What Do Mixture-of-Depth Routers Learn? Routing Patterns in Gemma 2

Anonymous Authors[1]

Abstract

Mixture-of-Depths (MoD) routers improve efficiency of transformer models by learning which tokens to process and which to bypass at each layer. However, their learned routing patterns have not been characterized in alternating-attention architectures, which mix local attention with sparse global attention to increase efficiency. We evaluate routing by training token-level MoD routers on Gemma 2 2B and find that the mean routing rate is significantly higher in full-attention layers than at sliding-window layers. We find that a token's part of speech is correlated with whether the router skips or processes it, and that the same category is often routed differently at different layers: for example, determiners are preferentially skipped at the shallowest target layer but preserved at deeper ones. Because these patterns hold within single layers, they are not subject to the confound (perfect coincidence of attention type and layer parity in Gemma 2) that limits our routing-rate result.

1. Introduction

Transformer language models process every token at every layer, regardless of how much computation that token actually requires. This is wasteful since not all tokens benefit equally from every layer. Mixture-of-Depths (MoD) (Raposo et al., 2024) addresses this inefficiency by adding a small, per-layer router (typically a single linear layer) that decides, for each token at each transformer block, whether the token receives full attention processing or bypasses the block via the residual stream. Router-Tuning (He et al., 2025) extends MoD to a parameter-efficient variant that trains only the routers on a frozen base model, which allows us to study the behavior of learned MoD routers.

In recent work (Gadhikar et al., 2024; Zhang et al., 2024; Tan et al., 2024; Luo et al., 2025; Laitenberger et al., 2026) the router is treated as an efficiency mechanism: given a target capacity (the fraction of tokens to process per layer), the router is trained to meet it and evaluated by perplexity at that capacity. The question of *what patterns these routers learn* and whether their decisions reflect any identifiable structure beyond gross token-importance heuristics has not been systematically investigated. Existing MoD work has focused on uniform-attention transformers, where every block uses the same attention type. Modern architectures such as Gemma 2 (Team et al., 2024) alternate full and sliding-window attention layer-by-layer, which could result in different router behavior. No prior work, to our knowledge, has trained MoD on such an architecture or characterized learned routing decisions in this regime. This is important because efficient-inference designs typically assume token importance reduces to simple heuristics (e.g., norm, attention scores), but if routers encode layer-conditional functions instead, then routing decisions become an interpretability target rather than an efficiency mechanism: a window into per-layer computation with implications for safety auditing and controllable inference.

We adapt token-level Router-Tuning to Gemma 2 2B and train four router configurations spanning two capacity-penalty weights and two training scales. Across all four, we find a robust *attention-type preference*: routing rate is consistently higher at full-attention layers than at sliding-window layers in shallow-middle MoD. We test whether the preference can be explained by (a) layer-aggregate residual-stream norm, (b) per-token norm dependence, or (c) per-layer task-loss sensitivity under attention ablation; none of the three accounts for it. We find that a linear part-of-speech probe further identifies a layer-dependent syntactic structure in routing decisions, indicating that the router learns a more complex function than uniform token-importance ranking.

We investigate the following research questions:

- **RQ1:** Do trained MoD routers route differently at full-attention vs. sliding-window layers in Gemma2 2B?
- **RQ2:** Can the differences be explained by simpler architectural properties e.g., (a) layer-aggregate residual-stream norm, (b) per-token residual-stream norm, or (c) per-layer task-loss sensitivity to attention ablation?

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

- **RQ3:** Do routing decisions encode interpretable linguistic features beyond token importance e.g., can per-token routing rate be predicted from part-of-speech tags?

2. Related Work

Raposo et al. (2024) introduced MoD and He et al. (2025) proposed Router-Tuning. Subsequent dynamic-depth work has explored alternatives to learned routers: attention-based routing (Gadhikar et al., 2024), post-hoc adaptation of pre-trained models (Zhang et al., 2024; Tan et al., 2024), multimodal MoD (Luo et al., 2025), and residual-stream gating (Laitenberger et al., 2026). All treat the router as an efficiency mechanism, none analyze its learned decisions. Lawson & Aitchison (2025) report that learning to skip middle layers fails to improve over dense baselines, motivating the finer-grained per-token routing MoD provides.

The work most related to ours is Antoine et al. (2024), who train MLP probes over routing paths in six Mixture-of-Experts models and find that part-of-speech is predictive of expert assignment, treating routing as a multi-class expert-selection problem. Our work differs in the following ways: (1) we ask whether part-of-speech predicts the binary skip/process decision in MoD, rather than the multi-class expert assignment they study, (2) we investigate an alternating-attention architecture rather than the uniform-attention MoE models they study, and (3) we identify layer-conditional sign flips; the same POS preferentially skipped at one target layer can be preserved at another.

