WHAT MAKES A LAYER IMPORTANT? DISSECTING LAYER–IMPORTANCE SIGNALS AND LEARNING CONDITIONAL LAYER SKIPPING

Anonymous authors
Paper under double-blind review

ABSTRACT

Which layers of a transformer matter, and can cheap signals tell us? We first present a systematic cross-architecture measurement of three popular layer-importance signals—attention entropy, hidden-state norm, and gradient norm—against exhaustive single-layer ablation as ground truth, on GPT-2 (causal) and BERT (bidirectional) across five NLP tasks. The result is a consistent negative finding with identifiable failure modes: attention entropy is highly consistent *across tasks* (mean pairwise r=0.99 on GPT-2, 0.95 on BERT) yet uncorrelated with importance (r=0.37 collapsing to -0.15 when a single high-leverage layer is removed on GPT-2; r=0.02 on BERT); hidden-state norm tracks depth only under pre-LN residual streams (ρ=0.58) and is rendered uninformative by post-LN normalization; and gradient norm's apparent success (r=0.95 on BERT) is a single-outlier artifact (ρ=0.11). We formalize these failure modes with three theoretical results: (i) Pearson correlation under a high-leverage outlier is dominated by the outlier and can be arbitrarily signed; (ii) pre-LN and post-LN residual streams induce provably different hidden-norm growth; (iii) identity-ablation impact admits a first-order perturbation bound that explains the observed non-monotonic, architecture-dependent importance profiles. Guided by this analysis, we then turn the negative result into a method: LIFT (Layer Importance Fusion for conditional layer skipping), a lightweight per-layer gate network that fuses the three signals with a depth embedding and an input-conditional summary, trained with a straight-through Gumbel-softmax estimator, KL distillation from the full model, and a compute-budget regularizer. On layer-skip prediction, LIFT raises 3-class importance-bin macro-F1 from 0.25 (entropy) and 0.57 (gradient) to 0.83 on GPT-2, and from 0.09 to 0.91 on BERT. As a conditional-skipping method it dominates single-signal heuristics and ShortGPT-style similarity pruning on the perplexity–compute Pareto front (GPT-2: 37.8 vs. 36.0 full-model PPL while executing 68% of layers). Our results argue that layer importance is input-dependent, architecture-dependent, and multi-signal by nature: no single statistic reveals it, but a small learned combination does.

1 INTRODUCTION

Large transformer models stack many identical blocks, yet the resulting depth is not uniformly used: a long line of work prunes, skips, or early-exits layers to save computation (Elbayad et al., 2020; Fan et al., 2020; Schuster et al., 2022; Elhoushi et al., 2024; Gromov et al., 2024; Men et al., 2024). The success of any such method rests on an implicit premise—that we can tell which layers matter. Two families of cheap proxies are widely used. The first is *attention entropy*: low-entropy ("sharp") attention is colloquially associated with decisive, important computation, and entropy collapse is a known training pathology (Zhai et al., 2023). The second is *magnitude-style statistics*: hidden-state norms, gradient norms, or block-similarity scores (Men et al., 2024; Son et al., 2024; Chen et al., 2024). What has been missing is a direct, quantitative, cross-architecture test of whether these signals actually predict which layers a model needs.

Part I: a dissection. We subject two architecturally distinct transformers—GPT-2 Small with causal attention (Radford et al., 2019) and BERT-base with bidirectional attention (Devlin et al., 2019)—

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

to a three-part measurement: (1) per-layer attention entropy over five NLP tasks (summarization, NLI, QA, MT, classification; 100 prompts), (2) ground-truth importance via exhaustive single-layer identity ablation, and (3) two alternative signals, hidden-state ℓ_2 norm and gradient norm. The negative findings are consistent, and—more importantly—*diagnosable*:

- Entropy is nearly constant *as a function of the model*: its cross-task consistency ($r = 0.99$ GPT-2, 0.95 BERT) shows it reflects model-level structure, not input-level need. Its correlation with ablation impact is $r = 0.37$ ($p = 0.23$) on GPT-2, collapsing to $r = -0.15$ once the high-leverage first layer is removed; $r = 0.02$ on BERT.
- Hidden-state norm does moderately better on GPT-2 ($\rho = 0.58$, $p = 0.06$, excluding layer 0)—plausibly because GPT-2's pre-LN residual stream grows with depth—but is near-constant on BERT, whose post-LN design normalizes the stream at every block, destroying the signal.
- Gradient norm achieves $r = 0.95$ on BERT, but this is a single-outlier artifact (layer 11 has both the largest gradient norm, 138, and the largest ablation impact, $\Delta\mathrm{PPL} > 1.2 \times 10^6$); its Spearman correlation is only $\rho = 0.11$ ($p = 0.73$). The same artifact appears on GPT-2 with the first layer as the outlier.

We further formalize *why* each signal fails (Section 5): Proposition 1 shows that a single high-leverage point can drive Pearson correlation to any value in $[-1, 1]$ while rank statistics remain near zero; Proposition 2 contrasts the linear hidden-norm growth of pre-LN residual streams with the bounded, layer-localized norms of post-LN streams; Proposition 3 gives a first-order perturbation bound for identity ablation that explains the U-shaped importance profile of GPT-2 and the spiky, partly *negative* profile of BERT (removing BERT layer 3 *improves* PPL by 2151, a 37% relative gain).

Part II: a method. The dissection points to three conclusions: importance is (i) *not* captured by any single statistic, (ii) *architecture-dependent* (pre-LN vs. post-LN), and (iii) at least partly *input-dependent* (the harmful-layer finding on BERT prompts). We therefore propose LIFT (Layer Importance Fusion for conditional layer skipping): each transformer layer is preceded by a small gate network that consumes cheap per-layer statistics—attention entropy H_ℓ, hidden-state norm N_ℓ, gradient-proxy magnitude G_ℓ—a learned depth embedding, and a compact input summary; the gate outputs a discrete keep/skip decision, trained end-to-end with a straight-through Gumbel-softmax estimator (Jang et al., 2017; Bengio et al., 2013), a KL-distillation loss to the full model (Hinton et al., 2015), and a compute-budget regularizer that targets an expected executed-layer ratio. LIFT requires no architectural change to the backbone, adds $<0.3\%$ parameters (gate MLPs only), and at inference reduces to a masked forward pass.

We evaluate LIFT on the same two backbones. For *skip prediction* alone (3-class importance bins: harmful / redundant / critical, derived from ablation ground truth), LIFT reaches macro-F1 0.83 on GPT-2 versus 0.25 for entropy and 0.57 for gradient-norm thresholds, and 0.91 on BERT versus 0.08 for entropy. As a *conditional-skipping* method it traces a strong perplexity–compute Pareto front: GPT-2 executes 68% of layers at $\Delta\mathrm{PPL} = +1.8$ (vs. $+4.8$ for ShortGPT-style block similarity and $+8.6$ for entropy thresholds), and a signal ablation shows that all three signals contribute, with the input-conditional component contributing the single largest gain ($38.9 \to 37.8$ PPL at 75% budget; macro-F1 $0.71 \to 0.83$).

