# WHO CONTRIBUTES? COUNTERFACTUAL ATTRIBUTION IN SPARSE MIXTURE-OF-EXPERTS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Sparse Mixture-of-Experts (MoE) models have become a key architecture for scaling large language models, yet understanding which experts truly contribute to model performance remains challenging. Existing measures such as routing frequency and gate weight describe expert usage, but overlook that MoE outputs are jointly produced by multiple experts whose contributions may be complementary or redundant. We introduce **CONTRAIL**, a counterfactual attribution framework that separately measures the contribution of selected expert computation and the utility of router decisions. For expert attribution, **CONTRAIL** models co-routed experts as a cooperative game and derives closed-form Shapley values from a quadratic approximation of the task loss, capturing both individual and interaction effects with a single backward pass. For router attribution, its Coalition-Consistent Gate Estimator evaluates a mass-conserving counterfactual swap between the nearest selected and excluded experts. We evaluate **CONTRAIL** on four instruction-tuned sparse MoE models using general, math and medical datasets. Across expert removal, cross-task intervention, router analysis, and sparse fine-tuning, **CONTRAIL** more reliably identifies consequential experts and routing decisions than routing frequency and existing attribution baselines, while revealing task-specific interactions and higher-impact adaptation targets.

## 1 INTRODUCTION

***Sparse Mixture-of-Experts (MoE)*** architectures have emerged as a key paradigm for scaling large language models (LLMs), underpinning leading models such as DeepSeek-V3 and Qwen3-235B-A22B (Dai et al., 2024; Yang et al., 2025). Their central advantage lies in conditional computation: rather than activating all model parameters for every token, an MoE model dynamically routes each token to only a small subset of specialized experts (Shazeer et al., 2017; Lepikhin et al., 2021; Fedus et al., 2022). This sparse activation decouples total model capacity from per-token computational cost, enabling models to scale their parameter capacity without a proportional increase in active computation. Under comparable active-parameter budgets, MoE models can outperform dense counterparts with advantage becoming more pronounced as total capacity increases.

***Despite this progress, identifying which experts truly matter remains challenging.*** Existing analyses often infer expert importance using routing statistics, such as how frequently an expert is selected or how much gate weight it receives. (Arnold et al., 2024; Zhou et al., 2022). However, these measures characterize expert ***usage***, rather than ***functional contribution***. An expert may be selected frequently or receive a large gate weight while providing information that is largely redundant with its co-routed experts. Conversely, an expert with a smaller gate weight may provide a complementary contribution that is important to the prediction. By overlooking this cooperative structure, usage-based measures cannot fully capture how experts jointly affect downstream performance.

This motivates a coalition-based view of expert attribution, in which an expert is evaluated through its marginal contribution across subsets of the experts selected on the same factual route. ***Shapley value*** (Shapley, 1953; Lundberg & Lee, 2017; Ghorbani & Zou, 2019; 2020) provides a principled way to implement this coalition-based attribution: treating the experts selected on the factual route as players, it assigns each expert a marginal contribution that accounts for both its individual effect and its interactions with co-routed experts. However, MoE behavior depends on both the computation performed by ***the selected experts*** and ***the router's choice of which experts to activate***. Removing



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

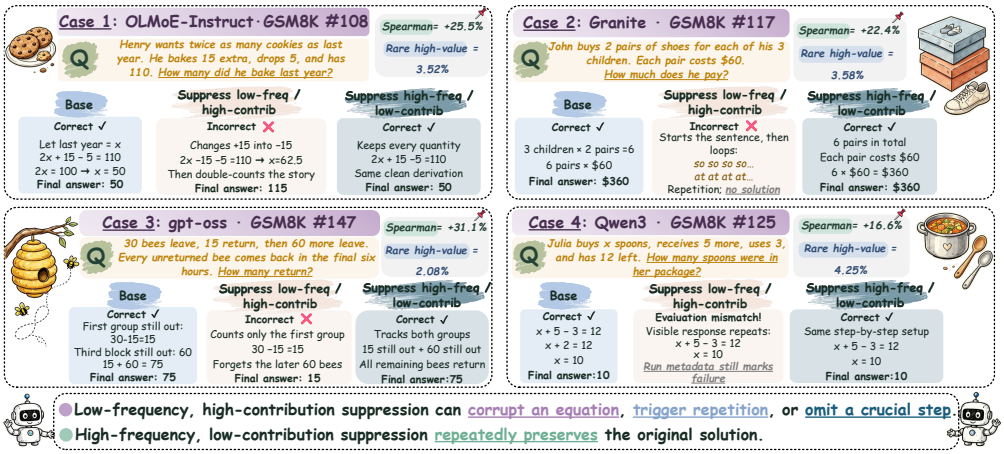


Figure 1: GSM8K cases. Detailed responses and the case-selection protocol appear in Section E.2.

an expert and rerouting the token changes both factors, making their effects difficult to disentangle. Expert computation and router selection must therefore be attributed separately.

To this end, we introduce **CONTRAIL**, a counterfactual attribution framework for sparse MoE models. It separately measures how each selected expert contributes to the task outcome and whether the router's selection is preferable to an alternative. These effects cannot be distinguished by conventional remove-and-reroute interventions, which change both the executed expert computation and the selected expert set. We therefore define two corresponding counterfactual attribution targets: ❶ ***Route-conditioned expert attribution.*** For each token and MoE layer, **CONTRAIL** models the selected expert outputs as a cooperative game conditioned on the factual route. To avoid exponential coalition enumeration, it approximates the task loss with a quadratic surrogate whose Shapley values admit an exact closed-form solution, capturing both individual and co-routed interaction effects with a single backward pass. ❷ ***Boundary-aware router attribution.*** **CONTRAIL** attributes the routing decision through a counterfactual swap between the lowest-logit selected expert and the highest-logit excluded expert. Its *Coalition-Consistent Gate Estimator (CCGE)* provides a mass-conserving estimate of the task-utility difference between the factual selection and its nearest alternative.

We evaluate **CONTRAIL** on four instruction-tuned MoE architectures across language modeling, mathematical reasoning, and knowledge-intensive tasks, demonstrating three practical applications: ❶ ***Expert identification and pruning.*** Under matched removal budgets, **CONTRAIL** more reliably identifies consequential expert computation than routing frequency and existing attribution baselines, including rare but high-contribution experts; ❷ ***Router diagnosis.*** CCGE identifies consequential routing decisions by measuring how local changes in top-$k$ membership affect downstream behavior. It provides a counterfactual basis for analyzing router sensitivity, enabling comparison with standard gradient-based estimators (Bengio et al., 2013); ❸ ***Task-specific sparse adaptation.*** Task-conditioned contribution scores identify the experts most relevant to a given objective, enabling targeted fine-tuning under limited trainable-parameter budgets.

