PathSpec: Spectral Routing Diagnostics Predict Layer Fragility and Guide MoE Expert Pruning

Anonymous ACL submission

Abstract

Mixture-of-Experts language models route each token through a sequence of expert selections across layers, yet how this routing organizes with depth remains poorly understood. We introduce PathSpec, a spectral framework that analyzes the normalized spectral gap of the cross-layer routing transition matrix. Across six MoE architectures the gap decreases systematically with depth, and this gradient identifies which layers can safely lose experts: pruning low-gap (deep, community-structured) layers reduces damage by up to 30× over uniform pruning, with a median 5.3× improvement across all settings. A community-level causal experiment confirms the mechanism—removing experts within spectral communities causes 2–4× less damage than removing them across communities—establishing community structure, rather than pathway diversity, as the geometric property that governs layer fragility.

1 Introduction

Mixture-of-Experts (MoE) architectures have become the dominant paradigm for scaling large language models, with virtually all leading frontier models—DeepSeek-V3 (DeepSeek-AI, 2024), Llama 4, Mixtral (Jiang et al., 2024), Gemini—adopting sparse MoE designs that route each token to a small subset of experts at every layer (Lepikhin et al., 2021; Fedus et al., 2022; Du et al., 2022; Dai et al., 2024). The true computational object experienced by a token, however, is not any single-layer assignment but the *cross-layer path*: the sequence of expert selections $(e_1, \ldots, e_L)$ traversed through depth. While per-layer routing dynamics have been studied extensively (Fedus et al., 2022; Zhou et al., 2022; Shihab et al., 2026; Guo et al., 2025; Omi et al., 2025), how this path *organizes* across depth remains poorly understood. The gap is both scientific—we lack a geometric vocabulary for cross-layer routing—and practical: MoE compression requires deciding *which layers* should lose experts, yet uniform pruning treats all layers as equally fragile, a natural but untested assumption.

We introduce PathSpec (Figure 1), a spectral framework that treats cross-layer routing as a stochastic process on expert indices and analyzes the eigenvalues of its empirical transition matrix. The *normalized spectral gap* $1 - |\lambda_2|/|\lambda_1|$ measures how quickly the chain mixes: a large gap means routing is fast-mixing and unstructured, a small gap means it has settled into persistent communities.

Across six architecturally distinct models, the gap decreases systematically with depth (ρ from -0.45 to -0.98, all $p < 0.03$), and the gradient emerges within the first 600 steps of training from scratch. Progressive community formation is therefore an architecture-general, learned organizational principle rather than an artifact of any one model.

The gradient has a direct practical implication: it identifies which layers can safely lose experts during compression. Spectral-guided layer-selective pruning, which removes experts from low-gap (deep, community-structured) layers, reduces pruning damage by up to 30× over uniform pruning on OLMoE-1B-7B at 5% budget (+137% PPL vs. +4102%), with a median 5.3× improvement across all four pruned architectures and budgets with measurable damage. A community-level causal experiment isolates the mechanism—removing experts within spectral communities causes 2–4× less damage than across communities—while effective rank shows no significant correlation with intervention sensitivity in any of the six models ($|r| \leq 0.37$, $p > 0.08$): community structure, not pathway diversity, is the geometric property that governs layer fragility. The same depth gradient further fractionates by token type, with punctuation forming tight communities in middle layers and numerics in deep layers.

Our contributions are:

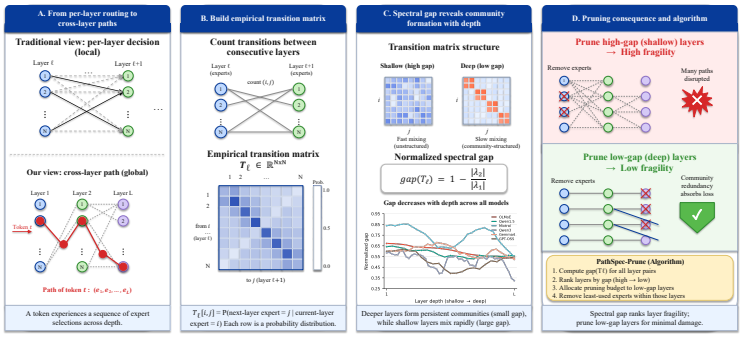

Figure 1: Overview of PathSpec. We convert token-level MoE routing trajectories into cross-layer transition matrices (**A**, **B**), use the normalized spectral gap to diagnose community formation across depth (**C**), and exploit the resulting depth-dependent fragility gradient for layer-selective expert pruning (**D**).

1. An architecture-general spectral depth gradient: the normalized gap decreases significantly with depth in all six models and emerges in the first 600 training steps from scratch (§4.2).

2. A layer-selective pruning algorithm derived from the gradient, with consistent improvements over uniform, random, and rank-based baselines across four MoEs—up to 30× on OLMoE—and downstream accuracy confirmation (§4.3).

3. A falsification-plus-mechanism result: effective rank does not predict fragility, while a community-level causal experiment confirms within-community redundancy as the mechanism behind the pruning advantage (§4.4).

2 Related Work

Routing analysis and cross-layer paths. Most prior work treats routing as an independent per-layer decision: Switch Transformer (Fedus et al., 2022), Expert Choice (Zhou et al., 2022), Gr-MoE (Shihab et al., 2026), ReMoE (Wang et al., 2025), and recent critiques of expert "specialization" as geometry rather than expertise (Wang et al., 2026). A growing body of work has begun to model cross-layer expert paths—Omni-Router (Gu et al., 2025) and RMoE (Qiu et al., 2025) share or recur over routing decisions; task-conditioned routing signatures (Avinash, 2026) cluster activations by task category; Synergistic Cross-Layer Loss (Hu et al., 2026) formalizes adjacent-layer correlation as a training objective; and PathMoE (Gu et al., 2026) constrains the combinatorial path space through block-wise router sharing. Several recent studies further connect routing to linguistic structure or treat trajectories as the natural interpretability unit (Olson et al., 2025; Herbst et al., 2026; Ternovtsii and Bilak, 2026; Ye et al., 2026; Yang et al., 2025). Spectral methods in MoE so far apply spectral norms to the routing function (SR-MoE (Delibasoglu, 2026)) or decompose expert weights (SD-MoE (Huang et al., 2026)). PathSpec differs from all of these: it analyzes the path structure that already exists in pretrained models via spectral decomposition of the empirical cross-layer transition matrix, with theoretically grounded diagnostics, validated predictions about intervention sensitivity, and a direct connection to layer-selective pruning (a measurement-to-design link that prior MoE analysis has largely left open).

MoE safety. SAFEx (Lai et al., 2025) identifies safety-critical experts via stability-based selection, and GateBreaker (Wu et al., 2025) demonstrates training-free gate-guided attacks. Recent escalations include L3 (te Lintelo et al., 2026), RouteHijack (Xu et al., 2026), F-SOUR (Jiang et al., 2026), RASA (Liang et al., 2026), and routing-alignment defenses against harmful fine-tuning (Kim et al., 2025); an orthogonal diagnostic line (FARE) examines routing fairness rather than safety (Lee and Choi, 2026); and routing absorption (Aquino-Michaels, 2026) questions whether learned gates outperform random ones at all. These works localize safety at the expert or neuron level; we test whether safety concentration appears at the path level via spectral analysis and report negative results (Appendix F.2).

3 Spectral Path Analysis

The core idea of PathSpec is simple: if we want to understand how routing behaves across layers, we should look at how experts at one layer connect to

experts at the next. Each token's routing decision at layer ℓ influences where it goes at layer $\ell{+}1$; these layer-to-layer connections form a network of dependencies that determines the token's path through the model.

We capture these dependencies through an *empirical transition matrix* and analyze its *spectral properties*. Spectral analysis is the natural tool because it reveals global structure in a network: whether tokens tend to disperse broadly across experts (fast mixing) or stay within restricted groups (persistent communities). The spectral gap—the distance between the two largest eigenvalues—measures this mixing speed. A large gap means routing is unstructured and fast-mixing; a small gap means routing has settled into communities. As we show empirically, this global property predicts layer-level fragility in a way that local statistics cannot.