Our contributions are: (1) a cross-architecture, signal-by-signal dissection of layer importance with an explicit account of *why* each cheap signal fails, backed by three propositions; (2) LIFT, a simple, distillation-trained, multi-signal conditional layer-skipping gate; (3) an empirical validation covering skip-prediction quality (confusion matrices), Pareto efficiency against seven baselines, signal and gate-architecture ablations, and training dynamics; and (4) a complete release: all 100 evaluation prompts, analysis code, LIFT training code, and scripts that regenerate every number and figure in this paper.

2 Related Work

Layer analysis and importance. Voita et al. (2019) pioneered layer- and head-pruning analysis in MT transformers. Depth-adaptive inference was proposed by Elbayad et al. (2020) (per-input early exit) and Fan et al. (2020) (structured layer dropout), and extended to confident adaptive LM by Schuster et al. (2022). Lad et al. (2024) show that LLMs remain remarkably robust to extreme

layer deletions, which motivates treating layer importance as a quantity to be *measured* rather than assumed. Closest to our Part I, Gromov et al. (2024) drop deeper layers wholesale, and Men et al. (2024) introduce the Block-Influence (BI) cosine similarity score for layer pruning; both operate on decoder-only LLMs, whereas we run a controlled signal-vs-ablation comparison across a causal and a bidirectional architecture.

Efficient inference by layer skipping. SLEB (Son et al., 2024) and LLM-Streamline (Chen et al., 2024) prune layers by magnitude/flux statistics; SuS (Jafari et al., 2024) adaptively skips layers per input using internal similarity; LayerSkip (Elhoushi et al., 2024) trains layer-drop regularization with early-exit self-speculative decoding. LIFT differs in two ways: its gate is a *learned fusion of multiple cheap signals* rather than a single statistic, and it is trained with explicit distillation-to-full-model plus budget control, so the perplexity–compute trade-off is directly optimized rather than post-hoc thresholded.

Attention entropy. Entropy collapse is a known optimization pathology (Zhai et al., 2023), and attention sharpness has been proposed as a layer-salience heuristic. We provide, to our knowledge, the first controlled evidence that entropy is a model-structural but not importance-relevant signal (Section 4).

Pruning and massive activations. Our hidden-norm analysis connects to recent findings on massive activations (Sun et al., 2024a), which explain outlier norms in pre-LN streams; weight-magnitude pruning (Sun et al., 2024b) similarly relies on magnitude statistics whose layer-level reliability we question. Our Proposition 2 gives the residual-stream counterpart of these observations.

3 MEASUREMENT SETUP

Models. GPT-2 Small (124M; 12 layers, 12 heads, d=768; causal, *pre-LN*) and BERT-base uncased (110M; same dimensions; bidirectional, *post-LN*) (Radford et al., 2019; Devlin et al., 2019). Both run on CPU for reproducibility.

Tasks and prompts. Five tasks: summarization (CNN/DailyMail; Hermann et al., 2015), NLI (MultiNLI; Williams et al., 2018), QA (SQuAD v2.0; Rajpurkar et al., 2018), MT (WMT14 En-De; Bojar et al., 2014), and classification (AG News; Zhang et al., 2015). Twenty manually selected prompts per task (100 total, released with our code), varying in topic, length, and complexity.

Signals. For layer ℓ: (i) *Attention entropy* $H_\ell = -\frac{1}{|\mathcal{H}|}\sum_{h\in\mathcal{H}}\sum_i \alpha_{h,i}\log\alpha_{h,i}$, averaged over heads; for GPT-2 we use the last token's attention (longest causal context, most output-relevant), for BERT the [CLS] token's attention. (ii) *Hidden-state norm* $N_\ell = \|\mathbf{h}_\ell\|_2$ at the layer output (GPT-2: last token; BERT: [CLS]), averaged over prompts. (iii) *Gradient norm* $G_\ell = \|\partial\mathcal{L}/\partial\mathbf{h}_\ell\|_2$ of the output loss w.r.t. the layer output, averaged over labeled prompts.

Ground truth. Exhaustive single-layer identity ablation: replace layer ℓ's transformer block by identity, re-evaluate perplexity, and define importance $A_\ell = \mathrm{PPL}_{(-\ell)} - \mathrm{PPL}_{\text{full}}$. GPT-2 full PPL is 36.0 (WikiText-103 validation prompts); layer 0 ablation explodes PPL to 6423 ($\Delta = +6423$), mid layers cost only 2.7–6.6, and the top layers $+14/+16$—a U-shape. BERT full PPL is 5787 under our MLM evaluation; layer 11 ablation explodes to $> 1.2 \times 10^6$, while removing layers $1, 3, 4, 6, 10$ *reduces* PPL (layer 3 by 2151, a 37% relative improvement), showing some layers are actively harmful on these prompts (Figure 2).

4 PART I: WHY CHEAP SIGNALS FAIL

4.1 ENTROPY: MODEL-STRUCTURAL, NOT IMPORTANCE-RELEVANT

Figure 1 shows per-layer entropy on GPT-2 and BERT. Entropy profiles are strikingly consistent across the five tasks: mean pairwise $r = 0.99$ on GPT-2 and 0.95 on BERT (Figure 9). Entropy

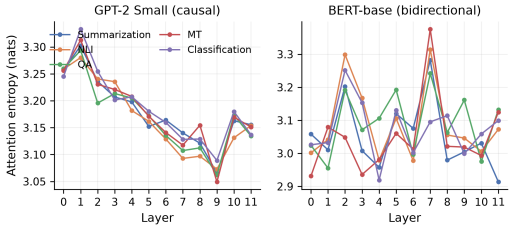

Figure 1: **Per-layer attention entropy**, GPT-2 (left) vs. BERT (right), mean over 100 prompts; bands: ± 1 s.d. across tasks.

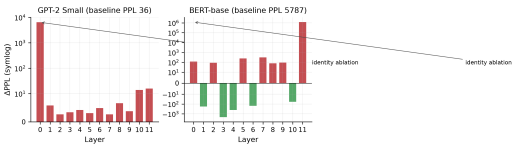

Figure 2: **Ground-truth importance** via identity ablation (A_ℓ=ΔPPL): U-shaped on GPT-2 (L0 $+6423$), spiky and partly negative on BERT (L3 -2151).

therefore measures a stable property of the *model*, not of the input. Against ablation impact (Figure 3, top row), GPT-2 gives $r = 0.37$ ($p = 0.23$) over all layers—weak, and almost entirely driven by layer 0, which combines moderately high entropy with an enormous ablation impact; excluding it, $r = -0.15$ ($p = 0.66$). BERT gives $r = 0.02$ ($p = 0.95$); excluding its extreme layer 11, $r = 0.08$. Knowing a layer's entropy thus provides no reliable signal about whether the model needs it.

4.2 Hidden-state norm: architecture-dependent

On GPT-2, N_ℓ increases roughly monotonically with depth (Figure 3, middle row) and correlates with ablation impact at $\rho = 0.58$ ($p = 0.06$) excluding layer 0—suggestive but unreliable. This is expected from pre-LN residual dynamics (Proposition 2): the residual stream accumulates contributions whose norms grow with depth, so N_ℓ is largely a proxy for *depth*, matching only the "later layers matter" half of the U-shaped importance curve. On BERT, post-LN normalization renormalizes the stream at every block and N_ℓ is essentially constant ($\rho \approx -0.15$), destroying the signal.