Our contributions are threefold: ❶ We formulate sparse-MoE attribution around two explicit counterfactual targets, separating the functional contribution of factual expert computation from the task utility of discrete router selection. ❷ We derive closed-form, interaction-aware Shapley values for the local quadratic expert game and introduce a mass-conserving CCGE for the nearest-boundary router exchange. ❸ We validate these attribution scores across four MoE architectures through matched expert-removal, router-swap, interaction, cross-task, and sparse fine-tuning interventions.

## 2 Background and Problem Setup

### 2.1 Sparse Mixture-of-Experts Layers

Consider an MoE layer receiving a token representation $\boldsymbol{x} \in \mathbb{R}^{d_x}$ and an expert index set $\mathcal{E} = \{1, \dots, E\}$. A router produces logits $\boldsymbol{\ell}(\boldsymbol{x}) \in \mathbb{R}^E$ and the realized support $\mathcal{S}_k(\boldsymbol{x}) = \mathrm{TopK}(\boldsymbol{\ell}(\boldsymbol{x}), k)$. The expert game permits $1 \le k \le E$; the selected–excluded router game additionally requires $k < E$. We assume deterministic tie breaking and, whenever a local routing boundary is differentiated, a strict selected–excluded margin. Different MoE implementations normalize router

weights differently, so we write the post-routing weight abstractly as $g_e(\boldsymbol{x}) \geq 0$, with $g_e(\boldsymbol{x}) = 0$ for $e \notin \mathcal{S}_k(\boldsymbol{x})$. The router game additionally assumes that the boundary selected expert has positive mass, $g_a(\boldsymbol{x}) > 0$, as holds for the softmax-based gates studied here. If $\boldsymbol{f}_e(\boldsymbol{x}; \boldsymbol{\theta}_e) \in \mathbb{R}^{d_y}$ denotes the output of expert $e$, the factual MoE output is $\boldsymbol{y}^\star(\boldsymbol{x}) = \sum_{e \in \mathcal{S}_k(\boldsymbol{x})} g_e(\boldsymbol{x}) \boldsymbol{f}_e(\boldsymbol{x}; \boldsymbol{\theta}_e)$.

### 2.2 Shapley attribution

Let $N = \{1, \ldots, M\}$ be a finite player set and let $\mathcal{U} : 2^N \to \mathbb{R}$ be a characteristic function. For a uniformly random permutation $\pi$ of $N$, let $P_i^\pi$ denote the players preceding $i$. The Shapley value is equivalently written as $\phi_i(\mathcal{U}) = \sum_{C \subseteq N \setminus \{i\}} \frac{|C|!(M-|C|-1)!}{M!} \big[\mathcal{U}(C \cup \{i\}) - \mathcal{U}(C)\big] = \mathbb{E}_\pi[\mathcal{U}(P_i^\pi \cup \{i\}) - \mathcal{U}(P_i^\pi)]$. The value is invariant to adding a constant to $\mathcal{U}$ and satisfies the standard null-player, symmetry, linearity, and efficiency axioms. In particular, efficiency allocates the utility gain relative to the empty coalition: $\sum_{i \in N} \phi_i(\mathcal{U}) = \mathcal{U}(N) - \mathcal{U}(\varnothing)$.

## 3 CONTRAIL: Counterfactual Attribution Framework

**CONTRAIL** organizes sparse-MoE attribution around two targets that are conflated by ordinary rerouting ablations: the computation executed by selected experts and the discrete routing decision that selected them. The expert game is conditioned on the factual hard route and assigns utility to the gated outputs that were actually executed. The router game instead compares the factual route with the nearest one-swap support alternative at the top-$k$ boundary. Both games use the same downstream scalar attribution loss.

### 3.1 Route-conditioned expert attribution

Fix a probe example $x$ and a token representation $\boldsymbol{x}$ within that example, and write $\mathcal{S} = \mathcal{S}_k(\boldsymbol{x})$ for its factual selected experts. Let $d_y$ denote the dimension of the MoE output. The factual gated output of an active expert is $\boldsymbol{\delta}_e(\boldsymbol{x}) := g_e(\boldsymbol{x}) \boldsymbol{f}_e(\boldsymbol{x})$, and $\boldsymbol{y}^\star = \sum_{e \in \mathcal{S}} \boldsymbol{\delta}_e$. A coalition $C \subseteq \mathcal{S}$ retains exactly the factual contributions indexed by $C$, $\boldsymbol{y}_C := \sum_{e \in C} \boldsymbol{\delta}_e = \boldsymbol{y}^\star - \sum_{m \in \mathcal{S} \setminus C} \boldsymbol{\delta}_m$. Thus removing an expert means zeroing its factual gated contribution while holding the selected set, recorded gate weights, token input, and every other expert output fixed.

For every task, we measure expert contribution with respect to a scalar loss $\mathcal{L}_x$: the summed teacher-forced causal cross-entropy on its gold target tokens (defined precisely in Equation (78)). To make the downstream dependence explicit, let $\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y})$ denote the loss obtained by replacing the factual MoE output $\boldsymbol{y}^\star$ with $\boldsymbol{y}$, while keeping all other inputs and model parameters fixed and recomputing the downstream network normally. Because that downstream graph can itself contain hard top-$k$ routers, $\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y})$ is generally only piecewise smooth as $\boldsymbol{y}$ varies. Importantly, this does not affect the definition of the exact expert game: taking negative loss as utility, we define

$$\mathcal{U}_{\boldsymbol{x}}^{\exp}(C) := -\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}_C), \qquad C \subseteq \mathcal{S}. \tag{1}$$

Computing Shapley values directly from the nonlinear game in Equation (1) requires evaluating the loss under many expert coalitions. To obtain a tractable approximation, we locally approximate $\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y})$ around the factual output $\boldsymbol{y}^\star$ using a second-order Taylor expansion. Because $\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y})$ may be piecewise smooth due to downstream hard routing, this approximation applies locally within a downstream routing region where the required $C^2/C^3$ regularity conditions hold (Sections B.1 and B.9). Let $\boldsymbol{u}_\star := \nabla_{\boldsymbol{y}} \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star) \in \mathbb{R}^{d_y}$ and $\boldsymbol{H}_\star := \nabla_{\boldsymbol{y}}^2 \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star) \in \mathbb{R}^{d_y \times d_y}$, with $\boldsymbol{H}_\star = \boldsymbol{H}_\star^\top$. For $\boldsymbol{d} = \boldsymbol{y} - \boldsymbol{y}^\star$, define the second-order Taylor loss model $q_\star(\boldsymbol{d}) := \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star) + \boldsymbol{u}_\star^\top \boldsymbol{d} + \frac{1}{2} \boldsymbol{d}^\top \boldsymbol{H}_\star \boldsymbol{d}$, and the associated quadratic characteristic function

$$\mathcal{U}_{\boldsymbol{x},2}^{\exp}(C) := -q_\star(\boldsymbol{d}_C), \qquad \boldsymbol{d}_C := \boldsymbol{y}_C - \boldsymbol{y}^\star. \tag{2}$$