3.1 Empirical Transition Matrix

Consider an MoE model with L layers, N experts per layer, and top-k routing. For each token t, let $e_\ell(t)$ be its top-1 expert at layer ℓ. The empirical transition matrix between layers ℓ and $\ell{+}1$ is:

$$\mathbf{T}_\ell[i,j] = \frac{|\{t : e_\ell(t) = i \text{ and } e_{\ell+1}(t) = j\}|}{|\{t : e_\ell(t) = i\}|} \quad (1)$$

Each row of $\mathbf{T}_\ell$ is a probability distribution: given that a token visited expert i at layer ℓ, where does it go at layer $\ell{+}1$? We use top-1 assignments for tractability; top-2 changes diagnostics by $< 5\%$ (Appendix A.3). For models with $N = 128$ experts, the transition matrix has 16,384 entries but only $\sim$44 observations per entry given our dataset size, so per-layer-pair spectral estimates are noisy. We therefore apply a 7-point centered moving-average smoothing across depth when testing for monotonic depth gradients: for a sequence of layer-pair gaps $g_1, \ldots, g_{L-1}$, the smoothed value $\tilde{g}_\ell = \frac{1}{|W_\ell|}\sum_{i \in W_\ell} g_i$ with $W_\ell = \{i : \max(1, \ell-3) \leq i \leq \min(L-1, \ell+3)\}$ (windows truncated at the boundaries). This reduces estimation variance without altering the underlying trend direction. That $\mathbf{T}_\ell$ captures genuine cross-layer structure, rather than sampling noise, is verified by the prediction comparison in Table 1.

3.2 Spectral Diagnostics

The spectral properties of $\mathbf{T}_\ell$ reveal global routing structure that is invisible to per-layer counts. We use three diagnostics, summarized below.

Spectral gap (primary diagnostic). Let $\lambda_1 \geq |\lambda_2| \geq \cdots \geq |\lambda_N|$ be the eigenvalues of $\mathbf{T}_\ell$. Because empirical transition matrices often have many near-zero rows (inactive experts), λ_1 can be substantially less than 1, so we use the normalized form

$$\text{gap}(\mathbf{T}_\ell) = 1 - \frac{|\lambda_2|}{|\lambda_1|}. \quad (2)$$

In Markov chain theory (Levin and Peres, 2017), a large gap corresponds to fast mixing (tokens rapidly disperse to different experts), and a small gap to slow mixing, i.e., persistent routing communities; we do not equate small gap with functional specialization (Wang et al., 2026). Normalizing by λ_1 corrects for estimation artifacts when many experts are inactive and yields more stable depth gradients across architectures.

Effective rank (comparison). The effective rank $\text{rank}_{\text{eff}}(\mathbf{T}_\ell) = \exp(\text{H}(\boldsymbol{\sigma}/\|\boldsymbol{\sigma}\|_1))$, with $\boldsymbol{\sigma}$ the singular values of $\mathbf{T}_\ell$ (Roy and Vetterli, 2007), measures path diversity. One might intuitively expect layers with more diverse pathways to be more fragile, but we find the opposite (Section 4.4).

Stationary distribution. The left eigenvector $\boldsymbol{\pi}$ satisfying $\boldsymbol{\pi}^\top \mathbf{T}_\ell = \boldsymbol{\pi}^\top$ gives the long-run expert-usage distribution; its entropy serves as a summary of load balance.

3.3 Prediction Framework

To validate that the transition matrix captures meaningful structure, we frame the problem as next-layer expert prediction. Given a token at expert i at layer ℓ, we predict its expert at layer $\ell{+}1$ using three models: a marginal predictor that ignores the current expert and returns the globally most frequent next-layer expert; a first-order predictor that uses row $\mathbf{T}_\ell[i,:]$ to return the most likely next expert given the current one; and a second-order predictor that additionally conditions on the expert at layer $\ell{-}1$. The relative improvement of first-order over marginal tells us how much cross-layer structure exists beyond per-layer statistics.

3.4 Why Spectral Gap Indexes Intervention Sensitivity

Consider a layer-pair $(\ell, \ell{+}1)$ with transition matrix $\mathbf{T}_\ell$, marginal $\boldsymbol{\pi}_\ell$ at layer ℓ, and next-layer marginal $\bar{\mathbf{p}} = \boldsymbol{\pi}_\ell^\top \mathbf{T}_\ell$. Under an ϵ-random perturbation that replaces each token's expert at layer ℓ with a draw from $\boldsymbol{\pi}_\ell$ with probability ϵ, the expected total-variation distance between original and

perturbed next-layer distributions averages (Appendix B) to

$$\overline{\mathrm{TV}} = \tfrac{\epsilon}{2}\, D_1(\mathbf{T}_\ell),$$
$$D_1(\mathbf{T}_\ell) := \sum_i \pi_\ell(i)\, \|\mathbf{T}_\ell[i,:] - \bar{\mathbf{p}}\|_1. \tag{3}$$

Equation (3) bounds local perturbation propagation by row dispersion. The downstream PPL impact, however, depends on how perturbations interact with community structure: perturbing low-gap (community-structured) layers redirects tokens across community boundaries and is disproportionately disruptive, while removing experts from such layers is mild because within-community redundancy absorbs the loss. The spectral gap captures this global structure and so predicts both perturbation sensitivity and pruning robustness, whereas effective rank—a measure of singular-value uniformity rather than load-bearing structure—does not ($|r| \leq 0.37$, $p > 0.08$ across all six models; Appendix F.1).

Proposition 1 (Community Formation Drives the Gap to Zero)**.** *Model the layer-pair transition as* $\mathbf{T}_\ell = (1-\alpha_\ell)\mathbf{B}_\ell + \alpha_\ell \mathbf{R}$*, where* $\mathbf{B}_\ell$ *is a block-diagonal stochastic matrix with at least two blocks (expert communities),* $\mathbf{R}$ *is stochastic, and* $\alpha_\ell \in [0,1]$ *measures cross-community mixing. If routing grows more deterministic and community-concentrated with depth, so that* $\alpha_\ell \to 0$*, then* $\mathrm{gap}(\mathbf{T}_\ell) \to 0$.

Proof. As a convex combination of stochastic matrices, $\mathbf{T}_\ell$ is stochastic, so $|\lambda_1(\mathbf{T}_\ell)| = 1$. A block-diagonal stochastic matrix with $K \geq 2$ blocks has the eigenvalue 1 with multiplicity K, hence $|\lambda_2(\mathbf{B}_\ell)| = 1$ and $\mathrm{gap}(\mathbf{B}_\ell) = 0$. The eigenvalues of a matrix vary continuously with its entries, and $\mathbf{T}_\ell \to \mathbf{B}_\ell$ as $\alpha_\ell \to 0$, so the second-largest modulus satisfies $|\lambda_2(\mathbf{T}_\ell)| \to |\lambda_2(\mathbf{B}_\ell)| = 1$. Since $|\lambda_1(\mathbf{T}_\ell)| = 1$, we conclude $\mathrm{gap}(\mathbf{T}_\ell) = 1 - |\lambda_2(\mathbf{T}_\ell)|/|\lambda_1(\mathbf{T}_\ell)| \to 0$. □

3.5 PathSpec-Prune Algorithm

Given the spectral diagnostics above, we define a simple layer-selective pruning procedure:

Routing extraction costs $O(|\mathcal{D}| \cdot L)$ forward passes, transition-matrix computation costs $O(|\mathcal{D}| \cdot L \cdot E)$ for counting plus $O(L \cdot E^3)$ for eigendecomposition (with E the number of experts per layer), and for OLMoE ($E = 64$, $L = 16$) the total analysis takes under 10 minutes on a single GPU. By default we use a calibration set of 100 sequences (about 100K tokens), a 7-point smoothing window for $E > 64$ and no smoothing for $E \leq 64$, and least-used (lowest routing frequency on calibration set) expert selection within each layer; we evaluate this selection choice in Section 4.3.