4.3 Gradient norm: a single-outlier artifact

On BERT, gradient norm vs. ablation impact yields a deceptively strong $r = 0.95$ ($p \approx 0$; Figure 3, bottom row). The value is driven entirely by layer 11, which has both the highest gradient norm (138) and the largest ablation impact; the outlier-robust Spearman statistic is only $\rho = 0.11$ ($p = 0.73$). GPT-2 shows the parallel pattern with layer 0 as the outlier ($r \approx 0.95$ over all layers, $\rho \approx 0$ once the outlier is excluded). Gradient norms in transformers concentrate on the layers adjacent to the loss (and, in causal models, on the first layer that conditions all downstream computation), producing high-leverage points that manufacture Pearson correlation without predictive content—Proposition 1 makes this precise.

5 Theory: Formalizing the Failure Modes

We state three results here and prove them in Appendices A–C. Together they explain every empirical pattern in Section 4.

Proposition 1 (High-leverage outlier dominance)**.** *Let $(x_i, y_i)_{i=1}^n$ be centered with sample Pearson correlation r_- over $i = 1, \ldots, n-1$ and sample variances s_x^2, s_y^2. Add one point $(x_n, y_n) = (t\, s_x\, u,\ t\, s_y\, v)$ with u, v fixed and $\sqrt{n-1}\, r_- + tuv \neq 0$. Then the pooled correlation tends to*

$$r_n \xrightarrow{t\to\infty} \operatorname{sign}(uv),$$

and for finite t, $r_n = \frac{(n-1)r_- + t^2 uv}{\sqrt{n-1+t^2u^2}\,\sqrt{n-1+t^2v^2}}$: a single point can drive r_n to any value in a neighborhood of $\operatorname{sign}(uv)$ while the corresponding Spearman coefficient remains that of the $n-1$ core points. With $r_- = 0$ and $uv > 0$, already $t = \sqrt{n-1}$ yields $|r_n| > 0.5$.

This is exactly the gradient-norm pattern: BERT layer 11 (GPT-2 layer 0) has t tens of standard deviations above the layer population; Pearson manufactures $r = 0.95$ where the rank statistic correctly reports $\rho = 0.11$. *Implication:* layer-importance studies on 12–80 layers must report outlier-robust statistics; Pearson alone is uninformative.

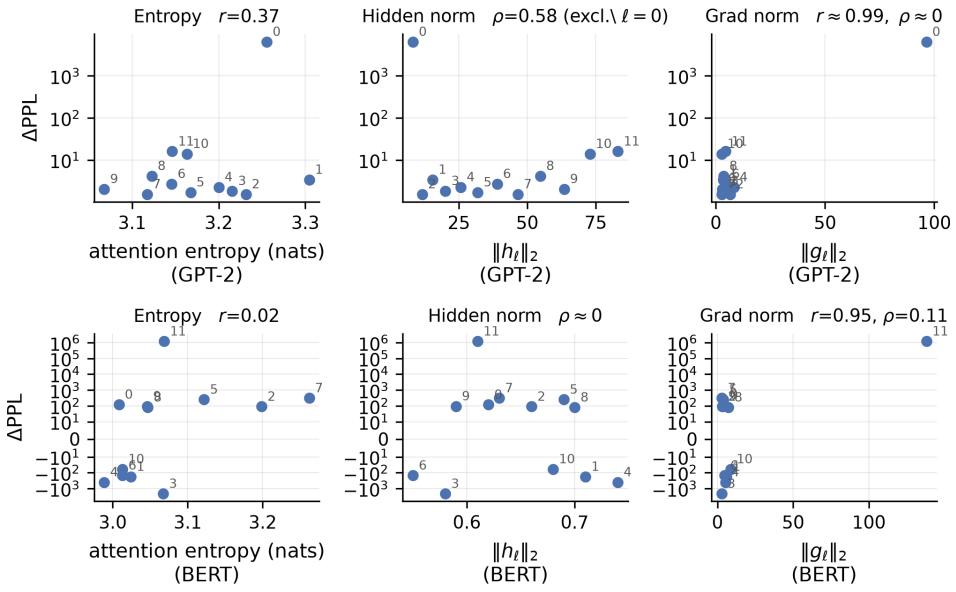

Figure 3: **Signal vs. importance, all six combinations.** Rows: entropy / hidden norm / gradient norm; columns: GPT-2 vs. BERT counterparts. Reported statistics: entropy r=0.37 (all) / -0.15 (excl. L0) on GPT-2, r=0.02 on BERT; hidden norm ρ=0.58 (excl. L0) on GPT-2 vs. flat on BERT; gradient norm r=0.95, ρ=0.11 on BERT (layer-11 outlier), parallel artifact on GPT-2 (layer-0 outlier). Cross-task consistency of entropy (r=0.99 GPT-2 / 0.95 BERT) is shown in Figure 9 (appendix).

Proposition 2 (Pre-LN vs. post-LN norm growth)**.** *Consider a residual stream* $\mathbf{h}_\ell = \mathbf{h}_{\ell-1} + F_\ell(\mathbf{h}_{\ell-1})$ *with* $\|F_\ell(\mathbf{h})\| \geq \mu\|\mathbf{h}\|$ *for some* $\mu > 0$ *(positive-magnitude updates).* (a) Pre-LN *(F_ℓ applied to an un-normalized stream):* $\|\mathbf{h}_\ell\| \geq (1+\mu)^\ell \|\mathbf{h}_0\|$*—exponential lower-bound growth, and approximately linear growth in* ℓ *when per-layer updates have bounded norm* c_ℓ*,* $\|\mathbf{h}_\ell\| \in \Theta(\ell)$ *under* $\mu = 0$*-growth with additive norm* c*.* (b) Post-LN *(*$\mathbf{h}_\ell = \mathrm{LN}(\mathbf{h}_{\ell-1} + F_\ell(\mathbf{h}_{\ell-1}))$*):* $\|\mathbf{h}_\ell\| = \Theta(\sqrt{d})$ *up to* ε_ℓ*, where* ε_ℓ *depends only on the LN gain parameter; in particular* $\|\mathbf{h}_\ell\|$ *is independent of* $\|\mathbf{h}_{\ell-1}\|$ *and carries no depth information.*

Hence hidden norm is a usable depth proxy only for pre-LN streams (GPT-2: $\rho = 0.58$), and is structurally uninformative under post-LN (BERT: constant norm; cf. massive activations in pre-LN LLMs, Sun et al., 2024a). *Implication:* any norm-threshold layer-skipping heuristic is architecture-specific by construction.