Michel et al. (2019) and Voita et al. (2019) score each attention head by the impact of removing it and find that many heads can be uniformly pruned with little task degradation; MoD routers instead make a per-token, per-layer, context-dependent decision. To our knowledge, no prior work has trained MoD on an alternating-attention architecture or investigated their routing behavior.

3. Problem Formulation

We start from Gemma 2 2B (Team et al., 2024), a 26-layer decoder-only language model whose transformer blocks alternate full and sliding-window attention layer-by-layer. We add token-level MoD routing to the deepest 13 layers (layers 12–24) and leave layer 25 dense, following He et al. (2025)'s convention of targeting the deepest layers except the last, based on prior evidence that deeper layers are more redundant than shallow ones, which carry essential representations. We train only the routers on a frozen base model so that any structure observed in routing decisions reflects what the router has learned to read from fixed base-model representations rather than confounded co-adaptation between the two. Freezing the base model means any routing structure we observe must already exist in the pre-trained residual stream; it cannot be the result of the base model adapting its representations to make routing easier.

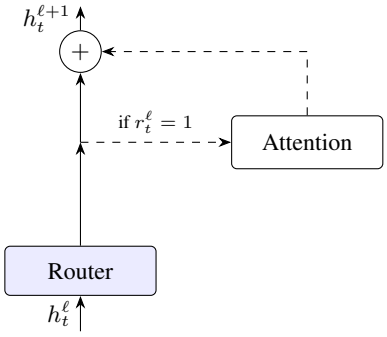

Figure 1. One MoD-target block. The per-token residual h_t^ℓ is read by a small linear router, which emits a binary routing decision $r_t^\ell = \sigma(W_\ell h_t^\ell) \in \{0, 1\}$ (binarized at 0.5 via a straight-through estimator). When $r_t^\ell = 1$ the attention sub-layer is computed and its output is added back to the residual stream (dashed path); when $r_t^\ell = 0$ the attention computation is skipped and the residual passes through unchanged. The MLP sub-layer (not shown) is unchanged. This pattern repeats at each of the 13 MoD-target layers (12–24).

Following He et al. (2025) Eq. 1 (token-level variant), and as illustrated in Figure 1, at each MoD-target layer ℓ a single linear router $W_\ell \in \mathbb{R}^d$ produces a per-token routing score $r_t^\ell = \sigma(W_\ell^\top h_t^\ell)$ (with σ the sigmoid), binarized via a straight-through estimator at threshold 0.5. Training freezes the base-model weights and trains only the routers (∼30K parameters in total) against a combined loss $\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda \cdot \text{ReLU}(\bar{c} - c^*)$, where $\bar{c}$ is the mean routing rate across the batch, $c^* = 0.5$ is the target capacity (the desired fraction of tokens to process per layer), and λ is the gradient scale on the capacity penalty (higher λ enforces stricter targeting; lower λ lets task loss dominate). We train routers using causal language-modelling loss (next-token cross-entropy) on the Alpaca instruction-tuning corpus: 52K (instruction, response) pairs from Stanford CRFM, matching the training setup of He et al. (2025), and evaluate on a held-out corpus (WikiText-2 (Merity et al., 2016)) so that routing patterns reflect generalization rather than memorization.

4. Experiments

We train four checkpoints testing $\lambda \in \{0.01, 0.1\}$ with two training scales (5K samples × 1 epoch; 52K × 3 epochs), spanning an order of magnitude on each axis:

- **strict-short** (λ=0.1, 5K × 1)
- **gentle-short** (λ=0.01, 5K × 1)
- **gentle-long** (λ=0.01, 52K × 3)
- **strict-long** (λ=0.1, 52K × 3)

Strict-long serves as our lead checkpoint for per-token analyses (RQ2b, RQ3, Appendix B): it pairs full training (52K×3) with the paper-spec capacity penalty (λ=0.1).

To determine if a router that sees *only* per-token norm can reproduce the attention-type preference, we train a 2-parameter-per-layer baseline router $\sigma(s_\ell \|h\| + b_\ell)$. This is simply a learned scale and offset on the per-token residual-stream norm under **strict-long**'s training configuration.

All evaluation is on WikiText-2 test (200 documents tokenized and truncated to max-length 256; 33,608 tokens total). For each ℓ, we record the per-token routing decision $r_t^\ell \in \{0, 1\}$ and the per-token residual stream entering the layer. In our per-token analyses, we exclude the first 4 positions per sample to avoid attention-sink outliers (Xiao et al., 2024), leaving 32,808 tokens.

4.1. RQ1: Full vs. sliding window attention

For each WikiText-2 sample and depth range $\mathcal{R}$, we compute mean routing rates across full-attention and sliding-window MoD layers in $\mathcal{R}$. A paired t-test on per-sample means quantifies whether mean routing rate differs by attention type. We report effect size as Cohen's d, the standardized mean difference (Cohen, 1988), with thresholds in $\{0.2, 0.5, 0.8\}$ corresponding to small/medium/large effects.