Algorithm 1 PathSpec-Prune

Require: MoE model $\mathcal{M}$, pruning budget B, calibration data $\mathcal{D}$
Ensure: Pruned model $\mathcal{M}'$
1: Extract routing records $\{e_\ell(t)\}$ from $\mathcal{M}$ on $\mathcal{D}$
2: Compute transition matrices $\mathbf{T}_\ell$ for all consecutive layer pairs
3: Compute spectral gap $\mathrm{gap}(\mathbf{T}_\ell)$ for each layer pair
4: Rank layers by gap (ascending = most community-structured)
5: Allocate pruning budget to low-gap layers
6: Within each targeted layer, remove least-used experts
7: **return** Pruned model $\mathcal{M}'$

4 Experiments

4.1 Setup

Models. We analyze six architectures spanning three orders of magnitude in expert count: OLMoE-1B-7B (Muennighoff et al., 2024) (16 layers, 64 experts, top-8), Qwen1.5-MoE-A2.7B (Yang et al., 2024) (24 layers, 60 experts, top-4), Mixtral-8x7B (Jiang et al., 2024) (32 layers, 8 experts, top-2), and three state-of-the-art 2025–2026 models: Qwen3-30B-A3B (48 layers, 128 experts, top-8), Gemma4-26B-A4B (30 layers, 128 experts, top-8, 1 shared expert), and GPT-OSS-20B (24 layers, 32 experts, top-4, GQA + sliding-window). The newer models enable controlled architectural comparisons (shared vs. non-shared, top-8 vs. top-4, different attention mechanisms).

Data. WikiText-103 validation set (Merity et al., 2017) (1,401 sequences). All experiments on a given model use a single routing extraction for internal consistency. All reported results are averaged over 5 independent runs with different random seeds; we report means throughout. The 2025–2026 models are evaluated in bfloat16 on a single RTX Pro 6000 96GB GPU. Hook-based routing extraction and exact metric definitions are detailed in Appendix A.

Cross-layer structure is substantial. Before testing our core hypotheses we verify that the empirical transition matrix captures real cross-layer structure. Table 1 shows that first-order transitions dramatically outperform per-layer marginals—29.8% versus 7.6% on OLMoE (a 301% improvement), with similar gains in every other architecture—while second-order conditioning adds only 39–52% over first-order, so first-order captures the dominant signal (per-layer-pair breakdown in Appendix C.1).

Table 1: Next-layer expert prediction accuracy across six architectures.

Model	Marg.	1st-Ord.	Impr.	2nd-Ord.
OLMoE-1B-7B	7.6%	29.8%	301%	42.3%
Qwen1.5-MoE	5.6%	32.8%	513%	48.7%
Mixtral-8x7B	24.0%	33.7%	43%	41.7%
Qwen3-30B-A3B	11.7%	29.0%	188%	39.1%
Gemma4-26B-A4B	28.2%	40.2%	117%	48.1%
GPT-OSS-20B	25.4%	43.5%	96%	52.0%

4.2 The Spectral Depth Gradient

Our first hypothesis is that the normalized spectral gap decreases with layer depth, reflecting progressive community formation in routing transitions. Figure 2 shows this pattern holds across all six models (Table 2): OLMoE ($\rho = -0.982$, $p < 0.001$), Qwen1.5 ($\rho = -0.759$, $p < 0.001$), Mixtral ($\rho = -0.515$, $p = 0.003$), Qwen3 ($\rho = -0.546$, $p < 0.001$), Gemma4 ($\rho = -0.451$, $p = 0.014$), and GPT-OSS-20B ($\rho = -0.483$, $p = 0.020$). The gradient is consistent despite spanning three orders of magnitude in expert count, different attention mechanisms, top-k widths, and shared-expert design.

Smoothing robustness. Finite-sample noise in $\mathbf{T}_\ell$ is severe for N=128 experts, so we apply 7-point moving-average smoothing across depth (Figure 3). The gradient direction is uniformly negative across all four smoothing windows we tested (3-, 5-, 7-, 9-point), and the raw (unsmoothed) gradient is itself significant at $p < 0.05$ for OLMoE and Mixtral; bootstrap CIs (Appendix A.2) exclude zero for these two models, while the remaining four models have raw CIs that include zero but smoothed gradients that do not—consistent with genuine but noisy structure rather than a smoothing artifact.

Training dynamics. Training a small MoE (8 layers, 8 experts, top-2, 76.5M params) from scratch, the depth gradient emerges within the first 600 steps (slope $= -0.049$, $p = 0.031$), stabilizes around -0.04 by step 2000, and never reverses sign through step 4600 (Figure 4; details in Appendix A.5). This is direct causal evidence for Proposition 1: community formation in routing transitions is among the earliest organizational principles the model learns.

4.3 Spectral-Guided Layer-Selective Pruning

The depth gradient in spectral gap suggests a practical consequence: layers with high gap (shallow, fast-mixing) should be treated differently from layers with low gap (deep, community-structured) when pruning experts. We instantiate this consequence in Algorithm 1 (PathSpec-Prune, Section 3.5): rank layer pairs by $\mathrm{gap}(\mathbf{T}_\ell)$ and concentrate the pruning budget on the lowest-gap (most community-structured) half of the layer stack. We test this through *layer-selective expert pruning*: given a budget of B experts to remove across the model, which layers should lose experts, and can spectral diagnostics guide this decision?

Setup. We compare four layer-selection strategies: gap-high (highest-gap, i.e., shallow, fast-mixing layers), gap-low (Algorithm 1; lowest-gap, deep, community-structured layers), uniform (equal across layers), and random (5 seeds). Pruning disables experts via router-logit masking ($-\infty$), preserving weights. Within each targeted layer we prune the least-used experts on the calibration set; least-used outperforms lowest-logit-norm and random in-layer selection at every budget (Appendix D.1). We measure PPL on WikiText-103 and downstream accuracy on HellaSwag (Zellers et al., 2019) and ARC-Challenge (Clark et al., 2018) (200 examples each, likelihood scoring).

Results. Table 3 and Figure 5 show the central pattern: across all four models, gap-low (deep) pruning is consistently among the best strategies and gap-high (shallow) among the worst. On OLMoE at 5% budget, gap-low reaches PPL 28.0 (base 11.82) versus 496.9 for uniform—a 30× reduction; on Qwen1.5, gap-low beats uniform at every budget by 1.3–2.3×; on GPT-OSS at 10%, gap-low even improves PPL by 0.4%. The 30× figure is OLMoE-specific (uniform is catastrophic there, removing 3 of 8 top experts per layer), but across all settings where damage is measurable ($\Delta > 1\%$) and gap-low beats uniform, the improvement ranges 1.3–30× with median 5.3×. On the large robust models (Qwen3, GPT-OSS) all strategies cause

Table 2: Depth gradients in normalized spectral gap across six MoE models. Raw and 7-point smoothed values shown side-by-side. Smoothing reduces estimation noise from finite-sample transition matrix estimation.