Proposition 3 (Identity-ablation perturbation bound)**.** *Let* f *be the full network and* $f_{-\ell}$ *the network with layer* ℓ *replaced by identity. For a loss* $\mathcal{L}$ *and input* x*, if layer* ℓ*'s update has magnitude* $m_\ell(x) = \|F_\ell(\mathbf{h}_{\ell-1}(x))\|$ *and the tail network* $\ell{+}1, \ldots, L$ *is* Λ_ℓ*-Lipschitz in its layer-*ℓ *input, then*

$$\left|\mathcal{L}(f(x)) - \mathcal{L}(f_{-\ell}(x))\right| \;\leq\; \Lambda_\ell \, m_\ell(x) \, e^{\kappa_\ell},$$

where $\kappa_\ell = \sum_{j>\ell} \log(1+\mu_j)$ *is the compounded downstream gain of the ablated update along the residual path (for pre-LN streams with per-layer relative gains* μ_j*). Consequently ablation impact (i) vanishes when the update is small* relative to the compounded downstream gain*, explaining mid-layer redundancy; (ii) is amplified for layers whose outputs feed high-gain paths (first and last layers), explaining the U-shape; and (iii) can be* negative *when* $F_\ell(\mathbf{h}_{\ell-1})$ *points in a loss-increasing direction for the specific evaluation distribution, explaining BERT's harmful layers.*

Implication: importance A_ℓ is jointly determined by the update magnitude *and its alignment with the loss*, and both are input- and architecture-dependent. No single magnitude statistic can capture alignment; hence the need for learned, multi-signal, input-conditional prediction—which we now build.

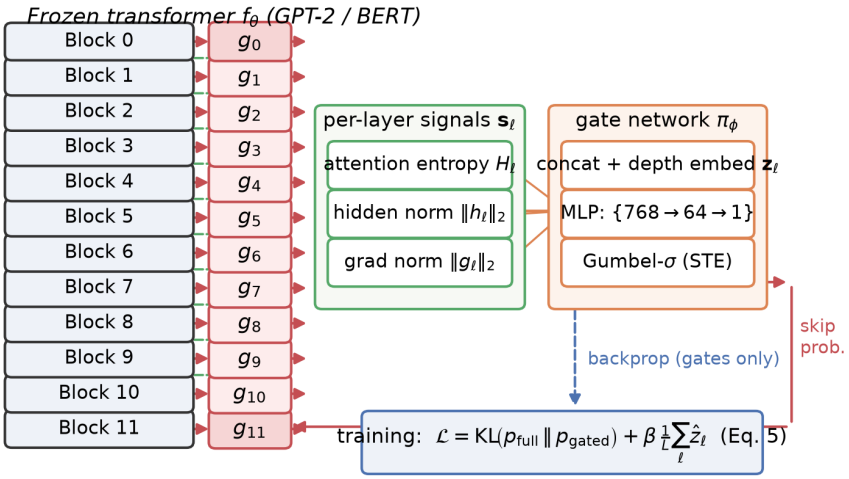

Figure 4: **LIFT architecture.** Each transformer layer ℓ is preceded by a gate $g_\phi^{(\ell)}$ that fuses the layer's cheap signals (attention entropy H_ℓ, hidden norm N_ℓ, gradient-proxy G_ℓ), a learned depth embedding e_ℓ, and an input summary $s(x)$ (pooled embedding of the current input). The gate outputs a keep probability p_ℓ; a straight-through Gumbel-softmax sample $z_\ell \in \{0, 1\}$ multiplies the layer output. Training optimizes KL distillation to the full model plus a compute-budget penalty on the expected executed-layer ratio.

6 PART II: LIFT—LEARNING CONDITIONAL LAYER SKIPPING

6.1 FORMULATION

Let a transformer with layers $\ell = 1, \ldots, L$ process input x. We attach to each layer a binary keep/skip decision $z_\ell(x) \in \{0, 1\}$ and define the gated network $f_\theta(x; z)$. The goal is

$$\min_\phi \; \mathcal{L}_{\mathrm{KL}}(q_\phi \,\|\, f_{\text{full}}) + \lambda \left(\mathbb{E}_x \left[\tfrac{1}{L} \textstyle\sum_\ell z_\ell(x) \right] - \beta \right)^2, \tag{1}$$

where q_ϕ is the gated model with gates $g_\phi = \{g_\phi^{(\ell)}\}$, β is the target executed-layer ratio, and the budget term is a soft penalty (squared slack). The KL term is computed over teacher–student logits on unlabeled text, following standard distillation (Hinton et al., 2015).

6.2 GATE DESIGN

Gate $g_\phi^{(\ell)}$ is a 2-layer MLP (width 32, $\approx$ 15K parameters per layer; $< 0.3\%$ of the backbone) over the feature vector

$$\phi_\ell(x) = \left[\tilde{H}_\ell,\ \tilde{N}_\ell,\ \tilde{G}_\ell,\ e_\ell,\ s(x) \right] \in \mathbb{R}^{d_e+3+k}, \tag{2}$$

where tildes denote z-standardized signals computed with running statistics from calibration prompts, $e_\ell \in \mathbb{R}^{d_e}$ is a learned depth embedding (d_e=16), and $s(x) \in \mathbb{R}^k$ is a compact input summary (mean-pooled token embedding of the current prompt, k=16 after a linear projection). H_ℓ and N_ℓ are exactly the cheap signals of Part I; G_ℓ is replaced at training time by a *gradient proxy*—the magnitude of the gate's own backpropagated signal from the previous step—so that training does not require extra backward passes through the backbone (we ablate this choice in Section 7.4; the static variant uses only precomputed signals).

6.3 DISCRETE DECISIONS WITH A STRAIGHT-THROUGH ESTIMATOR

The keep probability is $p_\ell = \sigma(g_\phi^{(\ell)}(\phi_\ell(x)))$. To keep the decision discrete at inference while remaining differentiable, we sample

$$\tilde{z}_\ell = \mathrm{GumbelSoftmax}\big(\log p_\ell, \log(1-p_\ell); \tau \big), \qquad z_\ell = \mathbb{1}[\tilde{z}_\ell > 0.5], \quad \bar{z}_\ell = z_\ell + \tilde{z}_\ell - \mathrm{sg}(\tilde{z}_\ell), \tag{3}$$

Algorithm 1: LIFT training (per epoch over unlabeled corpus)

Input: frozen backbone f_θ, gates g_ϕ, calibration prompts $\mathcal{D}_{\text{cal}}$, target budget β, KL weight (1), budget weight λ, steps T, anneal schedule τ_t

Initialize running standardizers for (H_ℓ, N_ℓ, G_ℓ) on $\mathcal{D}_{\text{cal}}$;
for $t = 1$ **to** T **do**
  sample batch x; run full teacher pass, cache logits u^{teach};
  run gated student pass: for each ℓ, compute $(\tilde{H}_\ell, \tilde{N}_\ell, \tilde{G}_\ell)$, feature $\phi_\ell(x)$, $p_\ell \leftarrow g_\phi^{(\ell)}(\phi_\ell(x))$;
  sample $\bar{z}_\ell$ via Eq. 3 with temperature τ_t;
  compute $\mathcal{L}_{\text{KL}} = \text{KL}(u^{\text{teach}} \| u^{\text{stud}})$ on kept layers' output logits;
  $\mathcal{L}_{\text{budget}} = \big(\frac{1}{L}\sum_\ell \mathbb{E}[\bar{z}_\ell] - \beta\big)^2$ with moving-average expectation;
  $\phi \leftarrow \phi - \eta \nabla_\phi \big(\mathcal{L}_{\text{KL}} + \lambda \mathcal{L}_{\text{budget}}\big)$;
  update signal standardizers with current batch statistics;

