLM evaluation objective.

6 Conclusion

We provide the first direct, two-model test of whether attention entropy predicts layer importance in transformer language models. The answer is consistently negative across both architectures: it does not. On GPT-2 Small (causal attention) without the first layer, entropy versus ablation correlation is $r = -0.15$ ($p = 0.66$). On BERT-base (bidirectional attention), the correlation is $r = 0.02$ ($p = 0.95$). We acknowledge that both estimates rest on 12 data points per model, which limits definitive conclusions; the consistent near-zero values across two architecturally distinct models nonetheless constitute meaningful negative evidence. Two alternative signals—hidden state norm and gradient magnitude—show limited promise but suffer from fundamental limitations: norm growth is neutralized by post-layer normalization in BERT, and gradient magnitude's apparent strength is a single-outlier artifact driven by the final layer. We also report that attention entropy is markedly more task-dependent in bidirectional transformers than in causal ones, and that some BERT layers appear actively counterproductive for MLM on our evaluation prompts—a finding that warrants further investigation, particularly given potential identity-ablation artifacts. We hope these empirical findings encourage the community to develop richer, possibly multi-signal or learned approaches to the layer importance prediction problem, rather than relying on intuitive but empirically unsupported heuristics.

References

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In NeurIPS.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In NAACL.

Schuster, T., Fisch, A., Gupta, J., Dehghani, M., Bahri, D., Tran, V. Q., Tay, Y., & Metzler, D. (2022). Confident adaptive language modeling. In NeurIPS.

Goyal, S., Choudhury, A. R., Raje, S., Chakaravarthy, V. T., Sabharwal, Y., & Verma, A. (2020). PoWER-BERT: Accelerating BERT inference via progressive word-vector elimination. In ICML.

Fan, A., Grave, E., & Joulin, A. (2020). Reducing transformer depth on demand with structured dropout. In ICLR.

Clark, K., Khandelwal, U., Levy, O., & Manning, C. D. (2019). What does BERT look at? An analysis of BERT's attention. In BlackboxNLP @ ACL.

Vig, J., & Belinkov, Y. (2019). Analyzing the structure of attention in a transformer language model. In BlackboxNLP @ ACL.

Kovaleva, O., Romanov, A., Rogers, A., & Rumshisky, A. (2019). Revealing the dark secrets of BERT. In EMNLP.

Tenney, I., Das, D., & Pavlick, E. (2019). BERT rediscovers the classical NLP pipeline. In ACL.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Technical Report.

Ghader, H., & Monz, C. (2017). What does attention in neural machine translation pay attention to? In IJCNLP.

Voita, E., Talbot, D., Moiseev, F., Sennrich, R., & Titov, I. (2019). Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In ACL.

Xin, J., Tang, R., Lee, J., Yu, Y., & Lin, J. (2020). DeeBERT: Dynamic early exiting for accelerating BERT inference. In ACL.

Kim, G., & Cho, K. (2021). Length-adaptive transformer: Train once with length drop, use anytime with search. In ACL.

Sajjad, H., Dalvi, F., Durrani, N., & Nakov, P. (2023). On the effect of dropping layers of pre-trained transformer models. Computer Speech & Language.

Jawahar, G., Sagot, B., & Seddah, D. (2019). What does BERT learn about the structure of language? In ACL.

Hao, Y., Dong, L., Wei, F., & Xu, K. (2021). Self-attention attribution: Interpreting information interactions inside transformer. In AAAI.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P. (2015). Teaching machines to read and comprehend. In NeurIPS.

Williams, A., Nangia, N., & Bowman, S. R. (2018). A broad-coverage challenge corpus for sentence understanding through inference. In NAACL.

Rajpurkar, P., Jia, R., & Liang, P. (2018). Know what you don't know: Unanswerable questions for SQuAD. In ACL.

Bojar, O., et al. (2014). Findings of the 2014 workshop on statistical machine translation. In WMT.

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In NeurIPS.

Elhage, N., et al. (2021). A mathematical framework for transformer circuits. Transformer Circuits Thread.

Liu, Z., et al. (2023). Deja Vu: Contextual sparsity for efficient LLMs at inference time. In ICML.

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In ICML.