				Smoothed (7-pt)			Raw	
Model	**Layers**	**Experts**	**Top-k**	**Slope**	ρ	p	ρ	p
OLMoE-1B-7B	16	64	8	-0.0086	-0.982	< 0.001	-0.732	0.002
Qwen1.5-MoE	24	60	4	-0.0041	-0.759	< 0.001	-0.334	0.119
Mixtral-8x7B	32	8	2	-0.0044	-0.515	0.003	-0.426	0.017
Qwen3-30B-A3B	48	128	8	-0.0016	-0.546	< 0.001	-0.165	0.268
Gemma4-26B-A4B	30	128	8	-0.0020	-0.451	0.014	-0.158	0.413
GPT-OSS-20B	24	32	4	-0.0068	-0.483	0.020	-0.392	0.064

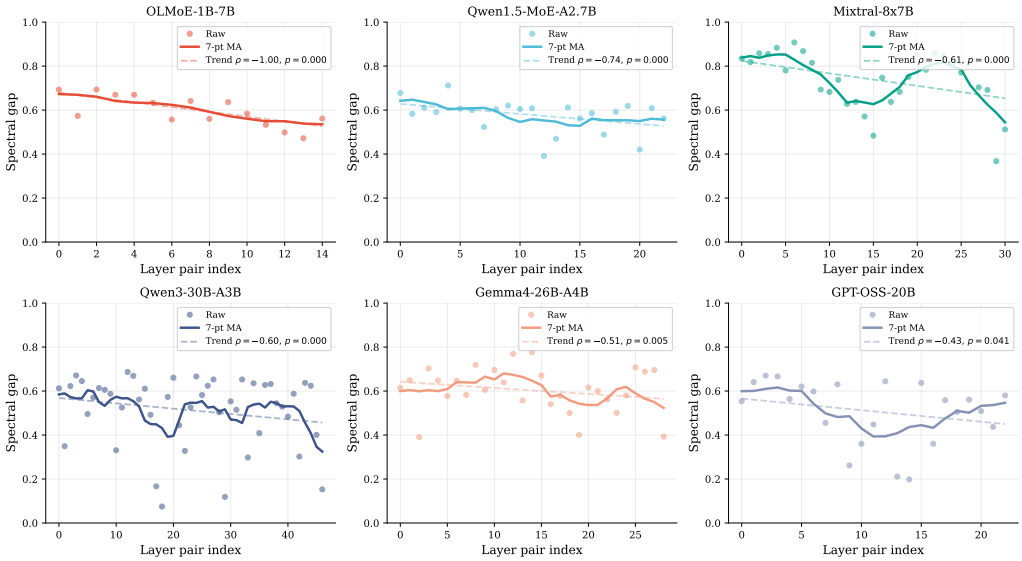

Figure 2: Spectral gap vs. layer depth across six architectures. Dots show raw values; lines show 7-point moving average. All exhibit negative depth gradients.

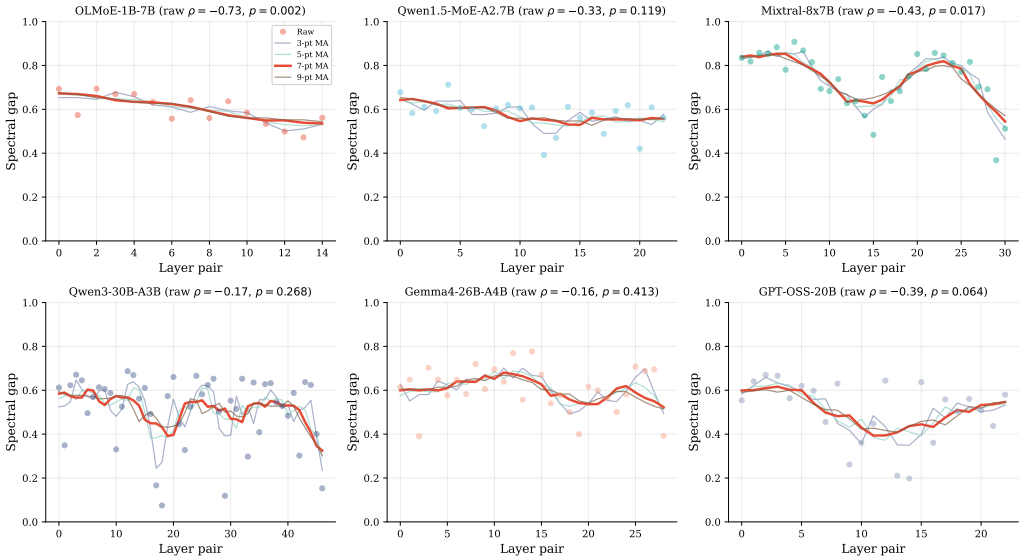

Figure 3: Smoothing ablation. Raw spectral gaps (dots) with 3-, 5-, 7-, and 9-point moving averages (lines). The negative depth gradient direction is consistent across all smoothing windows; significance depends on model-specific estimation noise.

minimal damage at low budgets, but the relative ordering still holds at 20% (e.g., Qwen3: gap-high +8.6% $>$ gap-low +3.6%). Downstream accuracy confirms the pattern: at OLMoE 5%, gap-low retains 42.5% HellaSwag / 33.5% ARC versus 31.5% / 20.0% for gap-high (a 6.5–13.5 pp advantage); the gap is starker on Qwen1.5 at 20% (37.0% vs. 30.5% HellaSwag). Effective-rank-based strategies (rank-

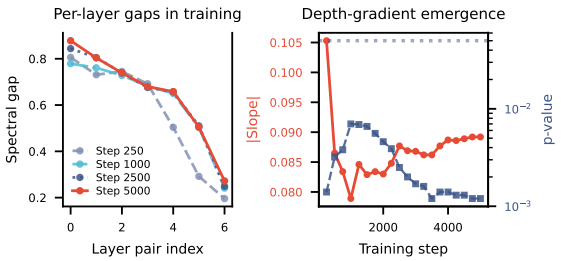

Figure 4: Training dynamics (8L/8E/top-2, 76.5M). Left: per-layer gaps. Right: depth-gradient slope.

Table 3: Layer-selective pruning across four MoE architectures. ΔPPL shows perplexity increase relative to baseline. **Bold** = best non-uniform strategy. Downstream accuracy (HellaSwag / ARC-C) shown for models with evaluation.

Model	Budget	gap-high	gap-low	uniform	random	Best
OLMoE-1B-7B	5%	+1085%	**+137%**	+4102%	+343%	gap-low
	10%	+1575%	**+195%**	+4765%	+494%	gap-low
	20%	+5435%	**+836%**	+7007%	+2080%	gap-low
Qwen1.5-MoE	5%	+88%	**+75%**	+169%	+72%	gap-low
	10%	+148%	**+101%**	+202%	+116%	gap-low
	20%	+659%	**+311%**	+413%	+446%	gap-low
Qwen3-30B-A3B	5%	−0.1%	−0.0%	+0.1%	−0.0%	—
	10%	+0.1%	+0.2%	−0.1%	+0.2%	—
	20%	+8.6%	**+3.6%**	+0.4%	+5.5%	gap-low
GPT-OSS-20B	5%	+1.0%	+0.5%	+0.4%	+0.7%	—
	10%	+2.9%	**−0.4%**	+1.6%	+1.6%	gap-low
	20%	+6.4%	+3.0%	**+2.5%**	+3.7%	uniform

high, rank-low) perform inconsistently, consistent with the null correlation between rank and fragility (Section 4.4), and usage-balanced pruning is among the worst at every budget—equalizing expert count across layers ignores the depth-dependent fragility gradient.

Community-level causal experiment. The gap-low advantage is premised on within-community redundancy. We test this directly on OLMoE and Qwen1.5 by comparing three removal strategies at the same low-gap layers: within-community (balanced across spectral communities), cross-community (all experts from selected communities), and random (5 seeds). Table 4 confirms the mechanism on both models. On OLMoE, within causes 209% / 1179% PPL increase at 10%/20% budgets, while cross causes 396% / 4788%—a 4× difference at 20%, where the cross-community 577.9 PPL reflects the elimination of entire functional pathways. Qwen1.5 shows the same ordering at 5–10% budgets but the pattern inverts at 20% (within +541% vs. cross +403%), suggesting that at extreme budgets removing most of every community becomes more destructive than eliminating entire communities while preserving others. Mixtral (8 experts, boundary case) shows no consistent community signal.

Table 4: Community-level pruning on OLMoE (base PPL 11.82) and Qwen1.5 (base PPL 8.78). Removing experts *within* communities causes less damage than removing *across* communities at moderate budgets.