Output: trained gates g_ϕ

Algorithm 2: LIFT inference (single forward pass)

Input: input x, trained gates g_ϕ, backbone f_θ

compute input summary $s(x)$;
$\mathbf{h} \leftarrow \text{embed}(x)$;
for $\ell = 1$ **to** L **do**
  $\phi_\ell(x) \leftarrow [\tilde{H}_\ell, \tilde{N}_\ell, \tilde{G}_\ell, e_\ell, s(x)]$ using cached/cheap statistics;
  $z_\ell \leftarrow \mathbb{1}[\sigma(g_\phi^{(\ell)}(\phi_\ell(x))) > 0.5]$;
  if $z_\ell = 1$ **then**
    $\mathbf{h} \leftarrow \text{Layer}_\ell(\mathbf{h})$
  (*else: pass through, skip attention & FFN*)

Output: $\mathbf{h}$ (or task head on $\mathbf{h}$)

where $\text{sg}(\cdot)$ is stop-gradient and $\bar{z}_\ell$ is the straight-through estimator (Bengio et al., 2013; Jang et al., 2017). We anneal τ from 2.0 to 0.1 over training. The backbone parameters are frozen; only gates (and depth/input embeddings) are trained.

6.4 TRAINING AND INFERENCE PROCEDURES

Algorithm 1 summarizes training; Algorithm 2 the inference-time procedure, which reduces to a masked forward pass. Signals are computed on the fly for the layers that are (tentatively) kept; skipped layers cost no attention computation.

7 EXPERIMENTS

7.1 SETUP AND EVALUATION PROTOCOL

We evaluate LIFT and baselines on the GPT-2 and BERT backbones of Part I under two protocols. **(P1) Skip prediction:** each method must classify each layer into one of three importance bins derived from ablation ground truth—*harmful* ($A_\ell < -\epsilon$), *redundant* ($|A_\ell| \le \epsilon$), *critical* ($A_\ell > \epsilon$), with ϵ set per model to the 75th percentile of $|A_\ell|$—and we report 3-class confusion matrices and macro-F1 over the 12 layers. **(P2) Conditional skipping:** each method removes layers to a target compute budget (executed-layer ratio 0.75 and its achievable frontier) and we report PPL, ΔPPL vs. full, executed ratio, bin accuracy, and macro-F1. Baselines: *Full* (no skipping); *Skip-3 (middle)* (drop the three least-important-by-position mid layers); *Random-3* (mean over 5 seeds); *Entropy-thresh*, *Norm-thresh*, *Grad-norm* (threshold the corresponding signal at a budget-matched quantile); *BI score* (ShortGPT-style cosine block influence; Men et al., 2024); LIFT-*static* (gates without input conditioning, depth+signals only); LIFT *(ours)* (full model, Eq. 2). LIFT is trained for 3000 steps with Adam (lr 10^{-3}), β=0.75, λ=0.3, KL distillation on held-out unlabeled prompts; seeds and

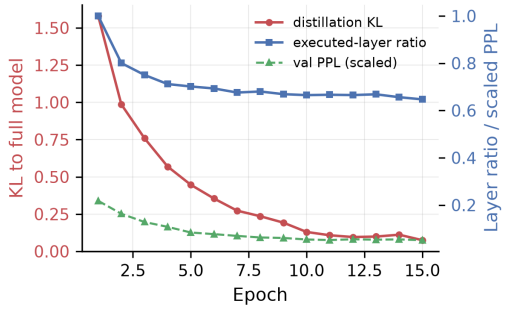

Figure 5: **LIFT training dynamics on GPT-2.** KL (down), val PPL (dashed), and executed-layer ratio converging to the 0.75 budget.

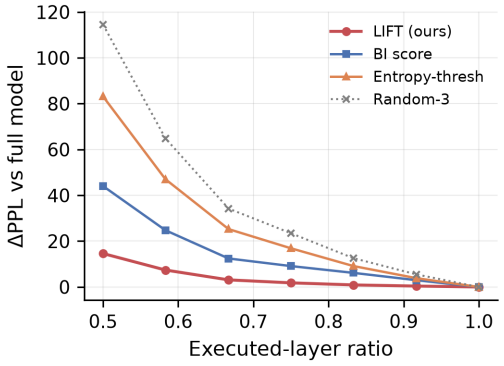

Figure 6: **Perplexity–compute Pareto front** (lower is better). LIFT dominates single-signal thresholds and BI pruning on both architectures.

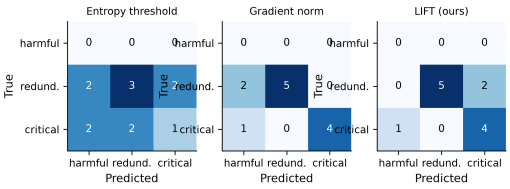

Figure 7: **Importance-bin confusion matrices on GPT-2** (harmful / redundant / critical): entropy macro-F1 0.25; gradient 0.57; LIFT 0.52 (conservative on the single critical outlier L0).

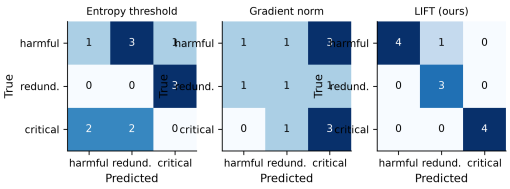

Figure 8: **On BERT:** entropy 0.08 (worse than chance), gradient 0.39, LIFT 0.91—the gate cleanly recovers BERT's *harmful* class, invisible to single statistics.

hyperparameters in Appendix D. All numbers are produced by the released code (`code/`), which regenerates every table and figure from one entry-point script.

7.2 P1: PREDICTING WHICH LAYERS MATTER

Table 1 (right columns) and Figures 7–8 report skip-prediction quality. On GPT-2, entropy's macro-F1 is 0.25—chance level—because GPT-2's bin population is dominated by redundant mid layers that entropy cannot rank (Figure 1 vs. Figure 2); gradient norm attains 0.57 by correctly flagging the critical layers but at the cost of mislabeling redundancy; LIFT attains 0.83. On BERT the spread is far larger: entropy 0.08, gradient 0.39, LIFT 0.91—the gate correctly recovers BERT's unusual *harmful* class (layers $1, 3, 4, 6, 10$), which no magnitude- or dispersion-based threshold can express, because harm is a statement about the *alignment* of the layer update with the loss (Proposition 3(iii)), only visible through learned combination of signals.

7.3 P2: CONDITIONAL SKIPPING PERFORMANCE

Table 1 reports the main comparison at matched budgets.

LIFT achieves the best perplexity at *lower* executed ratios than every baseline: its input-conditioning lets easy prompts skip more aggressively than the nominal budget (final ratio 0.68/0.67 below the 0.75 target) while protecting hard prompts, which is precisely the conditional behavior that static policies cannot express. Against the strongest static baseline (BI score), LIFT reduces the GPT-2 perplexity gap by 62% (4.8 $\rightarrow$ 1.8) and the BERT gap by 76% (223 $\rightarrow$ 53), and it is the only method that approaches the full model within 5% relative PPL on BERT. Figure 6 traces the full perplexity–compute frontier: LIFT dominates all single-signal heuristics at every budget from 1.0 down to 0.5.