The nonlinear expert game admits an exact path-integral representation, but its marginal utility depends on curvature along the entire removal path. We obtain a tractable local game by freezing this curvature at the factual output $\boldsymbol{y}^\star$, yielding the quadratic characteristic function in Equation (2). Its

Shapley value(Xu et al., 2025) is exact for this quadratic game and takes the closed form

$$\begin{aligned}\phi_e^{(2)}(\boldsymbol{x}) &= \underbrace{-\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e}_{\text{direct}} + \underbrace{\frac{1}{2}\boldsymbol{\delta}_e^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e}_{\text{self-curvature}} + \underbrace{\frac{1}{2}\sum_{m\in\mathcal{S}\setminus\{e\}} \boldsymbol{\delta}_e^\top \boldsymbol{H}_\star \boldsymbol{\delta}_m}_{\text{co-routed interaction}} \\ &= -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e^\top \boldsymbol{H}_\star \Big(\sum_{m\in\mathcal{S}} \boldsymbol{\delta}_m\Big).\end{aligned} \tag{3}$$

The decomposed form separates the expert's first-order effect from its local self-curvature and its coupling with the other experts executed on the same route. The compact form is also computationally useful: the complete quadratic Shapley value requires only the aggregate routed displacement $\sum_{m\in\mathcal{S}} \boldsymbol{\delta}_m$, rather than explicit enumeration of coalitions or expert pairs.

To quantify pairwise interactions between co-routed experts, we define $\psi_{em}^{(2)}(\boldsymbol{x}) := \boldsymbol{\delta}_e^\top \boldsymbol{H}_\star \boldsymbol{\delta}_m$. The co-routed contribution to $\phi_e^{(2)}$ is therefore $\frac{1}{2}\sum_{m\in\mathcal{S}\setminus\{e\}} \psi_{em}^{(2)}$. The factor $1/2$ follows directly from Shapley averaging: for any pair $\{e, m\}$, $m$ appears after $e$ in a uniformly random permutation with probability $1/2$, so their pairwise quadratic effect is allocated equally between the two experts. Moreover, $\psi_{em}^{(2)}$ equals the excess quadratic loss increase from jointly removing $e$ and $m$ relative to removing them separately; positive values therefore indicate super-additive joint-removal damage, while negative values indicate sub-additivity. The exact nonlinear marginal, closed-form Shapley derivation, joint-removal identity, axioms, and approximation bounds are given in Sections B.2 to B.4, B.8 and B.9.

### 3.2 Boundary-aware router attribution

Route-conditioned expert attribution isolates the contribution of expert computation while holding the factual routing decision fixed. However, it does not capture the contribution of the routing decision itself—namely, which experts are selected into the top-$k$ support. We therefore introduce a complementary router attribution to quantify this selection decision.

Let $\boldsymbol{\ell} := \boldsymbol{\ell}(\boldsymbol{x}) \in \mathbb{R}^E$ be the factual router-logit vector, where $E$ is the number of experts. Let $a$ be the lowest-logit selected expert and $b$ the highest-logit excluded expert, i.e., the experts immediately inside and outside the top-$k$ selection boundary, respectively.

$$a := \arg\min_{e\in\mathcal{S}} \ell_e, \qquad b := \arg\max_{e\notin\mathcal{S}} \ell_e, \qquad \gamma := \ell_a - \ell_b > 0. \tag{4}$$

We assume $\gamma > 0$, excluding exact ties at the top-$k$ selection boundary. Let $\boldsymbol{v} := \boldsymbol{e}_a - \boldsymbol{e}_b$ where $\boldsymbol{e}_j$ is the $j$-th standard basis vector in $\mathbb{R}^E$. For an arbitrary logit vector $\boldsymbol{z} \in \mathbb{R}^E$, the routing decision between $a$ and $b$ switches when their logits are equal, $\boldsymbol{z}_a = \boldsymbol{z}_b$. Since $\boldsymbol{v}^\top \boldsymbol{z} = \boldsymbol{z}_a - \boldsymbol{z}_b$, the corresponding switching boundary is $\mathcal{H}_{ab} = \{\boldsymbol{z} \in \mathbb{R}^E : \boldsymbol{v}^\top \boldsymbol{z} = 0\}$. Its Euclidean distance from the factual logits is $\gamma/\sqrt{2}$, and the orthogonal projection onto this boundary is $\boldsymbol{\ell}^0 := \boldsymbol{\ell} - \frac{\gamma}{2}\boldsymbol{v}$. To obtain a counterfactual on the other side of the switching boundary, we reflect the factual logits $\boldsymbol{\ell}$ across the boundary: $\boldsymbol{\ell}^- := 2\boldsymbol{\ell}^0 - \boldsymbol{\ell} = \boldsymbol{\ell} - \gamma\boldsymbol{v}$. This reflection exchanges the logits of $a$ and $b$, so that $b$ replaces $a$ in the top-$k$ expert, while all other expert memberships remain unchanged under $\gamma > 0$ and the deterministic tie-breaking rule. Thus, "nearest-boundary" identifies the boundary pair; $\boldsymbol{\ell}^-$ provides a matched counterfactual routing decision on the opposite side of that boundary. In particular, $\boldsymbol{\ell}^-$ is the matched reflected endpoint rather than the closest point in the open neighboring routing region. The proof is in Section B.5.

**CONTRAIL** turns this geometric swap into a matched functional counterfactual. Let $\mu := g_a(\boldsymbol{x}) > 0$ be the factual gate mass of $a$. We transfer this mass from $a$ to $b$ while leaving all other selected contributions unchanged,

$$\boldsymbol{y}^- = \boldsymbol{y}^\star - \boldsymbol{\delta}_r, \qquad \boldsymbol{\delta}_r := \mu\big(\boldsymbol{f}_a(\boldsymbol{x}) - \boldsymbol{f}_b(\boldsymbol{x})\big). \tag{5}$$

Thus, $\boldsymbol{y}^-$ differs from $\boldsymbol{y}^\star$ only by replacing the contribution $\mu\boldsymbol{f}_a(\boldsymbol{x})$ with $\mu\boldsymbol{f}_b(\boldsymbol{x})$, allowing us to measure the value of choosing the factual boundary expert $a$ over its first excluded alternative $b$ without introducing changes in the other selected expert contributions.