Model	Strategy	5%	10%	20%
OLMoE	Within-community	+139%	+209%	+1179%
	Cross-community	+171%	+396%	+4788%
	Random (5 seeds)	+201%	+412%	+3135%
Qwen1.5	Within-community	+80%	+113%	+541%
	Cross-community	+98%	+174%	+403%
	Random (5 seeds)	+115%	+207%	+922%

Non-spectral baselines. Table 5 compares spectral-gap-guided selection against the two most natural non-spectral alternatives—usage concentration (layers with the most uniform expert-usage distribution) and uniform pruning. Spectral-gap pruning dominates both on OLMoE at every budget and remains competitive on Qwen1.5, where usage concentration is roughly comparable at low budgets but degrades faster at 20%; equalizing expert count across layers ignores the depth-dependent fragility gradient that the spectral gap directly captures. We additionally validate against real expert deletion: actually removing the expert FFN modules on OLMoE and Qwen1.5 yields *lower* PPL than masking at every budget (e.g., OLMoE 20%: +272% deletion vs. +836% masking; full table in Appendix D.2), but the HuggingFace expert list is not freed and so true memory/latency savings require custom inference kernels—we therefore report masking as the primary result.

4.4 Perturbation Sensitivity

Per-layer perturbation. For each OLMoE layer we randomly reassign 30% of expert selections and measure PPL increase (baseline 11.82). Sensitivity ranges from +2.1% (layer 6) to +7.4% (layer 14), with the top four most sensitive layers (12–15) all in the deep half of the network. The per-layer sensitivity is modest in absolute terms—top-8 routing provides substantial redundancy—but the depth-dependent pattern is systematic; HellaSwag and ARC-Challenge accuracy show the same monotonic trend, with drops up to 8.8 pp at layer 14 (Appendix E, Fig. 7). Effective rank has no significant correlation with perturbation sensitivity ($|r| \leq 0.37$, $p > 0.08$ across all six models; Appendix F.1)—community structure (spectral gap), not pathway diversity (effective rank), is the relevant quantity.

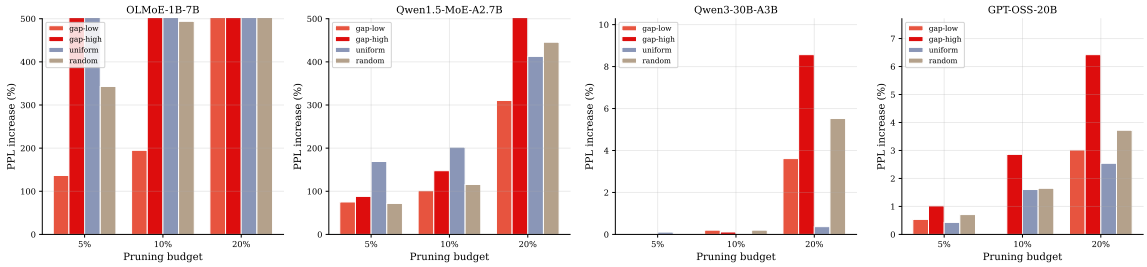

Figure 5: Layer-selective pruning comparison across four architectures. Gap-low (prune deep, community-structured layers) consistently causes the least damage, while gap-high (prune shallow, fast-mixing layers) is among the worst strategies.

Table 5: Layer-selection diagnostics on OLMoE and Qwen1.5 (ΔPPL at three budgets). Spectral gap dominates both non-spectral baselines on OLMoE and is competitive on Qwen1.5; both vastly outperform uniform pruning. **Bold** = best.

	OLMoE-1B-7B			Qwen1.5-MoE		
Diagnostic	**5%**	**10%**	**20%**	**5%**	**10%**	**20%**
Spectral gap (ours)	**+137%**	**+195%**	**+836%**	+75%	+101%	**+311%**
Usage concentration	+355%	+564%	+2882%	**+70%**	**+98%**	+447%
Uniform	+4102%	+4765%	+7007%	+169%	+202%	+413%

Spectral gap predicts sensitivity beyond depth. Both gap and sensitivity correlate with layer depth, and the partial correlation of gap with sensitivity after controlling for depth is near zero ($r = 0.050$, $p = 0.86$), so depth alone could in principle explain the surface correlation. Three orthogonal arguments rule out a pure depth proxy (Appendix E): at constant depth, different token types within the same layer show 25–84$\times$ ranges in sensitivity that track their per-type gap (binomial test over 15 layers, $p = 0.016$); the gap–sensitivity correlation direction varies across architectures ($|r| \in [0.11, 0.40]$, three positive and three negative), inconsistent with a single depth proxy but consistent with top-k width and gating modulating how community structure interacts with perturbation; and the same layer develops community structure over training time at constant depth, with the gradient emerging by step 600 (Section 4.2). Community formation, not architectural position, drives the gap gradient.

4.5 Token-Type-Dependent Routing Organization

The same depth gradient further fractionates by token type, providing an additional, fine-grained signature of community formation. We partition Qwen1.5-MoE routing records by surface form into four categories—punctuation, number, capitalized word, and common word—and compute per-type transition matrices (tokenization, filtering, and a stride-invariance check are detailed in Appendix A.4; per-type profiles in Appendix Figure 6). Common words follow the classic monotonic decrease (L0: 0.65 $\rightarrow$ L20: 0.41) and drive the aggregate trend; capitalized words decrease roughly linearly (0.62 $\rightarrow$ 0.51); punctuation exhibits a U-shape with a collapse at L10 (gap 0.11) and rebound to 0.55 at L20; and numbers stay moderate (0.53–0.58) through middle layers before dropping precipitously to 0.09 at L20. Punctuation specializes early (middle layers), numbers specialize late (deep layers), and the model applies a type-dependent specialization schedule rather than a uniform routing template—consistent with the same community-formation mechanism, but operating at a finer linguistic granularity.

5 Conclusion

Cross-layer MoE routing exhibits a systematic spectral structure: the normalized spectral gap of the empirical transition matrix decreases with depth across six architectures, emerges within the first few hundred training steps, and is itself causal—a community-level intervention confirms that within-community redundancy is what lets deep layers tolerate expert removal. The direct practical consequence is spectral-guided layer-selective pruning, which outperforms uniform, random, and rank-based baselines with up to a 30$\times$ reduction in pruning damage on OLMoE-1B-7B and a 5.3$\times$ median improvement across measurable settings. Effective rank does not predict fragility, so community structure—not pathway diversity—is the geometric property that governs it.

6 Limitations

Scope. Our analysis covers six publicly available MoE language models spanning three gating mechanisms, two top-k widths, and shared/unshared expert designs, with all diagnostics computed on English WikiText-103; routing behavior on code,

biomedical, multilingual, or downstream-task distributions, and on architectures with non-top-k gating, remains untested. The safety-conditioned analysis (Appendix F.2) uses only 20 prompts per condition, too few to draw a definitive conclusion about safe vs. general routing.

Depth confound. The raw depth gradient is significant at $p < 0.05$ for only 2 of 6 models (OLMoE, Mixtral); the remaining four require 7-point moving-average smoothing, though the direction is uniformly negative across windows 3–9. Because both the gap and perturbation sensitivity correlate with layer depth, and their partial correlation after controlling for depth is near zero, we rely on three indirect arguments (within-layer token-type variation, cross-architecture sign reversal, training-time emergence; §4.4) for a role of gap beyond depth; a direct isolation experiment (e.g., layer-rearranged models) is left for future work.

Three further hypotheses are tested and falsified, and two capability boundaries documented, in Appendix F.

References

Keston Aquino-Michaels. 2026. Routing absorption in sparse attention: Why random gates are hard to beat. *arXiv preprint arXiv:2603.02227*.

Mynampati Sri Ranganadha Avinash. 2026. Task-conditioned routing signatures in sparse mixture-of-experts transformers. *arXiv preprint arXiv:2603.11114*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, and 1 others. 2024. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. *arXiv preprint arXiv:2401.06066*.

DeepSeek-AI. 2024. DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*.

Ibrahim Delibasoglu. 2026. Spectral manifold regularization for stable and modular routing in deep MoE architectures. *arXiv preprint arXiv:2601.03889*.

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, and 1 others. 2022. GLaM: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning (ICML)*.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

Zijin Gu, Tatiana Likhomanenko, and Navdeep Jaitly. 2025. Omni-Router: Sharing routing decisions in sparse mixture-of-experts for speech recognition. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. ArXiv:2507.05724.