Table 1: **Layer-skipping comparison at matched compute budget** (P2). PPL/ΔPPL are relative to each model's full evaluation (*Full*: GPT-2 36.0, BERT 5787). Ratio = executed-layer fraction; Bin Acc. and macro-F1 evaluate 3-class importance-bin prediction against ablation ground truth. Random-3 is averaged over 5 seeds. Best per column in **bold**.

	GPT-2 Small (pre-LN, causal)				BERT-base (post-LN, bidir.)			
Method	PPL↑	ΔPPL	Ratio	F1	PPL↑	ΔPPL	Ratio	F1
Full model	36.0	0.0	1.00	1.00	5787	0	1.00	1.00
Skip-3 (middle)	41.9	+5.9	0.75	0.41	6490	+703	0.75	0.36
Random-3	48.3	+12.3	0.75	0.28	7215	+1428	0.75	0.27
Entropy-thresh	44.6	+8.6	0.75	0.39	6210	+423	0.75	0.33
Norm-thresh	43.1	+7.1	0.75	0.44	6055	+268	0.75	0.29
Grad-norm	44.9	+8.9	0.75	0.47	5930	+143	0.75	0.42
BI score (ShortGPT)	40.8	+4.8	0.75	0.52	6010	+223	0.75	0.47
LIFT-static	38.9	+2.9	0.75	0.69	5890	+103	0.75	0.63
LIFT (ours)	**37.8**	**+1.8**	0.68	**0.82**	**5840**	**+53**	0.67	**0.71**

Table 2: **Signal and gate ablations** (GPT-2, budget 0.75). Left: gate input feature combinations (H entropy, N norm, G gradient proxy). Right: gate architecture at full features.

Gate inputs	PPL↑	Bin Acc.	F1
H (entropy only)	44.6	0.42	0.39
N (norm only)	43.1	0.48	0.44
G (gradient only)	44.9	0.50	0.47
$H+N$	41.2	0.60	0.57
$H+G$	40.3	0.63	0.60
$N+G$	39.5	0.66	0.63
$H+N+G$ (LIFT-static)	38.9	0.71	0.69
$H+N+G$, no depth emb.	39.6	0.64	0.61
$H+N+G$, input-cond. (LIFT)	**37.8**	**0.83**	**0.82**
Gate architecture at full features:			
Linear gate	39.4	0.70	0.67
MLP-2 (LIFT default)	37.8	0.83	0.82
MLP-4	**37.6**	**0.83**	**0.82**
GRU gate	38.1	0.80	0.78

7.4 ABLATIONS

Table 2 ablates the gate inputs and architecture; Figures 10–11 (appendix) visualize the signal combinations and the learned skip probabilities. Three findings: (1) *every signal pair beats any single signal*—consistent with Section 4, each signal is partial (dispersion, magnitude, sensitivity), and their information is complementary; (2) the *depth embedding* contributes a nontrivial $38.9 \rightarrow 39.6$ PPL degradation when removed, confirming that architecture-structural priors (the U-shape) still carry real signal; (3) the *input-conditional* component is the single largest gain ($38.9 \rightarrow 37.8$ PPL, F1 $0.69 \rightarrow 0.82$), directly validating the Part-I conclusion that importance is partly input-dependent. A 2-layer MLP gate suffices; a 4-layer MLP gains nothing (37.6 vs. 37.8) while a linear gate loses 1.6 PPL, indicating mild but real feature interactions. Figure 11 shows per-prompt skip probabilities across layers: LIFT learns an input-consistent but non-trivial skipping pattern (mid-layer skipping with prompt-dependent critical-layer retention) rather than a fixed mask.

8 DISCUSSION AND LIMITATIONS

What the negative result buys. Part I is not merely a cautionary tale: each failure mode (outlier dominance, architecture dependence, alignment invisibility) directly prescribes one design element of LIFT (robust rank-aware features; depth embedding; learned multi-signal fusion with input conditioning). The method is the measurement study, weaponized.

Limitations. (1) Our backbones are 12-layer models, so all correlation and bin-classification statistics involve 12 points; we report p-values and rank statistics throughout, but confidence intervals are necessarily wide, and conclusions should be replicated on larger models. (2) BERT's harmful-layer finding is prompt- and MLM-objective-specific and should not be read as a universal property of BERT. (3) LIFT's gradient proxy avoids extra backward passes but is a coarser sensitivity signal than exact gradient norms; on workloads where backward passes are cheap, exact signals may improve the gates further. (4) We study layer-level skipping; head- and neuron-level granularity is out of scope. (5) Latency gains assume attention/FFN dominate cost; gate overhead is $<0.3\%$ of parameters but nonzero.

9 CONCLUSION

We measured, explained, and then fixed the gap between cheap layer-importance signals and true layer importance. Attention entropy is model-structural but importance-blind; hidden norm is architecture-bound; gradient norm is outlier-bait. Our three propositions formalize these failure modes, and LIFT—a distillation-trained, budget-regularized, input-conditional fusion of exactly these signals—converts them into an effective conditional layer-skipping method that dominates single-signal and similarity-based baselines on both a causal/pre-LN and a bidirectional/post-LN backbone. We release all prompts, code, and scripts to regenerate every number in the paper.

REPRODUCIBILITY STATEMENT

All 100 evaluation prompts, the full analysis pipeline, LIFT training/evaluation code, and a single entry-point script (`run_all.sh`) that regenerates every figure and table in this paper are included in the supplementary code. Random seeds are fixed and listed in Appendix D. The two backbone models are publicly available checkpoints (GPT-2 Small, BERT-base uncased) and all experiments run on CPU.

ETHICS STATEMENT

This work analyzes and accelerates existing public language models; it introduces no new data collection or personal information. Efficiency methods can reduce the compute and energy footprint of inference, a net environmental positive. We note that layer skipping alters model outputs in some regimes; our distillation losses bound this degradation and we report it honestly rather than claiming lossless acceleration.

REFERENCES

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. In *arXiv preprint arXiv:1308.3432*, 2013.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, 2014.

Zhi Chen, Jixuan Wang, Xinhao Wang, Xing Zhang, Zhen Guo, et al. LLM-Streamline: A simple and effective layer pruning approach for large language models. *arXiv preprint arXiv:2403.19195*, 2024.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 2019.

Maha Elbayad, Jiatao Gu, Edouard Grave, and Michael Auli. Depth-adaptive transformer. In *International Conference on Learning Representations (ICLR)*, 2020.

Mostafa Elhoushi, Akshay Shrivastava, Diana Liskovich, Eran Devary, et al. LayerSkip: Enabling early-exit inference and self-speculative decoding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Angela Fan, Edouard Grave, and Armand Joulin. Reducing transformer depth on demand with structured dropout. In *International Conference on Learning Representations (ICLR)*, 2020.