Table 1 reports the paired t-test for full vs. sliding mean routing rate within each depth range, across all four trained checkpoints. In shallow-middle MoD (layers 12–20), every checkpoint shows positive direction (mean routing rate is higher at full-attention layers than at sliding-window layers, $d > 0$), with Cohen's $d \in [0.31, 2.06]$ and all $p < 10^{-4}$, where **gentle-long** has the lowest skip rate (24% vs. 29–46% in the others). In deep MoD (layers 21–24) the results are more mixed: null in **strict-short**, reversed in **gentle-short**, weakly positive in **gentle-long**, strongly positive in **strict-long**. We note that both long runs (**gentle-long**, **strict-long**) show positive direction in deep MoD, while the two short-training runs (**strict-short**, **gentle-short**) do not. This indicates that the router routes differently by attention type in shallow-middle MoD across all four configurations, while the deep-MoD direction is inconsistent.

The norm-only baseline reproduces the shallow-middle preference more extremely ($d = 6.5$ on **strict-long**'s training configuration) but plateaus at ≈4.5× higher training perplexity than the learned linear router. The preference therefore emerges even from a minimal-capacity router under MoD training, while the learned linear router uses richer signal than norm alone.

4.2. RQ2: Influence of architectural properties

Next, we investigate the influence of three architectural properties on routing behavior.

Table 1. Paired full vs. sliding t-test across all four trained checkpoints, in shallow-middle MoD (layers 12–20) and deep MoD (layers 21–24). Shallow-middle direction is consistent across all four runs; deep direction varies.

Run	Skip	Cohen's d	full > slid %	p
Shallow-middle MoD (layers 12–20)				
strict-short	45.5%	+1.33	86.5%	$4 \cdot 10^{-46}$
gentle-short	28.5%	+2.06	98.5%	$1 \cdot 10^{-73}$
gentle-long	23.9%	+0.31	62.0%	$2 \cdot 10^{-5}$
strict-long	38.3%	+0.86	78.5%	$9 \cdot 10^{-26}$
Deep MoD (layers 21–24)				
strict-short	45.5%	−0.03	53.5%	0.64
gentle-short	28.5%	−1.00	11.5%	$6 \cdot 10^{-32}$
gentle-long	23.9%	+0.35	65.0%	$2 \cdot 10^{-6}$
strict-long	38.3%	+1.23	88.0%	$1 \cdot 10^{-41}$

RQ2a: Layer-aggregate norm: If the router preferred high-norm layers, full-attention layers would be preserved if and only if they had higher residual-stream norms at their input. Table 2 reports the paired t-test on residual-stream norms in base Gemma 2 2B and in the strict-long MoD context. In shallow-middle, full-attention layers have *lower* norms than sliding-window layers, the opposite direction from the routing preference.

Table 2. Layer-aggregate norm paired t-test, shallow-middle (12–20). Negative d indicates full < sliding in norm.

Context	Mean $\|h\|$ full vs. slid	d
Base Gemma 2 2B	83.86 vs. 84.76	−3.02
strict-long MoD context	82.97 vs. 84.31	−3.86

RQ2b: Per-token norm: If the router decided based on per-token norm alone, its score should correlate with per-token $\|h\|$ (Pearson correlation). For each MoD layer we compute the Pearson correlation r between the **strict-long** router's score and the per-token residual-stream norm. Mean $|r| = 0.166$ across MoD layers, max $|r| = 0.336$ at layer 17. At maximum, norm explains $r^2 \approx 11\%$ of variance in the router's score, so per-token norm cannot account for the attention-type preference observed in RQ1.

RQ2c: Task-loss asymmetry: If full-attention layers were more important than sliding-window layers, ablating them would raise perplexity more. We ablate each MoD-target attention sub-block in isolation in base Gemma 2 2B by zeroing the layer's attention output while leaving the rest of the model intact, and measuring the resulting WikiText-2 perplexity. In shallow-middle MoD layers, ablating full-attention layers raises perplexity by +1.56 on average while ablating sliding-window layers raises it by +1.22. The +0.34 difference is not statistically significant (Welch's t-

test, $p = 0.44$; Welch's variant is used because the two groups have unequal sample sizes and possibly unequal variances). To check for a generic depth effect (any deeper layer costing more to ablate regardless of attention type), we repeat the procedure on each layer's MLP sub-block; this is also flat ($p = 0.88$), so we do not detect a per-layer-importance asymmetry.

4.3. RQ3: POS probe for layer-dependent routing

POS is a syntactic property densely encoded in transformer residual streams (Tenney et al., 2019) and is the analysis target in the related MoE-routing study (Antoine et al., 2024), enabling direct comparison. For 200 WikiText-2 samples we tokenize once with spaCy (Honnibal et al., 2020) (word-level POS) and once with the Gemma 2 SentencePiece tokenizer (Team et al., 2024) (subword), assigning each subword token the POS of the spaCy word whose character span contains the subword's midpoint. Per layer, we compute (i) for each POS tag, whether its routing rate differs from the layer's marginal rate $\bar{P}(\text{process})$ using a binomial test (since each routing decision is binary), Bonferroni-corrected across 16 tags per layer; and (ii) a logistic regression from POS one-hot to router decision, reporting accuracy vs. majority-class baseline.