The counterfactual output $\boldsymbol{y}^-$ defines the finite effect of swapping $a$ with $b$. However, hard top-$k$ selection is discrete and therefore does not provide an ordinary derivative for this support change. To

obtain a local differential representation of the same swap, we introduce the ***Coalition-Consistent Gate Estimator (CCGE)***, a backward surrogate that linearly transfers gate mass between $a$ and $b$. Treating the recorded $\mu$ and $\gamma$ as stop-gradient constants, its full logit-space Jacobian is

$$\boldsymbol{J}^{\mathrm{CCGE}} = \frac{\mu}{2\gamma}\boldsymbol{v}\boldsymbol{v}^\top = \frac{\mu}{2\gamma}(\boldsymbol{e}_a - \boldsymbol{e}_b)(\boldsymbol{e}_a - \boldsymbol{e}_b)^\top. \tag{6}$$

By construction, CCGE affects only the gate weights of $a$ and $b$; all other gate weights are locally unchanged. It conserves transferred gate mass ($\mathbf{1}^\top \boldsymbol{J}^{\mathrm{CCGE}} = 0$), where $\mathbf{1} \in \mathbb{R}^E$ is the all-ones vector, and is rank one and positive semidefinite. Its only nonzero principal block is:

$$\boldsymbol{J}^{\mathrm{CCGE}}_{\{a,b\},\{a,b\}} = \frac{\mu}{2\gamma}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \qquad [\boldsymbol{J}^{\mathrm{CCGE}}]_{pq} = 0 \text{ if } p \notin \{a,b\} \text{ or } q \notin \{a,b\}. \tag{7}$$

To connect the local CCGE surrogate to the finite counterfactual in Equation (5), consider the straight-line path from the factual logits $\boldsymbol{\ell}$ to the reflected logits $\boldsymbol{\ell}^-$: $\boldsymbol{\ell}(t) = \boldsymbol{\ell} - t\gamma\boldsymbol{v}$, $t \in [0,1]$, where $\boldsymbol{\ell}(0) = \boldsymbol{\ell}$ and $\boldsymbol{\ell}(1) = \boldsymbol{\ell}^-$. Along this path, CCGE linearly transfers the gate mass $\mu$ from $a$ to $b$, giving the pairwise gate weights $(\mu(1-t), \mu t)$. Let $\boldsymbol{y}^{\mathrm{CCGE}}(t)$ denote the MoE output induced by this surrogate gate path. Let $\boldsymbol{F} := [\boldsymbol{f}_1, \ldots, \boldsymbol{f}_E] \in \mathbb{R}^{d_y \times E}$ collect the expert outputs as columns. Since $\boldsymbol{J}^{\mathrm{CCGE}}$ maps logit changes to gate-weight changes and $\boldsymbol{F}$ maps gate-weight changes to output changes, the chain rule gives

$$\underbrace{\boldsymbol{F}}_{d_y \times E}\ \underbrace{\boldsymbol{J}^{\mathrm{CCGE}}}_{E \times E}\ \frac{\mathrm{d}\boldsymbol{\ell}(t)}{\mathrm{d}t} = -\frac{\mu}{2}\boldsymbol{F}\boldsymbol{v}\underbrace{(\boldsymbol{v}^\top\boldsymbol{v})}_{2} = -\mu\boldsymbol{F}\underbrace{\boldsymbol{v}}_{\boldsymbol{e}_a - \boldsymbol{e}_b} = -\mu\left(\boldsymbol{F}\boldsymbol{e}_a - \boldsymbol{F}\boldsymbol{e}_b\right) \equiv -\boldsymbol{\delta}_r \in \mathbb{R}^{d_y}. \tag{8}$$

Therefore, using $\boldsymbol{y}^{\mathrm{CCGE}}(0) = \boldsymbol{y}^\star$, $\boldsymbol{y}^{\mathrm{CCGE}}(t) = \boldsymbol{y}^\star - t\boldsymbol{\delta}_r$. At $t = 1$, $\boldsymbol{y}^{\mathrm{CCGE}}(1) = \boldsymbol{y}^\star - \delta_r = \bar{\boldsymbol{y}}$. Thus, integrating the local CCGE operator along the reflected-logit path recovers exactly the finite one-swap counterfactual in Equation (5). CCGE therefore provides a local differential representation that is calibrated to the same counterfactual endpoint, rather than an unrelated differentiable approximation to the hard router.

Having defined a matched counterfactual for the local routing decision, we can now assign attribution to that decision. We treat the choice of $a$ over $b$ as a single local player $r$. The factual state corresponds to selecting $a$, with output $\boldsymbol{y}^\star$, whereas the counterfactual state corresponds to selecting $b$ instead, with output $\bar{\boldsymbol{y}}$. Since this game contains only one player, its Shapley value reduces to the utility difference between these two states. Using negative loss as utility,

$$\phi_r^{\mathrm{nl}}(\boldsymbol{x}) = \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^-) - \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star), \quad \phi_r^{(2)}(\boldsymbol{x}) = -\boldsymbol{u}_\star^\top\boldsymbol{\delta}_r + \frac{1}{2}\boldsymbol{\delta}_r^\top\boldsymbol{H}_\star\boldsymbol{\delta}_r. \tag{9}$$

The first equation is exact; the second is its second-order local approximation. The corresponding path-integral identity and remainder are derived in Sections B.6 and B.9.

### 3.3 Sparse output-space estimation

The second-order expert and router attributions in Equation (3) and Equation (9) require local curvature information through $\boldsymbol{H}_\star$. Explicitly computing and storing the full Hessian in the MoE output space is expensive. We therefore use a positive-semidefinite gradient-outer-product surrogate.