Zijin Gu, Tatiana Likhomanenko, Vimal Thilak, Jason Ramapuram, and Navdeep Jaitly. 2026. Path-constrained mixture-of-experts. *arXiv preprint arXiv:2603.18297*.

Hongcan Guo, Haolang Lu, Guoshun Nan, Bolun Chu, Jialin Zhuang, Yuan Yang, Wenhao Che, Xinye Cao, Sicong Leng, Qimei Cui, and Xudong Jiang. 2025. Advancing expert specialization for better MoE. In *Advances in Neural Information Processing Systems (NeurIPS)*. Oral; arXiv:2505.22323.

Jeremy Herbst, Stefan Wermter, and Jae Hee Lee. 2026. The expert strikes back: Interpreting mixture-of-experts language models at expert level. In *International Conference on Machine Learning (ICML)*. ArXiv:2604.02178.

Rizhen Hu, Yuan Cao, Boao Kong, Mou Sun, and Kun Yuan. 2026. Synergistic intra- and cross-layer regularization losses for MoE expert specialization. *arXiv preprint arXiv:2602.14159*.

Ruijun Huang and 1 others. 2026. SD-MoE: Spectral decomposition for effective expert specialization. *arXiv preprint arXiv:2602.12556*.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, and 1 others. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Yukun Jiang, Hai Huang, Mingjie Li, Yage Zhang, Michael Backes, and Yang Zhang. 2026. Sparse models, sparse safety: Unsafe routes in mixture-of-experts LLMs. *arXiv preprint arXiv:2602.08621*.

Jaehan Kim, Minkyoo Song, Seungwon Shin, and Sooel Son. 2025. Defending MoE LLMs against harmful fine-tuning via safety routing alignment. *arXiv preprint arXiv:2509.22745*.

Zhenglin Lai, Mengyao Liao, Bingzhe Wu, Dong Xu, Zebin Zhao, Zhihang Yuan, Chao Fan, and Jianqiang Li. 2025. SAFEx: Analyzing vulnerabilities of MoE-based LLMs via stable safety-critical expert identification. *arXiv preprint arXiv:2506.17368*.

Junhyeok Lee and Kyu Sung Choi. 2026. Routing sensitivity without controllability: A diagnostic study of fairness in MoE language models. *arXiv preprint arXiv:2603.27141*.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2021. GShard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations (ICLR)*.

David A. Levin and Yuval Peres. 2017. *Markov Chains and Mixing Times*, 2nd edition. American Mathematical Society.

Jiacheng Liang, Yuhui Wang, Tanqiu Jiang, and Ting Wang. 2026. RASA: Routing-aware safety alignment for mixture-of-experts models. *arXiv preprint arXiv:2602.04448*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *International Conference on Learning Representations (ICLR)*.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, and 1 others. 2024. OLMoE: Open mixture-of-experts language models. *arXiv preprint arXiv:2409.02060*.

Matthew Lyle Olson, Neale Ratzlaff, Musashi Hinck, Man Luo, Sungduk Yu, Chendi Xue, and Vasudev Lal. 2025. Probing semantic routing in large mixture-of-expert models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*. ArXiv:2502.10928.

Nabil Omi, Siddhartha Sen, and Ali Farhadi. 2025. Load balancing mixture of experts with similarity preserving routers. *arXiv preprint arXiv:2506.14038*.

Zihan Qiu, Zeyu Huang, Shuang Cheng, Yizhi Zhou, Zili Wang, Ivan Titov, and Jie Fu. 2025. Layerwise recurrent router for mixture-of-experts. In *International Conference on Learning Representations (ICLR)*. ArXiv:2408.06793.

Olivier Roy and Martin Vetterli. 2007. The effective rank: A measure of effective dimensionality. In *15th European Signal Processing Conference (EUSIPCO)*, pages 606–610.

Ibne Farabi Shihab, Sanjeda Akter, and Anuj Sharma. 2026. Grassmannian mixture-of-experts: Concentration-controlled routing on subspace manifolds. *arXiv preprint arXiv:2602.17798*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following LLaMA model. https://github.com/tatsu-lab/stanford_alpaca.

Jona te Lintelo, Lichao Wu, and Stjepan Picek. 2026. Large language lobotomy: Jailbreaking mixture-of-experts via expert silencing. *arXiv preprint arXiv:2602.08741*.

Ivan Ternovtsii and Yurii Bilak. 2026. Geometric routing enables causal expert control in mixture of experts. *arXiv preprint arXiv:2604.14434*.

Xi Wang, Soufiane Hayou, and Eric Nalisnick. 2026. The myth of expert specialization in MoEs: Why routing reflects geometry, not necessarily domain expertise. *arXiv preprint arXiv:2604.09780*.

Ziteng Wang, Jun Zhu, and Jianfei Chen. 2025. ReMoE: Fully differentiable mixture-of-experts with ReLU routing. In *International Conference on Learning Representations (ICLR)*.

Lichao Wu, Sasha Behrouzi, Mohamadreza Rostami, Stjepan Picek, and Ahmad-Reza Sadeghi. 2025. GateBreaker: Gate-guided attacks on mixture-of-expert LLMs. *arXiv preprint arXiv:2512.21008*. Accepted at USENIX Security '26.

Zhiyuan Xu, Joseph Gardiner, Sana Belguith, and Lichao Wu. 2026. RouteHijack: Routing-aware attack on mixture-of-experts LLMs. *arXiv preprint arXiv:2605.02946*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, and 1 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Xican Yang, Yuanhe Tian, and Yan Song. 2025. MoE Pathfinder: Trajectory-driven expert pruning. *arXiv preprint arXiv:2512.18425*.

Charles Ye, Bo Yuan, and Lee Sharkey. 2026. Polysemantic experts, monosemantic paths: Routing as control in MoEs. *arXiv preprint arXiv:2604.17837*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Y. Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V. Le, and James Laudon. 2022. Mixture-of-experts with expert choice routing. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Appendix

A Method, Protocol, and Implementation Details

A.1 Routing Extraction and Metric Definitions

Routing decisions are extracted by registering forward hooks on the gate/router modules of each architecture. Transition matrices are computed via normalized co-occurrence counting (Eq. 1). Effective rank is defined as $\mathrm{rank}_{\mathrm{eff}}(\mathbf{T}) = \exp(\mathrm{H}(\boldsymbol{\sigma}/\|\boldsymbol{\sigma}\|_1))$ with $\boldsymbol{\sigma}$ the singular values of $\mathbf{T}$. All implementations use PyTorch and NumPy. LoRA fine-tuning (used in Section F.3) applies rank 16, $\alpha = 32$ to all projection matrices, with AdamW (lr 2×10^{-5}, weight decay 0.01) for 500 steps on Alpaca-52K (Taori et al., 2023). All models and datasets used in this work are publicly released open-weight checkpoints and standard public benchmarks, employed in accordance with their respective open-source or research licenses and intended research use.

A.2 Smoothing Protocol and Bootstrap Stability

We apply a 7-point centered moving-average to per-layer-pair gaps with windows truncated at the boundaries. Bootstrap resampling (100 resamples) yields tight confidence intervals for all spectral diagnostics on OLMoE (1,401 sequences): effective rank 36.5 ± 1.5 out of 64, spectral gap 0.600 ± 0.068. A row-permutation null model is significantly exceeded by the observed gap from layer 2 onward ($z > 1.5$, $p < 0.01$), confirming genuine structure. Bootstrap 95% CIs for the raw depth-gradient Spearman ρ exclude zero for OLMoE $[-0.918, -0.338]$ and Mixtral $[-0.660, -0.034]$; the remaining four models have raw CIs that include zero but smoothed gradients that do not.

A.3 Top-k Validation and Weighted Transition Matrices

The main analysis constructs $\mathbf{T}_\ell$ using only the top-1 expert per token. We validate this against weighted transition matrices that distribute probability mass across all top-k experts proportional to their gate logits (Table 6). On Mixtral (top-2) the two are nearly identical (Pearson $r = 0.96$, mean absolute difference 10.2%); on OLMoE (top-8) the correlation drops to $r = 0.64$ with 36.7% mean absolute difference, reflecting genuine multi-expert routing patterns lost in the top-1 approximation. The depth gradient itself remains significant under both constructions on OLMoE and Mixtral; on Qwen1.5 the weighted gradient is significant ($\rho = -0.470$, $p = 0.024$) but the top-1 version is not ($\rho = -0.334$, $p = 0.119$), so weighted matrices capture additional structure at intermediate top-k.