Table 3 reports per-tag deviations from the marginal routing rate at each target layer; all entries are Bonferroni-significant at $\alpha' = 0.0031$ except PROPN at layer 13 ($p = 0.47$, included for layer-23 contrast).

Table 3. Selected per-POS deviation from marginal routing rate. Positive Δ = preferentially preserved; negative = preferentially skipped. All entries Bonferroni-significant at $\alpha' = 0.0031$ except † ($p = 0.47$, shown for layer-23 contrast).

POS	Layer 13	Layer 23	Layer 24
CCONJ	+31.8	+19.9	+14.1
PUNCT	+17.1	+7.5	−22.2
DET	−11.3	+11.5	+14.2
NUM	+5.7	+6.2	+22.9
VERB	−13.0	−6.8	−5.6
PROPN	−0.5†	−12.5	−3.4
SPACE	−49.0	+38.7	−32.4

Values are $(P(\text{process} \mid \text{POS}) - \bar{P}(\text{process})) \times 100$pp.

Aggregate probe: A logistic regression from POS one-hot to router decision achieves test accuracy 0.576 at layer 13 and 0.597 at layer 24, vs. majority baselines of 0.510 and 0.506 – a lift of +6.6pp and +9.1pp respectively. At layer 23 the aggregate probe matches the majority baseline despite per-tag effects. This indicates that POS predicts routing better than majority chance at L13 and L24, providing aggregate evidence (alongside the per-tag tests) that routing decisions encode part-of-speech information.

Layer-dependent inversions: Determiners are preferentially skipped at layer 13 (−11pp) and preferentially preserved at deeper layers (+11pp at layer 23, +14pp at layer 24) – a shallow-vs-deep inversion. Punctuation preserves the same direction across full-attention layers (kept: +17pp at layer 13, +7.5pp at layer 23) but flips at the deepest sliding-window layer (−22pp at layer 24), so its inversion is specific to layer 24, not a shallow-vs-deep split. Space tokens are universally skipped at layer 13 (0%), universally preserved at layer 23 (100%), and preferentially skipped at layer 24 (17%). This indicates that the router treats the same syntactic category differently at different layers, rather than applying a fixed priority across layers.

5. Discussion and Conclusion

We trained four MoD router configurations on Gemma 2 2B. Across all four, routing rate is higher at full-attention than sliding-window layers in shallow-middle MoD (Cohen's d up to 2.06, all $p < 10^{-4}$); this is the attention-type preference posed by RQ1. None of the three properties tested in RQ2 explains it: layer-aggregate norm runs the wrong direction, per-token norm explains at most ∼11% of router-score variance, and per-layer ablation shows no detectable importance asymmetry. RQ3 finds that part of speech predicts routing at every target layer, with a layer-conditional prediction function; the same POS can be preferentially preserved at one layer and skipped at another.

Our work has some limitations, namely that probe correlation does not establish causation. The residual stream encodes POS densely (residual-to-POS probe accuracy > 0.93 at all three target layers), so any direction in residual space will correlate with some POS structure. We have shown that routing decisions are organized along a syntactic axis; we have not shown that the router causally reads POS labels. Causal interventions on a token's POS-relevant features could help establish whether the router reads POS directly or only other concepts correlated with it.

In Gemma 2, every full-attention layer has an odd index and every sliding-window layer an even one, so the RQ1 result fits two readings equally well: "the router prefers full attention" or "the router prefers odd layers". Separating them would require a model where the two vary independently. The RQ3 POS results aren't subject to this: each per-tag effect lives within a single layer where attention type and parity are both fixed.

There are several ways to extend this work, such as replicating on a second alternating-attention architecture (to disentangle the parity confound), activation patching of POS-correlated directions (to test causal readout, Section 4.3), or per-feature analysis with Gemma Scope SAEs (Lieberum et al., 2024).

References

Antoine, E., Béchet, F., and Langlais, P. Part-of-speech sensitivity of routers in mixture of experts models, 2024. URL https://arxiv.org/abs/2412.16971.

Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features.

Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates, 1988.

Cunningham, H., Ewart, A., Riggs, L., Huben, R., and Sharkey, L. Sparse autoencoders find highly interpretable features in language models, 2023. URL https://arxiv.org/abs/2309.08600.

Gadhikar, A., Majumdar, S. K., Popp, N., Saranrittichai, P., Rapp, M., and Schott, L. Attention is all you need for mixture-of-depths routing, 2024. URL https://arxiv.org/abs/2412.20875.

He, S., Ge, T., Sun, G., Tian, B., Wang, X., and Yu, D. Router-tuning: A simple and effective approach for enabling dynamic-depth in transformers, 2025. URL https://arxiv.org/abs/2410.13184.