*For each task and probe example* $x$, we fix $\mathcal{L}_x$ as the summed teacher-forced causal cross-entropy over the gold target tokens; task-specific masks are detailed in Section D.2. This reduction is part of the estimator specification: because the outer product is quadratic in the loss gradient, replacing the sum by a mean can change the relative scale of the first- and second-order terms rather than merely multiplying the final score by a constant. *For attributed unit* $i$, let $\mathcal{S}_i$ be its factual support, $\boldsymbol{\delta}_{i,e}$ the gated output of expert $e$, and $\boldsymbol{y}_i^\star := \sum_{e \in \mathcal{S}_i} \boldsymbol{\delta}_{i,e}$ the factual MoE output. Let $x(i)$ denote the corresponding probe example and treat $\mathcal{L}_{x(i),i}(\boldsymbol{y}_i)$ as the intervention slice obtained by varying this output row while holding all other inputs fixed. Define $\boldsymbol{u}_i := \nabla_{\boldsymbol{y}_i}\mathcal{L}_{x(i),i}(\boldsymbol{y}_i^\star)$, and let $\boldsymbol{p}_i, \boldsymbol{w}_i \in \mathbb{R}^{d_y}$ be arbitrary displacements. The surrogate is the rank-one outer product

$$\widehat{\boldsymbol{H}}_i^{\mathrm{EF}} := \boldsymbol{u}_i\boldsymbol{u}_i^\top = \left[u_{i,p}u_{i,q}\right]_{p,q=1}^{d_y} \in \mathbb{R}^{d_y \times d_y}. \tag{10}$$

We refer to this construction as "empirical-Fisher-style" only to describe this outer-product form. With a sequence-level summed objective it is the outer product of the aggregate sequence gradient, not the sum of per-target empirical-Fisher terms and not an identity with the exact Hessian.

For a mini-batch $\mathcal{B}$ of $n_{\mathcal{B}} := |\mathcal{B}|$ units, write $\boldsymbol{p} = (\boldsymbol{p}_i)_{i\in\mathcal{B}}$ and $\boldsymbol{w} = (\boldsymbol{w}_i)_{i\in\mathcal{B}}$. Its bilinear form is

$$\widehat{K}_{\mathcal{B}}(\boldsymbol{p}, \boldsymbol{w}) := \frac{1}{n_{\mathcal{B}}} \sum_{i\in\mathcal{B}} \boldsymbol{p}_i^\top (\boldsymbol{u}_i \boldsymbol{u}_i^\top) \boldsymbol{w}_i = \frac{1}{n_{\mathcal{B}}} \sum_{i\in\mathcal{B}} (\boldsymbol{u}_i^\top \boldsymbol{p}_i)(\boldsymbol{u}_i^\top \boldsymbol{w}_i). \tag{11}$$

Thus, curvature terms require only scalar projections onto $\boldsymbol{u}_i$, without materializing curvature matrices. Expert attribution uses projections of each active expert's gated output and of the total factual MoE output; router attribution additionally uses the excluded boundary expert $b$ and the one-swap displacement in Equation (5). Further details appear in Sections B.9, C.1, C.3 and C.4.

# 4 EXPERIMENTS

We organize the experiments as a five-step argument that moves from identification to application:

- ***RQ1: Is routing usage the same as functional contribution?***
- ***RQ2: Is expert contribution task-specific?***
- ***RQ3: Does boundary-aware attribution identify sensitive routing decisions?***
- ***RQ4: How much do co-routed interactions change expert importance?***
- ***RQ5: Are high-contribution experts the best sparse adaptation targets?***

Additional results and implementation details are provided in the Appendix (Sections D and E).

## 4.1 EXPERIMENTAL SETUP

❶ ***Models.*** We study OLMoE-1B-7B-Instruct (Muennighoff et al., 2025), Qwen3-30B-A3B-Instruct-2507 (Yang et al., 2025), gpt-oss-20b (OpenAI, 2025), and Granite-3.1-3B-A800M-Instruct (Granite Team, IBM, 2024). Model-specific routing details are summarized in the appendix (Table 9) and detailed in Section D.1. ❷ ***Datasets.*** WikiText-2 (WT-2) measures language modeling with perplexity (PPL); GSM8K (Cobbe et al., 2021), MMLU (Hendrycks et al., 2021), and MedQA (Jin et al., 2021) measure mathematical reasoning, broad knowledge, and medical knowledge with accuracy (ACC). Attribution probes and intervention evaluation use disjoint splits (Section D.2). GSM8K uses zero-shot chat prompting with greedy decoding and gpt-oss uses its Harmony template. ❸ ***Metrics.*** The main intervention tables report either absolute ACC/PPL or explicitly labeled metric changes. RQ2 uses positive accuracy damage $\Delta\mathrm{ACC}_{\mathrm{damage}}$, whereas RQ3 retains the signed metric change $\Delta\mathrm{ACC}_{\mathrm{change}}$; exact sign conventions are given in Section D.2.

## 4.2 RQ1: IS ROUTING USAGE THE SAME AS FUNCTIONAL CONTRIBUTION?

We distinguish routing usage from functional contribution by comparing expert rankings and their effects under matched removal interventions. As shown in Table 1, **CONTRAIL** performs strongly in the separate top-ranked output-zero and bottom-ranked logit-mask tests, each compared under method-matched budgets. For the Top 3% groups, output-zero removal preserves the factual route and directly tests the fixed-route expert counterfactual; **CONTRAIL** causes the greatest damage in most displayed settings. For the Bottom 10% groups, logit masking intentionally allows rerouting and tests whether low-contribution rankings remain lower-impact under a support-changing intervention. The corresponding Top 1% and Bottom 20% results, together with all additional baselines, are reported in Table 11.

## 4.3 RQ2: IS EXPERT CONTRIBUTION TASK-SPECIFIC?

RQ2 asks ***what the experts identified in RQ1 are important for***. Prior routing analyses find that expert assignments respond to linguistic context and input semantics (Arnold et al., 2024; Olson et al., 2025); here we ask the stronger functional question of whether the experts carrying high task utility also change with the downstream objective. We test this causally by transferring task-conditioned expert rankings across tasks. As shown in Table 2, removing experts selected by the evaluation task itself (*Own*) causes greater damage than the strongest ranking transferred from another task (*Off*) in 18 of 24 model–task–budget comparisons. The pattern is particularly clear for GSM8K, where Own dominates Off across all four backbones and both pruning budgets. Thus, high contribution is not

Table 1: Causal pruning under matched percentage budgets. Absolute performance after layer-wise removal. Top 3% uses output-zero; Bottom 10% uses logit-mask. Random is the five-seed mean. Additional baselines and top 1%/Bottom 20% are reported in Table 11.