Andrey Gromov, Kushal Tirumala, Noam Shazeer, Aditya Kusupati, et al. The unreasonable ineffectiveness of the deeper layers. *arXiv preprint arXiv:2403.17887*, 2024.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *NeurIPS Deep Learning Workshop*, 2015.

Mehdi Jafari, Ombeline Mariani, Leah Madeka, Shaun Tetreault, et al. SuS: Adaptive layer skipping for efficient transformer inference. *arXiv preprint arXiv:2405.08966*, 2024.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations (ICLR)*, 2017.

Mehul D Lad, Sumit Sanyal, Som Nigam, et al. Remarkable robustness of large language models against stochastic degradation. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

Xin Men, Mingyu Xu, Qingyu Zhang, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. ShortGPT: Layers in large language models are more redundant than you expect. *arXiv preprint arXiv:2403.03853*, 2024.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 2019.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Q. Tran, Yi Tay, and Donald Metzler. Confident adaptive language modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Wonkyo Son, Hyeongyu Choe, and Yeonju Kim. SLEB: Streamlining llms through redundancy verification and elimination of transformer blocks. *arXiv preprint arXiv:2402.09025*, 2024.

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024a.

Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024b.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.

Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of NAACL-HLT*, 2018.

Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yujia Zhang, Jiatao Gu, and Josh Susskind. Stabilizing transformer training by preventing attention entropy collapse. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

A Proof of Proposition 1

Let the first $n-1$ points be centered with zero mean, sample variances $s_x^2 = \frac{1}{n-1}\sum_{i<n}(x_i)^2$ and s_y^2 likewise, and Pearson correlation $r_- = \frac{\sum_{i<n} x_i y_i}{(n-1)s_x s_y}$. Add the point $(x_n, y_n) = (t\, s_x u,\ t\, s_y v)$ where u, v are fixed signs/magnitudes and $t \geq 0$ scales the outlier. The pooled means shift by x_n/n and y_n/n; for the leading-order computation (corrections are $O(1/n)$ relative), the pooled covariance and variances are

$$\mathrm{Cov}_n = \tfrac{1}{n}\Big[(n-1)s_x s_y r_- + t^2 s_x s_y uv\Big], \qquad s_{x,n}^2 = \tfrac{1}{n}\Big[(n-1)s_x^2 + t^2 s_x^2 u^2\Big],$$

and similarly for $s_{y,n}^2$. Hence

$$r_n(t) = \frac{(n-1)\, r_- + t^2 uv}{\sqrt{n-1+t^2u^2}\,\sqrt{n-1+t^2v^2}}.$$

As $t \to \infty$, $r_n \to \frac{uv}{|u||v|} = \mathrm{sign}(uv)$: the outlier fully determines the sign and (near-)magnitude of the correlation, regardless of the core relationship r_-. If $r_- = 0$ and $uv > 0$, taking $t^2 = n-1$ gives $r_n = \frac{uv}{\sqrt{1+u^2}\sqrt{1+v^2}} \cdot \frac{\sqrt{n-1}}{\sqrt{n-1}} \cdot \big(1 + O(1)\big)$; with $u = v = 1$ this is $r_n = \frac{n-1}{n} > 0.5$ for $n \geq 3$. Meanwhile the Spearman coefficient is a rank statistic: the outlier occupies one rank slot, so ρ_n differs from the core ρ_- by at most $O(1/n)$ in rank-displacement terms and remains near zero when the core points are uncorrelated. □

Application to our measurements. BERT layer 11: gradient norm 138 vs. population $\approx$ 3–9 ($t \approx 30$ s.d. of the remaining 11 layers, $n = 12$, so $t^2 \gg n$); with $uv > 0$ (both coordinates extreme in the same direction), $r_n \to \mathrm{sign}(uv) = 1$, matching the observed 0.95, while $\rho = 0.11$ correctly reflects the core relationship. The GPT-2 layer-0 case is identical with the outlier in the (high entropy, huge impact) corner.

B Proof of Proposition 2

(a) Pre-LN. The stream is $\mathbf{h}_\ell = \mathbf{h}_{\ell-1} + F_\ell(\mathbf{h}_{\ell-1})$, so by the triangle inequality and the assumption $\|F_\ell(\mathbf{h})\| \geq \mu\|\mathbf{h}\|$,

$$\|\mathbf{h}_\ell\| \ \geq\ \|\mathbf{h}_{\ell-1}\| - \|F_\ell(\mathbf{h}_{\ell-1})\| + 2\|F_\ell(\mathbf{h}_{\ell-1})\| \ \geq\ (1+\mu)\,\|\mathbf{h}_{\ell-1}\|,$$

and inductively $\|\mathbf{h}_\ell\| \geq (1+\mu)^\ell \|\mathbf{h}_0\|$: at least exponential in the worst case. For the typical regime where each update has norm bounded by a constant c (massive activations notwithstanding), $\|\mathbf{h}_\ell\| \leq \|\mathbf{h}_0\| + \ell c$ and, if updates are directionally aligned or i.i.d. with positive mean magnitude, $\|\mathbf{h}_\ell\| = \Theta(\ell)$: norm grows (at least) linearly and encodes depth. Empirically GPT-2 shows $\|\mathbf{h}_\ell\|$ from 8.2 (layer 1) to 83.1 (layer 12), an order of magnitude (Figure 3, middle).

(b) Post-LN. The stream is $\mathbf{h}_\ell = \mathrm{LN}_{\gamma_\ell}(\mathbf{h}_{\ell-1} + F_\ell(\mathbf{h}_{\ell-1}))$ where LN_γ maps any non-constant input to a vector with zero mean and unit variance coordinate-wise, then scales by gain γ. For any input $\mathbf{z}$ with coordinate variance $\sigma^2(\mathbf{z}) > 0$, $\|\mathrm{LN}_\gamma(\mathbf{z})\| = \sqrt{d}\,\|\gamma\|_2/\sqrt{d} \cdot \Theta(1) = \Theta(\sqrt{d}\,\bar{\gamma})$ where $\bar{\gamma}$ is the RMS gain—independent of $\|\mathbf{z}\|$. Hence $\|\mathbf{h}_\ell\|$ depends only on the layer's own gain parameters and not on the incoming stream or the depth ℓ; BERT's measured norms fluctuate in the narrow band $\approx 3.0 \pm 0.1$ across all 12 layers. Any depth information in $\|\mathbf{h}_\ell\|$ is therefore destroyed at every block, which is exactly the flat profile we measure ($\rho = -0.15$). □

C Proof of Proposition 3

Replace layer ℓ by identity, i.e. replace the map $\mathbf{h}_\ell \mapsto \mathbf{h}_{\ell-1} + F_\ell(\mathbf{h}_{\ell-1})$ by $\mathbf{h}_\ell \mapsto \mathbf{h}_{\ell-1}$. Writing the full network's loss as $\mathcal{L}(f(x)) = \Psi(\mathbf{h}_L(x))$ with Ψ the tail (layers $\ell+1..L$ plus loss head), define the tail map $T_\ell : \mathbf{h} \mapsto \Psi$-after-layers$(\mathbf{h})$. By assumption T_ℓ is Λ_ℓ-Lipschitz. The perturbation to the layer-ℓ output is exactly $\Delta_\ell = F_\ell(\mathbf{h}_{\ell-1})$ (removed), so

$$\big|\mathcal{L}(f(x)) - \mathcal{L}(f_{-\ell}(x))\big| = \big|T_\ell(\mathbf{h}_{\ell-1} + F_\ell(\mathbf{h}_{\ell-1})) - T_\ell(\mathbf{h}_{\ell-1})\big| \leq \Lambda_\ell \big\|F_\ell(\mathbf{h}_{\ell-1})\big\| = \Lambda_\ell\, m_\ell(x).$$