Honnibal, M., Montani, I., Van Landeghem, S., and Boyd, A. spacy: Industrial-strength natural language processing in python. https://doi.org/10.5281/zenodo.1212303, 2020.

Laitenberger, F., Kopiczko, D. J., Snoek, C. G. M., and Asano, Y. M. What layers when: Learning to skip compute in LLMs with residual gates. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=aiP6XfaYZR.

Lawson, T. and Aitchison, L. Learning to skip the middle layers of transformers, 2025. URL https://arxiv.org/abs/2506.21103.

Lieberum, T., Rajamanoharan, S., Conmy, A., Smith, L., Sonnerat, N., Varma, V., Kramár, J., Dragan, A., Shah, R., and Nanda, N. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2, 2024. URL https://arxiv.org/abs/2408.05147.

Luo, Y., Luo, G., Ji, J., Zhou, Y., Sun, X., Shen, Z., and Ji, R. $\gamma-$mod: Exploring mixture-of-depth adaptation for multimodal large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=q44uq3tc2D.

Merity, S., Xiong, C., Bradbury, J., and Socher, R. Pointer sentinel mixture models, 2016.

Michel, P., Levy, O., and Neubig, G. Are sixteen heads really better than one?, 2019. URL https://arxiv.org/abs/1905.10650.

Raposo, D., Ritter, S., Richards, B., Lillicrap, T., Conway Humphreys, P., and Santoro, A. Mixture-of-depths: Dynamically allocating compute in transformer-based language models, 2024. URL https://arxiv.org/abs/2404.02258.

Tan, Z., Dong, D., Zhao, X., Peng, J., Cheng, Y., and Chen, T. Dlo: Dynamic layer operation for efficient vertical scaling of llms, 2024. URL https://arxiv.org/abs/2407.11030.

Team, G., Riviere, M., Pathak, S., Sessa, P. G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., Ferret, J., Liu, P., Tafti, P., Friesen, A., Casbon, M., Ramos, S., Kumar, R., Lan, C. L., Jerome, S., Tsitsulin, A., Vieillard, N., Stanczyk, P., Girgin, S., Momchev, N., Hoffman, M., Thakoor, S., Grill, J.-B., Neyshabur, B., Bachem, O., Walton, A., Severyn, A., Parrish, A., Ahmad, A., Hutchison, A., Abdagic, A., Carl, A., Shen, A., Brock, A., Coenen, A., Laforge, A., Paterson, A., Bastian, B., Piot, B., Wu, B., Royal, B., Chen, C., Kumar, C., Perry, C., Welty, C., Choquette-Choo, C. A., Sinopalnikov, D., Weinberger, D., Vijaykumar, D., Rogozińska, D., Herbison, D., Bandy, E., Wang, E., Noland, E., Moreira, E., Senter, E., Eltyshev, E., Visin, F., Rasskin, G., Wei, G., Cameron, G., Martins, G., Hashemi, H., Klimczak-Plucińska, H., Batra, H., Dhand, H., Nardini, I., Mein, J., Zhou, J., Svensson, J., Stanway, J., Chan, J., Zhou, J. P., Carrasqueira, J., Iljazi, J., Becker, J., Fernandez, J., van Amersfoort, J., Gordon, J., Lipschultz, J., Newlan, J., yeong Ji, J., Mohamed, K., Badola, K., Black, K., Millican, K., McDonell, K., Nguyen, K., Sodhia, K., Greene, K., Sjoesund, L. L., Usui, L., Sifre, L., Heuermann, L., Lago, L., McNealus, L., Soares, L. B., Kilpatrick, L., Dixon, L., Martins, L., Reid, M., Singh, M., Iverson, M., Görner, M., Velloso, M., Wirth, M., Davidow, M., Miller, M., Rahtz, M., Watson, M., Risdal, M., Kazemi, M., Moynihan, M., Zhang, M., Kahng, M., Park, M., Rahman, M., Khatwani, M., Dao, N., Bardoliwalla, N., Devanathan, N., Dumai, N., Chauhan, N., Wahltinez, O., Botarda, P., Barnes, P., Barham, P., Michel, P., Jin, P., Georgiev, P., Culliton, P., Kuppala, P., Comanescu, R., Merhej, R., Jana, R., Rokni, R. A., Agarwal, R., Mullins, R., Saadat, S., Carthy, S. M., Cogan, S., Perrin, S., Arnold, S. M. R., Krause, S., Dai, S., Garg, S., Sheth, S., Ronstrom, S., Chan, S., Jordan, T., Yu, T., Eccles, T., Hennigan, T., Kocisky, T., Doshi, T., Jain, V., Yadav, V., Meshram, V., Dharmadhikari, V., Barkley, W., Wei, W., Ye, W., Han, W., Kwon, W., Xu, X., Shen,

Z., Gong, Z., Wei, Z., Cotruta, V., Kirk, P., Rao, A., Giang, M., Peran, L., Warkentin, T., Collins, E., Barral, J., Ghahramani, Z., Hadsell, R., Sculley, D., Banks, J., Dragan, A., Petrov, S., Vinyals, O., Dean, J., Hassabis, D., Kavukcuoglu, K., Farabet, C., Buchatskaya, E., Borgeaud, S., Fiedel, N., Joulin, A., Kenealy, K., Dadashi, R., and Andreev, A. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Tenney, I., Das, D., and Pavlick, E. BERT rediscovers the classical NLP pipeline. In Korhonen, A., Traum, D., and Màrquez, L. (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL https://aclanthology.org/P19-1452/.