| Method | OLMoE | | | Qwen3 | | | gpt-oss | | | Granite | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC |
| ***Unmodified checkpoint*** | | | | | | | | | | | | |
| Unmodified checkpoint | 13.73 | 0.525 | 0.487 | 7.23 | 0.933 | 0.760 | 25.57 | 0.894 | 0.698 | 7.41 | 0.602 | 0.505 |
| ***Top 3%: output-zero*** | ***(stronger damage: PPL↑, ACC↓)*** | | | | | | | | | | | |
| CONTRAIL (Ours) | 21.85 Δ +59.1% | 0.036 Δ -93.1% | 0.384 Δ -21.1% | 10.02 Δ +38.5% | 0.060 Δ -93.6% | 0.675 Δ -11.1% | 28.64 Δ +12.0% | 0.743 Δ -16.9% | 0.659 Δ -5.6% | 14.18 Δ +91.4% | 0.011 Δ -98.2% | 0.282 Δ -44.2% |
| Shapley-MoE | 19.56 Δ +42.5% | 0.180 Δ -65.8% | 0.413 Δ -15.2% | 9.92 Δ +37.2% | 0.107 Δ -88.5% | 0.682 Δ -10.2% | 26.69 Δ +4.4% | 0.783 Δ -12.4% | 0.663 Δ -5.0% | 12.51 Δ +68.8% | 0.145 Δ -75.9% | 0.329 Δ -34.8% |
| Routing frequency | 17.14 Δ +24.8% | 0.035 Δ -93.3% | 0.332 Δ -31.8% | 9.53 Δ +31.7% | 0.295 Δ -68.4% | 0.710 Δ -6.5% | 26.35 Δ +3.1% | 0.712 Δ -20.4% | 0.634 Δ -9.2% | 16.76 Δ +126.2% | 0.017 Δ -97.2% | 0.337 Δ -33.3% |
| Random | 14.23 Δ +3.6% | 0.515 Δ -1.9% | 0.480 Δ -1.4% | 7.46 Δ +3.1% | 0.930 Δ -0.4% | 0.753 Δ -0.9% | 22.14 Δ -13.4% | 0.876 Δ -2.0% | 0.673 Δ -3.6% | 8.61 Δ +16.2% | 0.457 Δ -24.1% | 0.439 Δ -13.1% |
| ***Bottom 10%: logit-mask*** | ***(better preservation: smaller absolute change from checkpoint)*** | | | | | | | | | | | |
| CONTRAIL (Ours) | 13.60 Δ -0.9% | 0.529 Δ +0.8% | 0.454 Δ -6.8% | 7.23 Δ -0.05% | 0.927 Δ -0.7% | 0.758 Δ -0.2% | 25.37 Δ -0.8% | 0.892 Δ -0.2% | 0.675 Δ -3.3% | 7.68 Δ +3.6% | 0.593 Δ -1.5% | 0.435 Δ -13.9% |
| Shapley-MoE | 14.49 Δ +5.6% | 0.459 Δ -12.5% | 0.448 Δ -8.0% | 7.30 Δ +0.9% | 0.924 Δ -1.0% | 0.753 Δ -0.9% | 26.45 Δ +3.4% | 0.865 Δ -3.2% | 0.669 Δ -4.2% | 8.16 Δ +10.1% | 0.487 Δ -19.1% | 0.429 Δ -15.1% |
| Routing frequency | 13.86 Δ +0.9% | 0.530 Δ +1.0% | 0.463 Δ -4.9% | 7.24 Δ +0.09% | 0.932 Δ -0.09% | 0.751 Δ -1.1% | 25.22 Δ -1.4% | 0.887 Δ -0.8% | 0.671 Δ -3.9% | 7.91 Δ +6.7% | 0.603 Δ +0.2% | 0.438 Δ -13.3% |
| Random | 15.72 Δ +14.5% | 0.340 Δ -35.2% | 0.441 Δ -9.4% | 7.89 Δ +9.1% | 0.885 Δ -5.2% | 0.720 Δ -5.2% | 27.76 Δ +8.6% | 0.814 Δ -8.9% | 0.658 Δ -5.7% | 9.09 Δ +22.7% | 0.281 Δ -53.3% | 0.456 Δ -9.7% |

Table 2: Task-specific causal structure. Each cell reports $\Delta\mathrm{ACC}_{\mathrm{damage}}$ for the own-task ranking (Own) and the off-task ranking that causes the larger damage among the other two tasks (Off). Larger values mean larger causal damage.

| Budget / source | OLMoE | | | Qwen3 | | | gpt-oss | | | Granite | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MMLU | GSM8K | MedQA | MMLU | GSM8K | MedQA | MMLU | GSM8K | MedQA | MMLU | GSM8K | MedQA |
| **Top 1% Own** | 0.0821 | 0.3844 | 0.0338 | 0.0280 | 0.1403 | 0.0660 | 0.0115 | 0.0159 | 0.0503 | 0.1023 | 0.5785 | 0.0660 |
| Top 1% Off | 0.0382 | 0.1948 | 0.0471 | 0.0210 | 0.0740 | 0.0500 | 0.0318 | 0.0099 | 0.0157 | 0.0866 | 0.2009 | 0.0518 |
| **Top 3% Own** | 0.1030 | 0.4890 | 0.0479 | 0.0710 | 0.8734 | 0.1100 | 0.0388 | 0.1509 | 0.0299 | 0.2228 | 0.5906 | 0.0903 |
| Top 3% Off | 0.0836 | 0.3260 | 0.0518 | 0.0560 | 0.3650 | 0.0840 | 0.0775 | 0.0842 | 0.0597 | 0.1716 | 0.4693 | 0.1579 |

merely a model-wide property: experts identified for a task tend to be especially important for that task's computation. The structural results explain where this task dependence comes from. Figure 2 shows that changing the downstream task redistributes contribution across layers and experts, while Table 3 quantifies how much the resulting high-value expert sets are shared. GSM8K has little overlap with either MMLU or MedQA, consistent with its strong Own–Off causal separation. In contrast, MMLU and MedQA share a much larger fraction of their top experts, which is consistent with the smaller and occasionally reversed cross-task gaps observed between the knowledge-oriented tasks. Together, these results show that expert contribution is substantially task-conditioned: some expert support is shared across related tasks, but the experts that matter most depend on the downstream computation being attributed. Full cross-task results are reported in Section E.3.