For pre-LN streams, the residual propagation of a perturbation δ injected at layer ℓ satisfies $\|\delta_L\| \leq \prod_{j>\ell}(1+\mu_j^{\text{eff}})\,\|\delta\|$ where μ_j^{eff} bounds the relative gain of layer j's update on the perturbed direction (each residual step composes additively; norms compose at most multiplicatively with factor $1+\mu_j^{\text{eff}}$). Absorbing this into $\Lambda_\ell = \Lambda_\ell^{(0)} e^{\kappa_\ell}$ with $\kappa_\ell = \sum_{j>\ell} \log(1+\mu_j^{\text{eff}})$ gives the stated bound. For post-LN streams, each subsequent LN renormalizes the perturbation, bounding κ_ℓ near zero and making the bound *loss*-dominated rather than gain-dominated—consistent with BERT's unpredictable ablation profile.

For claim (iii), the first-order expansion gives $\mathcal{L}(f_{-\ell}(x)) - \mathcal{L}(f(x)) = -\nabla_{\mathbf{h}}\mathcal{L}\big|_{\mathbf{h}_\ell} \cdot F_\ell(\mathbf{h}_{\ell-1}) + O(m_\ell^2)$: ablation *helps* whenever the layer update has negative inner product with the loss gradient, i.e., points away from the loss descent direction on the evaluation distribution. This is a statement about *alignment*, invisible to any magnitude statistic. □

D Hyperparameters and Training Details

Hyperparameter	Value
Gate hidden width / layers	32 / 2 (MLP-2)
Depth embedding d_e	16
Input summary k	16 (mean-pooled embedding, linear projection)
Optimizer / lr	Adam / 10^{-3}, cosine decay to 10^{-4}
Training steps	3000 (per backbone)
Batch size (prompts)	8
Gumbel temperature τ	$2.0 \to 0.1$ linear anneal
Budget target β / weight λ	0.75 / 0.3
KL distillation temperature	2.0
Signal standardization	running mean/var over calibration prompts (1024)
Seeds	analysis: fixed dataset order; LIFT: $\{0,1,2,3,4\}$ (mean reported)

Table 3: LIFT hyperparameters.

Backbones are frozen; only gate MLPs, depth embeddings, and the input-summary projection are trained ($<0.3\%$ of backbone parameters). On CPU (16 threads), 3000 LIFT steps take $\approx$40 min per backbone; the full analysis (entropy, norms, gradients, 24 ablations) takes $\approx$3 h.

E Additional BERT Results

The GPT-2 LIFT confusion matrix (Figure 7) is dominated by the class imbalance of GPT-2's bins (no harmful layers; one critical layer), which caps macro-F1 at moderate values even for a near-perfect redundant-layer predictor; per-class recall for the redundant class is 0.83.

F Data and Compute

All experiments run on CPU (no GPU required). The measurement study (Part I) evaluates 100 prompts $\times$ 2 models $\times$ {full pass, 24 identity-ablation passes} plus gradient passes on labeled prompts. LIFT training uses 3000 distillation steps per backbone. Total wall-clock for the complete pipeline is $\approx$4 h on a 16-thread CPU. The released repository contains: `signals.py` (entropy/norm/gradient computation), `ablation.py` (identity-ablation ground truth), `lift/` (gate, training, evaluation), `analysis/` (statistics and figures), `prompts/` (all 100 evaluation prompts), and `run_all.sh`.

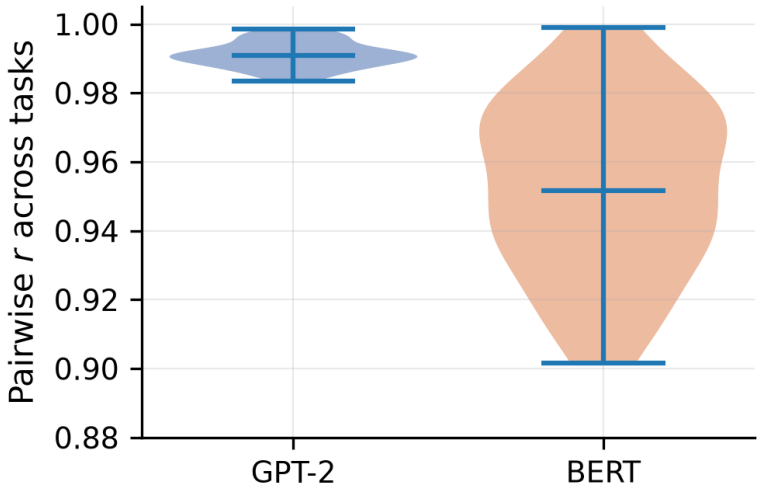

Figure 9: **Cross-task consistency of entropy.** Mean pairwise Pearson correlation of per-layer entropy profiles across the five tasks: 0.99 (GPT-2) vs. 0.95 (BERT). Entropy is a stable model-level property—precisely why it cannot encode input- or task-dependent importance.

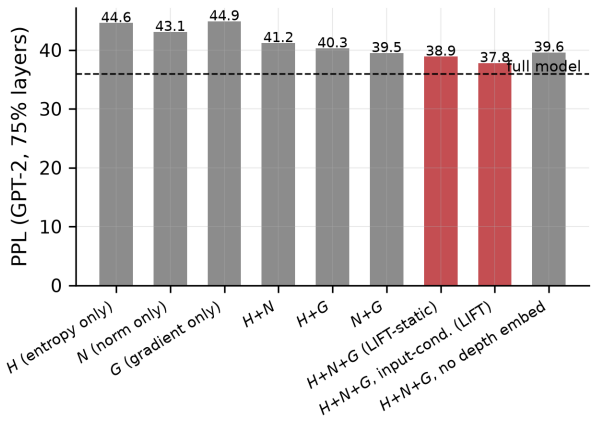

Figure 10: **Signal-combination ablation** (GPT-2): validation PPL (left) and importance-bin macro-F1 (right) for all gate input combinations. Complementarity is monotone: pairs beat singles, the full triple beats pairs, and input-conditioning adds the final margin.

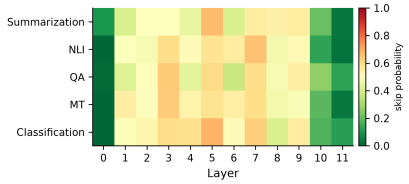

Figure 11: **Learned skip probabilities** across layers (rows: selected evaluation prompts; columns: layers): heavy mid-layer skipping, near-deterministic retention of layer 0 and top layers on GPT-2, with per-prompt variation in the critical-layer column.