Voita, E., Talbot, D., Moiseev, F., Sennrich, R., and Titov, I. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned, 2019. URL https://arxiv.org/abs/1905.09418.

Xiao, G., Tian, Y., Chen, B., Han, S., and Lewis, M. Efficient streaming language models with attention sinks, 2024. URL https://arxiv.org/abs/2309.17453.

Zhang, C., Zhong, M., Wang, Q., Lu, X., Ye, Z., Lu, C., Gao, Y., Hu, Y., Chen, K., Zhang, M., and Song, D. Modification: Mixture of depths made easy, 2024. URL https://arxiv.org/abs/2410.14268.

A. Full per-POS effect tables

Tables 4 to 6 report the full per-POS routing analysis on strict-long MoD activations at each target layer, for all 16 POS tags with $n \geq 30$ tokens. Each entry shows the number of tokens with that POS, the empirical $P(\text{process} \mid \text{POS})$, the deviation from the layer's marginal routing rate $\Delta = P(\text{process} \mid \text{POS}) - \bar{P}(\text{process})$, and the binomial test p-value. The pre-committed threshold from Section 3 requires both Bonferroni significance at $\alpha' = 0.05/16 = 0.0031$ and effect size $|\Delta| \geq 5$pp; entries meeting both are marked ✓.

Table 4. Layer 13 (full attention, shallow). Marginal $\bar{P}(\text{process}) = 0.490$, $n_{\text{total}} = 31{,}229$.

POS	n	$P(\text{process} \mid \text{POS})$	Δ (pp)	p	Pass
ADJ	1,919	0.447	-4.4	$1.4 \cdot 10^{-4}$	
ADP	3,273	0.591	$+10.0$	$1.2 \cdot 10^{-30}$	✓
ADV	760	0.566	$+7.6$	$3.4 \cdot 10^{-5}$	✓
AUX	868	0.536	$+4.6$	$7.3 \cdot 10^{-3}$	
CCONJ	772	0.808	$+31.8$	$1.8 \cdot 10^{-74}$	✓
DET	2,812	0.377	-11.3	$1.6 \cdot 10^{-33}$	✓
NOUN	5,634	0.414	-7.7	$1.0 \cdot 10^{-30}$	✓
NUM	3,064	0.547	$+5.7$	$3.1 \cdot 10^{-10}$	✓
PART	628	0.465	-2.5	$2.2 \cdot 10^{-1}$	
PRON	872	0.412	-7.8	$3.9 \cdot 10^{-6}$	✓
PROPN	4,635	0.485	-0.5	$4.7 \cdot 10^{-1}$	
PUNCT	3,052	0.661	$+17.1$	$1.7 \cdot 10^{-80}$	✓
SCONJ	214	0.388	-10.2	$3.2 \cdot 10^{-3}$	
SPACE	165	0.000	-49.0	$6.7 \cdot 10^{-49}$	✓
SYM	65	0.415	-7.5	$2.6 \cdot 10^{-1}$	
VERB	2,468	0.361	-13.0	$2.1 \cdot 10^{-38}$	✓

Table 5. Layer 23 (full attention, deep). Marginal $\bar{P}(\text{process}) = 0.613$, $n_{\text{total}} = 31{,}229$.

POS	n	$P(\text{process} \mid \text{POS})$	Δ (pp)	p	Pass
ADJ	1,919	0.747	$+13.4$	$3.0 \cdot 10^{-35}$	✓
ADP	3,273	0.570	-4.3	$5.4 \cdot 10^{-7}$	
ADV	760	0.670	$+5.6$	$1.4 \cdot 10^{-3}$	✓
AUX	868	0.631	$+1.8$	$3.0 \cdot 10^{-1}$	
CCONJ	772	0.812	$+19.9$	$1.1 \cdot 10^{-32}$	✓
DET	2,812	0.729	$+11.5$	$9.3 \cdot 10^{-38}$	✓
NOUN	5,634	0.555	-5.9	$2.6 \cdot 10^{-19}$	✓
NUM	3,064	0.676	$+6.2$	$9.8 \cdot 10^{-13}$	✓
PART	628	0.699	$+8.6$	$9.1 \cdot 10^{-6}$	✓
PRON	872	0.507	-10.7	$2.1 \cdot 10^{-10}$	✓
PROPN	4,635	0.489	-12.5	$4.7 \cdot 10^{-66}$	✓
PUNCT	3,052	0.688	$+7.5$	$1.1 \cdot 10^{-17}$	✓
SCONJ	214	0.561	-5.3	$1.2 \cdot 10^{-1}$	
SPACE	165	1.000	$+38.7$	$1.2 \cdot 10^{-35}$	✓
SYM	65	0.415	-19.8	$1.3 \cdot 10^{-3}$	✓
VERB	2,468	0.546	-6.8	$8.4 \cdot 10^{-12}$	✓

Table 6. Layer 24 (sliding-window attention, deep). Marginal $\bar{P}(\text{process}) = 0.494$, $n_{\text{total}} = 31{,}229$.