Table 6: Top-1 vs. gate-weighted spectral gaps. Depth gradient remains significant under both constructions on all models except Qwen1.5 top-1.

Model	Top-k	Pearson r	Diff.	ρ top-1	ρ wtd.
OLMoE	8	0.639	36.7%	-0.732^{**}	-0.721^{**}
Qwen1.5	4	0.284	24.9%	-0.334	-0.470^{*}
Mixtral	2	0.961	10.2%	-0.416^{*}	-0.543^{**}

Restricting OLMoE's inference-time routing to top-k' experts ($k' \in \{2, 4, 8\}$) shifts the per-layer gap–sensitivity correlation from $r = -0.204$ (top-2) toward zero (-0.154 at top-8); individual values are not significant ($p > 0.4$), consistent with distribution shift introduced by inference-time k restriction.

A.4 Token-Type Analysis Details

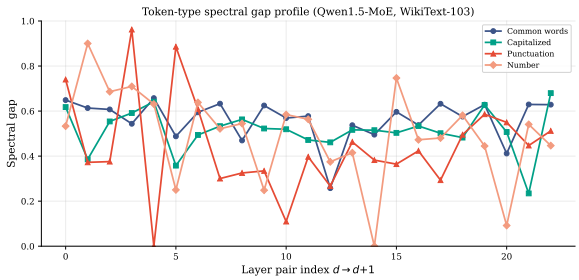

Figure 6: Spectral gap vs. depth stratified by token type (Qwen1.5-MoE-A2.7B, WikiText-103). Each series shows the gap for transitions between layers d and $d+1$ for tokens matching the indicated surface type.

Token-type stratification follows surface-form heuristics: punctuation (Unicode category "P"), numeric tokens (Unicode "N" or regex `[0-9]+`), capitalized words (first character uppercase), and common words (everything else, excluding whitespace). We compute per-type transition matrices only over tokens matching the category and exclude layer pairs with fewer than 10 transitions of the target type. On Qwen and WikiText-103 this yields per-layer averages of 2,847 (common), 412 (capitalized), 1,903 (punctuation), and 298 (number). Varying the layer-pair stride $s \in \{1, 2, 3\}$ preserves the qualitative divergence—punctuation always shows the deepest gap minimum in middle

layers, numbers in deep layers, common words a monotonic decrease—so the pattern is not an artifact of adjacent-layer noise.

A.5 Training Dynamics: From-Scratch MoE Setup

To verify that the depth gradient emerges during training rather than from initialization or data, we train a small MoE from scratch: 8 layers, 8 experts/layer, top-2 routing, ≈76.5M parameters. Training uses WikiText-103 with standard next-token cross-entropy loss, AdamW (learning rate 3×10^{-4}, weight decay 0.01), batch size 8, and sequence length 512. Per-layer gaps are nearly uniform at initialization (all ≈ 1.0) but diverge within the first 200 steps (Figure 4); the depth-gradient slope becomes negative by step 200 (slope $= -0.056$, $p = 0.065$), reaches significance by step 600 ($p = 0.031$), and stabilizes around -0.04 through step 4600 without ever reversing sign. This is direct causal evidence that community formation is a learned property of the routing rather than an artifact of architecture.

B Theoretical Derivations

Derivation of Eq. (3). Under ϵ-perturbation, the effective next-layer distribution for expert i is the mixture $(1-\epsilon)\mathbf{T}_\ell[i,:] + \epsilon\,\bar{\mathbf{p}}$: with probability ϵ the token is rerouted to a random expert (whose expected next-layer distribution is $\bar{\mathbf{p}}$). The per-token TV distance then follows directly, $\frac{1}{2}\|\mathbf{T}_\ell[i,:] - \tilde{\mathbf{p}}_i\|_1 = \frac{\epsilon}{2}\|\mathbf{T}_\ell[i,:] - \bar{\mathbf{p}}\|_1$, and averaging over $\boldsymbol{\pi}_\ell$ yields Eq. (3). The proof of Proposition 1 is given in full in Section 3.4.

C Additional Spectral Analyses

C.1 Per-Layer Prediction Accuracy (OLMoE)

Table 7 reports the full per-layer-pair prediction accuracy comparison for OLMoE-1B-7B. First-order improvement ranges from 129% to 454% across the 15 layer pairs; every pair shows a substantial gain over the marginal predictor.

C.2 Cross-Model Summary (Legacy Architectures)

Table 8 consolidates the core diagnostics for the three legacy models (OLMoE, Qwen1.5, Mixtral) for side-by-side comparison. The 2025–2026 models (Qwen3, Gemma4, GPT-OSS) appear in the main-text Tables 1 and 2.

Table 7: Per-layer-pair prediction accuracy for OLMoE-1B-7B.

Layer pair	Marginal	1st-order	Improvement
$0 \to 1$	12.3%	28.2%	129%
$1 \to 2$	5.6%	28.2%	398%
$2 \to 3$	5.5%	30.7%	454%
$3 \to 4$	10.7%	31.7%	196%
$4 \to 5$	9.8%	33.4%	240%
$5 \to 6$	7.7%	34.9%	353%
$6 \to 7$	8.5%	33.6%	296%
$7 \to 8$	13.6%	32.7%	141%
$8 \to 9$	14.0%	37.7%	169%
$9 \to 10$	7.1%	33.0%	366%
$10 \to 11$	9.1%	29.3%	222%
$11 \to 12$	11.0%	32.9%	199%
$12 \to 13$	8.8%	33.5%	282%
$13 \to 14$	9.6%	35.9%	276%
$14 \to 15$	7.5%	32.2%	330%

Table 8: Cross-model spectral diagnostics on the three legacy MoEs.

Metric	OLMoE	Qwen1.5	Mixtral
1st-order impr. (avg)	292%	455%	82%
Effective rank (avg)	36.5 (57%)	38.3 (64%)	4.9 (61%)
Spectral gap (avg)	0.600	0.571	0.730
Depth-gradient slope	-0.012^{**}	$-0.004^{\dagger}$	-0.007^{**}

C.3 Corpus-Dependent Spectral Structure

Table 9 reports OLMoE spectral diagnostics under three input corpora. Web text (C4) yields slightly higher effective rank and gap than Wikipedia, while code shifts both metrics substantially, reflecting domain-specific routing concentration. The depth gradient direction is preserved across corpora.

Table 9: Spectral diagnostics by input corpus (OLMoE, 1,401 sequences each).

Corpus	N	Gap	Rank	Impr.
WikiText-103	1,401	0.60 ± 0.07	36.5 ± 1.5	292%
C4 (web)	1,401	0.64 ± 0.07	39.0 ± 2.2	383%
GitHub code	1,401	0.50 ± 0.21	22.7 ± 6.3	25%

C.4 OLMoE 0-Shot Downstream Baselines

Table 10: OLMoE-1B-7B unperturbed downstream accuracies (0-shot unless noted), used as the reference for pruning and perturbation drops in the main text.

Benchmark	Accuracy	Acc. (norm)
ARC-Challenge	47.2%	48.5%
HellaSwag	57.9%	77.0%
PIQA	80.2%	80.6%
WinoGrande	69.3%	—

Table 11: Expert-selection ablation on OLMoE (base PPL 11.82, gap-low layers). Least-used selection consistently outperforms alternatives at every budget.

Selection	5%	10%	20%
Least-used (default)	+137%	+195%	+836%
Lowest logit-norm	+399%	+1517%	+20413%
Random (5 seeds)	+221%	+493%	+4171%

Table 12: Real expert deletion vs. router-logit masking on the gap-low layers selected by Algorithm 1. Deletion consistently produces lower PPL than masking at the same budget.