Table 3: Jaccard ranges for task-conditioned top-expert sets, taken across four models and 1% and 3% budgets (Figure 5 for cell-level similarity heatmaps).

| Task pair | Jaccard range | Structure |
|---|---|---|
| MMLU–GSM8K | 0.049–0.133 | Low |
| GSM8K–MedQA | 0.067–0.171 | Low |
| MMLU–MedQA | 0.455–0.641 | High |

### 4.4 RQ3: DOES BOUNDARY-AWARE ATTRIBUTION IDENTIFY ROUTING DECISIONS?

For every attributed token and MoE layer, all methods are scored on the same lowest-logit selected / highest-logit excluded boundary pair and are then aggregated into a layer score. We intervene on the top three layers selected by each score using the same nearest-boundary one-swap endpoint, so the comparison measures ranking quality rather than intervention strength. The five estimators, including two straight-through estimator (STE) baselines (Bengio et al., 2013), their backward operators, and the signed layer aggregation rule are defined precisely in Section D.5. A unified side-by-side comparison of their backward Jacobians, induced output displacements, and token-score forms is provided in Table 10. Table 4 shows that the full CCGE score ranks first in 10 of 12 model–task set-

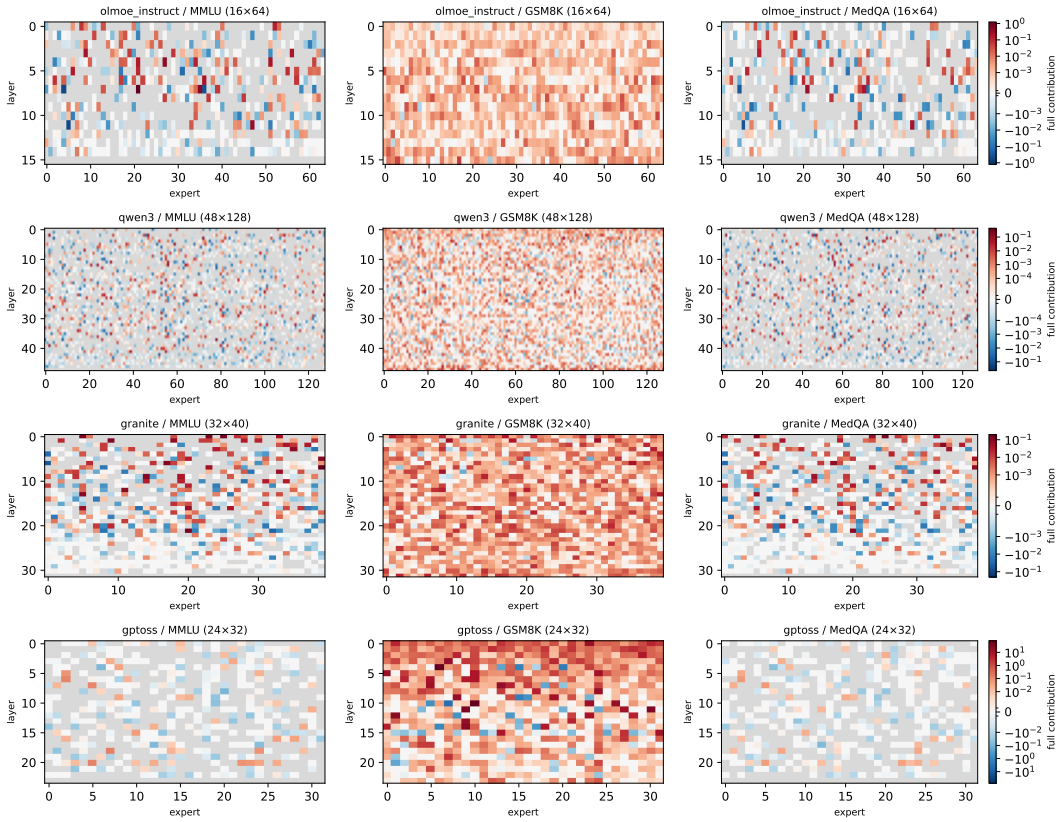


Figure 2: Task-conditioned contribution heatmaps (gray indicates an inactive expert).

Table 4: Router attribution under matched nearest-boundary one-swap interventions. Signed metric change after applying the same one-swap endpoint to the three layers selected by each score. The arrow points toward greater damage. Best results are in **bold** and second-best results are underlined.

| | OLMoE | | | Qwen3 | | | gpt-oss | | | Granite | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | WT-2 ↑ | GSM8K ↓ | MMLU ↓ | WT-2 ↑ | GSM8K ↓ | MMLU ↓ | WT-2 ↑ | GSM8K ↓ | MMLU ↓ | WT-2 ↑ | GSM8K ↓ | MMLU ↓ |
| Hard-route gradient | -0.0001 | -0.0129 | -0.0001 | +0.0095 | -0.0026 | -0.0005 | +1.8819 | -0.0076 | -0.0048 | +0.0072 | **-0.0197** | -0.0002 |
| Identity STE | +0.0206 | -0.0182 | -0.0023 | +0.0093 | -0.0032 | -0.0008 | +5.0822 | -0.0061 | -0.0039 | +0.0108 | -0.0106 | -0.0044 |
| Sigmoid STE | +0.0204 | -0.0176 | -0.0021 | +0.0094 | -0.0031 | -0.0008 | +4.9822 | -0.0064 | -0.0041 | +0.0119 | -0.0144 | -0.0053 |
| CCGE-linear | +0.0188 | **-0.0201** | -0.0027 | +0.0105 | -0.0038 | -0.0010 | +4.6745 | -0.0070 | -0.0007 | +0.0092 | -0.0158 | -0.0052 |
| **CCGE (Ours)** | **+0.0285** | -0.0194 | **-0.0031** | **+0.0112** | **-0.0042** | **-0.0012** | **+5.6400** | **-0.0091** | **-0.0055** | **+0.0136** | -0.0172 | **-0.0061** |

tings and second in the remaining two. It is strongest on all three Qwen3 tasks and all three gpt-oss tasks. CCGE-linear is strongest in one setting and second-best in four, indicating that aligning the backward signal with the nearest selected–excluded boundary already captures substantial routing sensitivity; the curvature-aware term further improves the ranking in most settings.

### 4.5 RQ4: How much do co-routed interactions change expert importance?

RQ4 isolates the co-routed pair term by comparing two otherwise matched corpus-level signals. Contribution uses $\Phi_e^{\text{corpus}}$, whereas Self$\times\rho$ uses $\rho_e \Phi_{e,\text{self}}^{\text{cond}}$: it removes the co-routed interaction term while preserving both the direct/self-curvature part and the same routing-frequency factor (definitions in Section C.3). Table 5 shows that across all cells, Full produces greater removal damage than Self$\times\rho$ in 20 cells, with two ties at the reported precision; at Top 9%, Full is more damaging in 9 of 12 cells. Direct and self-curvature effects therefore capture much of individual importance, while the interaction term provides a coalition-dependent correction whose causal impact becomes more visible at the larger removal budget. The sole exception is Granite, the smallest and most densely routed backbone: its experts are too weakly specialized for co-routed pairs to carry genuine synergy, and the interaction term accordingly contributes mostly estimation noise.