POS	n	$P(\text{process} \mid \text{POS})$	Δ (pp)	p	Pass
ADJ	1,919	0.599	$+10.5$	$2.6 \cdot 10^{-20}$	✓
ADP	3,273	0.386	-10.8	$1.2 \cdot 10^{-35}$	✓
ADV	760	0.437	-5.7	$1.8 \cdot 10^{-3}$	✓
AUX	868	0.626	$+13.2$	$7.2 \cdot 10^{-15}$	✓
CCONJ	772	0.635	$+14.1$	$5.0 \cdot 10^{-15}$	✓
DET	2,812	0.636	$+14.2$	$1.3 \cdot 10^{-51}$	✓
NOUN	5,634	0.491	-0.3	$7.1 \cdot 10^{-1}$	
NUM	3,064	0.723	$+22.9$	$8.4 \cdot 10^{-146}$	✓
PART	628	0.457	-3.7	$6.6 \cdot 10^{-2}$	
PRON	872	0.416	-7.8	$4.6 \cdot 10^{-6}$	✓
PROPN	4,635	0.460	-3.4	$4.9 \cdot 10^{-6}$	
PUNCT	3,052	0.272	-22.2	$4.0 \cdot 10^{-138}$	✓
SCONJ	214	0.551	$+5.7$	$1.0 \cdot 10^{-1}$	
SPACE	165	0.170	-32.4	$4.7 \cdot 10^{-18}$	✓
SYM	65	0.323	-17.1	$6.1 \cdot 10^{-3}$	
VERB	2,468	0.438	-5.6	$2.7 \cdot 10^{-8}$	✓

B. SAE reconstruction quality on MoD activations

Sparse autoencoders trained on residual stream activations have become a standard tool for decomposing transformer representations into interpretable features (Bricken et al., 2023; Cunningham et al., 2023; Lieberum et al., 2024). We do not train SAEs in this work; per-feature analysis with Gemma Scope SAEs is left as future work (Section 5). As a methodology check, we verify that Gemma Scope canonical SAEs (release `gemma-scope-2b-pt-res-canonical`, width 16K) transfer to strict-long MoD residual-stream activations with modest reconstruction degradation relative to the SAE's training distribution (base Gemma 2 2B). Activations are taken at the canonical residual-stream hook `blocks.`ℓ`.hook_resid_post` for each target $\ell \in \{13, 23, 24\}$ on 200 WikiText-2 test samples (32,808 tokens after excluding the first 4 positions per sample, which are known to be out-of-distribution for Gemma Scope SAEs due to attention-sink behaviour (Xiao et al., 2024)).

Table 7. SAE reconstruction quality on strict-long MoD activations and a base-Gemma reference at layer 13. Cosine = mean per-token cosine similarity between residual stream and SAE reconstruction; v.e. (global) is pooled variance explained across all tokens; v.e. (per-feat.) is the mean of per-feature variance explained; L_0 = mean number of active features per token. The base Gemma layer-13 row is a one-off diagnostic measured on the SAE's training distribution; — denotes metrics not recorded for the reference run.

Setting	cosine	v.e. (global)	v.e. (per-feat.)	L_0	dead frac
strict-long MoD, L13	0.895	0.695	0.554	78.6	0.132
strict-long MoD, L23	0.876	0.652	0.622	115.5	0.009
strict-long MoD, L24	0.871	0.630	0.586	102.1	0.022
Base Gemma, L13 (ref.)	0.927	0.754	—	78.3	—

Cosine drops by $\sim$3pp on MoD activations relative to the in-distribution base-Gemma reference at layer 13; global variance explained drops by $\sim$6–12pp; L_0 remains in the 78–115 range expected for canonical Gemma Scope SAEs. Dead-feature fraction is highest at the shallow layer (13.2%) and drops below 3% at the deep layers, consistent with deeper layers recruiting broader feature support. The degradation is modest enough that per-feature analysis (left to future work, Section 5) is feasible on MoD activations using off-the-shelf Gemma Scope SAEs.

C. Pre-committed thresholds and revisions

This appendix documents the pre-committed thresholds for the three RQ2 controls and the POS probe, plus two pre-commit revisions made during the study, each with reasoning. All thresholds were recorded in project decision logs before the corresponding measurements were collected.