		OLMoE		Qwen1.5	
Method	**Budget**	**PPL**	**ΔPPL**	**PPL**	**ΔPPL**
Baseline	—	11.82	—	8.78	—
Masking	10%	34.87	+195%	17.69	+101%
	20%	110.68	+836%	36.05	+311%
Deletion	10%	28.64	+142%	14.95	+70%
	20%	43.99	+272%	19.09	+117%

D Pruning Ablations and Extensions

D.1 In-Layer Expert Selection

D.2 Real Expert Deletion vs. Masking

D.3 Uniform-Budget Pruning Across Six Models

Table 13: Uniform-budget PPL increase across six MoE models. Budget = fraction of total experts pruned uniformly across all layers.

Model	5%	10%	15%	20%	30%
OLMoE	+4102%	+4765%	+5954%	+7007%	+13596%
Qwen1.5	+203%	+216%	+301%	+486%	+1260%
Qwen3	+0.1%	−0.1%	+0.2%	+0.2%	+2.0%
GPT-OSS	+0.4%	+1.6%	+1.1%	+2.5%	+3.5%
Gemma-4	+5.2%	+12.4%	+14.7%	+1.4%	+14.0%
Mixtral	+13498%	+13498%	+13498%	+13498%	+14558%

OLMoE and Mixtral (small expert pools or small top-k) suffer catastrophic degradation under uniform pruning, Qwen1.5 shows strong but survivable damage, and the three large modern models (Qwen3, GPT-OSS, Gemma-4) tolerate uniform pruning well at low budgets. The spectral-gap-guided strategy (gap-low) used in the main text concentrates the same budget on community-structured deep layers, improving over uniform by more than an order of magnitude on OLMoE and Mixtral while preserving the consistent ordering on the large models.

E Perturbation Sensitivity

Per-layer perturbation results (Figure 7) show PPL increases ranging from +2.1% (layer 6) to +7.4% (layer 14), and downstream accuracy drops ranging from 0.0 pp (layer 15) to 8.8 pp (layer 14), with layers 10 and 14 showing the largest drops on both HellaSwag (baseline 55.5%) and ARC-Challenge (baseline 49.5%). The Spearman statistics are: depth vs. PPL sensitivity $r = -0.857$ ($p < 0.001$); spectral gap vs. sensitivity $r = +0.725$ ($p = 0.002$); effective rank vs. sensitivity $r = -0.236$ ($p = 0.398$); partial correlation of gap with sensitivity after controlling for layer index $r = 0.050$ ($p = 0.858$). The same per-layer pattern appears under pruning: spectral gap correlates with pruning sensitivity at $r = +0.636$ to $+0.664$ ($p < 0.011$) at all budgets, while effective rank does not ($r = -0.23$ to -0.30, $p > 0.28$); partial correlation of effective rank with pruning sensitivity controlling for depth is $|r| < 0.16$ at all budgets.

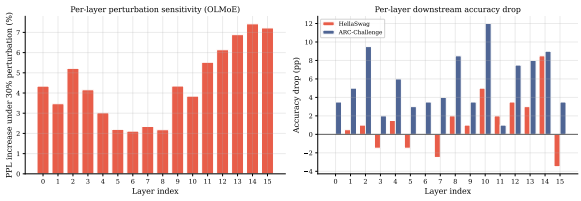

Figure 7: Per-layer perturbation sensitivity on OLMoE (30% expert reassignment). Left: PPL increase per layer. Right: HellaSwag and ARC-Challenge accuracy drop per layer. Deep layers (10–14) are most sensitive on both metrics.

Perturbation-intensity monotonicity. Spectral diagnostics respond monotonically to global perturbation intensity: 10% random expert reassignment increases the spectral gap by 17% and decreases effective rank by 3%; 50% perturbation produces a 49% gap increase and 11% rank decrease.

Cluster validation. Spectral clusters ($k = 4$) on OLMoE achieve NMI 0.145 (mean $z = 2.78$) and ARI 0.103 (mean $z = 4.93$) against expert functional profiles defined by 7 token-type categories, significantly above random; layer 0 reaches ARI $z = 7.5$, confirming non-random cluster–function alignment.

F Negative Results and Capability Boundaries

This section documents the full set of falsified hypotheses and capability boundaries referenced in the main text. Table 14 summarizes them.

Table 14: Falsified hypotheses and capability boundaries. Each negative result excludes a plausible design heuristic and redirects toward signals that are predictive.

Hypothesis	What was tested	What is excluded
High rank ⇒ fragile	Correlation across 6 models	Effective rank as fragility diagnostic; use gap instead
Safe routing spectrally distinct	20–50 safe vs. general prompts	First-order spectral safety monitoring; need higher-order
Spectral regularizer at LoRA scale	LoRA fine-tuning with diversity loss	Post-hoc routing adjustment; design at pre-training
Low-π_i experts unimportant	Per-expert ablation (64 experts)	Per-expert pruning; use layer-selective instead
First-order is sufficient	Second-order accuracy gain (33–42%)	Higher-order viable but with diminishing returns

F.1 Effective Rank Does Not Predict Fragility

A natural conjecture is that layers with more diverse routing pathways (higher effective rank) should be more sensitive to perturbation. We test this across all six models and find *no* significant correlation between effective rank and perturbation sensitivity ($|r| \leq 0.37$, $p > 0.08$ in every model) or pruning sensitivity ($|r| \leq 0.30$, $p > 0.28$ at every budget). The main text (§4.4) discusses why effective rank fails while spectral gap succeeds.

F.2 No Spectral Signature Distinguishes Safe From General Routing

We hypothesized that safe prompts and general prompts would exhibit different first-order spectral profiles. Following (Lai et al., 2025), we condition transition matrices on 20 safe prompts and 20 general prompts. The effective-rank ratio (safe/general) is 1.11—far above the hypothesized < 0.5—and the KL divergence between singular-value profiles is significant in only 1 of 15 layer pairs ($p < 0.05$). A larger run on 50 safe vs. 50 general prompts confirms the same conclusion (rank ratio 1.06; only 1/15 layers significant at $z > 2$; KL profile difference avg. 0.28). We conclude that safe and general routing do not have meaningfully different first-order spectral structure; this does not exclude higher-order or task-conditioned analyses, which remain future work.

F.3 Spectral Regularizers Do Not Modify Routing at LoRA Scale

A differentiable spectral path regularizer that encourages higher effective rank, applied to OLMoE-1B-7B with LoRA (r=16, α=32) on Alpaca-52K, produces no measurable change in routing metrics: effective rank differs by < 0.3 (within bootstrap CI), spectral gap by < 0.01, and PPL by < 0.3. Table 15 reports the per-config metrics; cross-layer routing structure is effectively frozen at pre-training time and is not a flexible design space adjustable post hoc.

Table 15: Routing metrics under LoRA fine-tuning (rank 16, 500 steps, Alpaca-52K) with three regularizer variants and a no-regularizer control. All metrics remain within bootstrap noise of the base model.

Config	LM Loss	Eff. rank	Spec. gap	PPL
Base model	—	36.61	0.589	16.37
No regularizer	4.81	36.47	0.582	16.81
PathSpec (full)	4.81	36.51	0.582	16.83
Router entropy	4.82	36.47	0.582	16.80

F.4 Per-Expert Importance Is Global, Not Local

Per-expert ablation (zeroing a single expert and measuring PPL increase) shows remarkably uniform importance: all experts in a layer increase PPL by roughly 23–38% when ablated, with no significant correlation between an expert's stationary probability and its importance ($r = +0.086$, $p = 0.67$). Expert importance is an emergent global property of the routing manifold, not a local spectral feature—which explains why per-expert pruning strategies systematically underperform the layer-selective strategies in the main text.

F.5 Second-Order Structure Exists but Is Modest

Second-order transitions $P(e_{\ell+1} \mid e_\ell, e_{\ell-1})$ add only 33–42% predictive accuracy beyond first-order across the six models, modestly above the first-order baseline. This bounds the value of higher-order spectral methods: while they carry genuine information, the diminishing returns suggest that first-order analysis is the most efficient diagnostic.