RQ2 control thresholds

(a) Layer-aggregate norm. *Threshold for "explains the preference":* shallow-middle norm Cohen's $d > 0$ in the *same* direction as the routing preference (full > sliding). *Rationale:* a router that preferred high-norm layers would incidentally preserve full-attention layers if and only if those layers had higher norms. *Outcome:* $d = -3.02$ in base Gemma 2 2B and $d = -3.86$ in the strict-long MoD context – opposite direction. Strongly rejected.

(b) Per-token norm dependence. *Threshold:* mean $|r| > 0.4$ across MoD layers between strict-long router score and per-token $\|h\|$ entering the router. *Rationale:* a Pearson r of 0.4 implies norm explains $\sim$16% of router-score variance per layer, the rough boundary above which "the router is mostly a norm-thresholder" would be substantively true. *Outcome:* mean $|r| = 0.166$, max $|r| = 0.336$ at layer 17 (at most $r^2 \approx 11\%$ of variance explained at any MoD layer). Below threshold; rejected.

(c) Per-layer task-loss asymmetry. *Threshold:* shallow-middle $\Delta(\text{full}) - \Delta(\text{sliding}) > 1.0$ PPL with $p < 0.05$, AND MLP-ablation specificity control flat (rules out a generic "deeper layers always cost more" confound). *Rationale:* a 1 PPL difference between attention types within shallow-middle would be a substantive layer-importance asymmetry defensible as a routing rationale. *Outcome:* $\Delta = +0.34$ PPL, $p = 0.44$, MLP control flat ($p = 0.88$). Below threshold; the test does not detect an asymmetry, with the power caveat noted in Section 4.2.

POS probe thresholds

Two complementary criteria, both pre-committed:

(A) Per-tag. At least one POS tag X at any target layer $\ell \in \{13, 23, 24\}$ satisfies $|P(\text{process} \mid \text{POS} = X) - \bar{P}_{\text{marginal}}| > 5\text{pp}$, Bonferroni-significant at $\alpha' = 0.05/n_{\text{tags}}$ where $n_{\text{tags}} = 16$ tags with $n \geq 30$ tokens.

(B) Aggregate probe. Logistic regression POS-onehot $\rightarrow$ router decision achieves test accuracy $>$ majority-class baseline $+5$pp at any target layer.

Either (A) or (B) clearing at any of layers $\{13, 23, 24\}$ counted as the probe-side criterion for workshop submission.

Outcome: (A) clears at all three layers (10/12/11 tags pass at L13/L23/L24 respectively); (B) clears at L13 ($+6.6$pp lift) and L24 ($+9.1$pp lift). At L23 (B) does not clear because the marginal $\bar{P} = 0.61$ leaves the binary classifier with insufficient operating range to register lift, even though per-tag asymmetries are large and pass (A). Both criteria met.

Pre-commit revisions

Revision 1: strict-long capacity band. *Original commit:* "strict-long overall skip $\in [0.40, 0.55] \rightarrow$ strict-long becomes the lead checkpoint; otherwise fall back to strict-short." *Result:* strict-long landed at 38.3% skip – 1.7pp below the band. *Revision:* strict-long promoted to lead despite the band miss, on substantive grounds: (i) strict-long has $\sim 10\times$ the training of strict-short (52K Alpaca samples $\times$ 3 epochs vs. 5K $\times$ 1) and is closer to the paper-spec configuration; (ii) strict-long has comparable per-layer balance to strict-short (10 of 13 layers in $[0.3, 0.7]$); (iii) 1.7pp below an intuited band is one unit of noise, not a substantively different routing regime. *Procedural lesson* (recorded with the revision): future pre-commits should attach decision bands to substantive criteria (e.g., "no layer in $\{23, 24\}$ degenerates" or "shallow-middle Cohen's $d > 0.5$") rather than arbitrary capacity windows.

Revision 2: SAE reconstruction-quality thresholds. *Original commit:* cosine $> 0.95 \rightarrow$ proceed; cosine in $[0.90, 0.95] \rightarrow$ proceed with caveat; cosine $< 0.90 \rightarrow$ pivot away from SAE-based analysis. Drawn from general SAE-literature intuition, not anchored to measured Gemma Scope performance. *Result:* a base-Gemma layer-13 diagnostic on the same evaluation pipeline gave cosine 0.927 – the original "proceed" threshold was unreachable even on the SAE's training distribution. *Revision:* thresholds re-anchored to measured base-Gemma performance. Excellent: cosine > 0.92 AND var. explained > 0.7 (base-Gemma parity); good: cosine > 0.85 AND var. explained > 0.5 (modest degradation, usable); pivot below cosine 0.85 OR var. explained 0.3. strict-long MoD landed in the "good" band (B). *Procedural lesson:* reconstruction-metric pre-commits should anchor to a measured ceiling on the in-distribution case before being set, not to a guessed expected quality.