Table 5: Interaction ablation across Top 1%, 3%, and 9% removal budgets. Absolute post-removal PPL/ACC under output-zero. Full includes the co-routed interaction term; Self$\times\rho$ removes it while retaining the same direct/self-curvature part and routing-frequency factor.

| | OLMoE | | | Qwen3 | | | gpt-oss | | | Granite | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | WT-2 | GSM8K | MMLU | WT-2 | GSM8K | MMLU | WT-2 | GSM8K | MMLU | WT-2 | GSM8K | MMLU |
| ***Top 1%*** ***(stronger damage: PPL↑, ACC↓)*** | | | | | | | | | | | | |
| **Full** | 19.76 | 0.140 | 0.405 | 8.54 | 0.793 | 0.736 | 35.51 | 0.878 | 0.686 | 8.79 | 0.024 | 0.403 |
| Self$\times\rho$ | 19.58 | 0.138 | 0.442 | 8.58 | 0.812 | 0.744 | 31.22 | 0.877 | 0.686 | 8.50 | 0.018 | 0.370 |
| ***Top 3%*** ***(stronger damage: PPL↑, ACC↓)*** | | | | | | | | | | | | |
| **Full** | 21.85 | 0.036 | 0.384 | 10.02 | 0.060 | 0.675 | 28.64 | 0.743 | 0.659 | 14.18 | 0.011 | 0.282 |
| Self$\times\rho$ | 22.10 | 0.036 | 0.401 | 10.03 | 0.053 | 0.690 | 29.42 | 0.794 | 0.654 | 14.43 | 0.017 | 0.286 |
| ***Top 9%*** ***(stronger damage: PPL↑, ACC↓)*** | | | | | | | | | | | | |
| **Full** | 54.97 | 0.015 | 0.291 | 17.42 | 0.012 | 0.580 | 34.92 | 0.637 | 0.597 | 368.64 | 0.009 | 0.252 |
| Self$\times\rho$ | 49.62 | 0.021 | 0.316 | 16.95 | 0.020 | 0.605 | 31.41 | 0.599 | 0.582 | 377.77 | 0.013 | 0.253 |

Table 6: Sparse expert fine-tuning from the unmodified instruction-tuned checkpoint. Absolute post-training performance under matched Top 3% and 10% trainable-expert budgets.

| | OLMoE | | | Qwen3 | | | gpt-oss | | | Granite | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Selection method | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC |
| ***Unmodified checkpoint*** | | | | | | | | | | | | |
| Unmodified checkpoint | 13.73 | 0.525 | 0.487 | 7.23 | 0.933 | 0.760 | 25.57 | 0.894 | 0.698 | 7.41 | 0.602 | 0.505 |
| ***Top 3% trainable experts*** ***(better performance: PPL↓, ACC↑)*** | | | | | | | | | | | | |
| Routing frequency | 12.50 Δ -8.9% | 0.584 Δ +11.2% | 0.523 Δ +7.3% | 6.36 Δ -12.0% | 0.956 Δ +2.4% | 0.810 Δ +6.6% | 23.51 Δ -8.1% | 0.916 Δ +2.4% | 0.729 Δ +4.5% | 6.68 Δ -9.9% | 0.660 Δ +9.6% | 0.546 Δ +8.1% |
| **CONTRAIL** (Ours) | 11.99 Δ -12.7% | 0.608 Δ +15.9% | 0.538 Δ +10.4% | 6.09 Δ -15.9% | 0.963 Δ +3.2% | 0.825 Δ +8.7% | 22.59 Δ -11.6% | 0.925 Δ +3.5% | 0.743 Δ +6.4% | 6.36 Δ -14.1% | 0.684 Δ +13.7% | 0.563 Δ +11.6% |
| ***Top 10% trainable experts*** ***(better performance: PPL↓, ACC↑)*** | | | | | | | | | | | | |
| Routing frequency | 11.83 Δ -13.8% | 0.616 Δ +17.3% | 0.542 Δ +11.3% | 6.08 Δ -16.0% | 0.964 Δ +3.3% | 0.825 Δ +8.7% | 22.27 Δ -12.9% | 0.929 Δ +3.9% | 0.748 Δ +7.1% | 6.28 Δ -15.2% | 0.691 Δ +14.7% | 0.568 Δ +12.4% |
| **CONTRAIL** (Ours) | 11.54 Δ -15.9% | 0.630 Δ +20.0% | 0.551 Δ +13.1% | 5.96 Δ -17.6% | 0.967 Δ +3.6% | 0.832 Δ +9.5% | 21.72 Δ -15.1% | 0.935 Δ +4.6% | 0.756 Δ +8.3% | 6.11 Δ -17.5% | 0.704 Δ +16.9% | 0.577 Δ +14.3% |

### 4.6 RQ5: ARE HIGH-CONTRIBUTION EXPERTS THE BEST SPARSE ADAPTATION TARGETS?

We reuse the task-conditioned rankings from RQ2 and fine-tune only the top 3%/10% experts in each layer. Routers, attention blocks, embeddings, layer norms, and all unselected expert feed-forward networks (FFNs) remain frozen. The main comparison uses native routing frequency and **CONTRAIL**; the remaining detailed selectors are reported in Table 14. Table 6 shows that, under every matched trainable-expert budget, **CONTRAIL** selects the subset that yields the strongest improvement from the same pretrained starting point. The advantage is largest when adaptation is most constrained and narrows as more experts become trainable, closing the causal loop with RQ1: the experts that are most costly to remove are also the most effective parameters to update. **CONTRAIL** therefore turns contribution attribution into a practical selector for parameter-efficient adaptation.

## 5 CONCLUSION

We introduce **CONTRAIL**, a counterfactual attribution framework for measuring functional contribution in sparse Mixture-of-Experts models. Across four MoE architectures, **CONTRAIL** identifies the experts whose removal most affects model behavior and whose targeted fine-tuning yields the largest gains under the same sparse training budget. The resulting attributions reveal structure that routing usage alone misses: expert relevance is strongly task-dependent, and co-routed interactions refine importance beyond individual terms. Extending attribution from expert computation to discrete routing decisions, CCGE scores each choice against a mass-conserving one-swap counterfactual at the nearest switching boundary. Together, these results support viewing routing traces as the computational context for attribution rather than as importance estimates themselves.

### AI USE STATEMENT

Generative AI tools were used to assist with language polishing of the manuscript. All AI-assisted edits were reviewed by the authors, who take responsibility for the final content of this work.

## REFERENCES

Stefan Arnold, Marian Fietta, and Dilara Yesilbas. Routing in sparsely-gated language models responds to context. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 15–22. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.blackboxnlp-1.2.

Mikel Artetxe, Shruti Bhosale, Naman Goyal, Todor Mihaylov, Myle Ott, Sam Shleifer, Xi Victoria Lin, Jingfei Du, Srinivasan Iyer, Ramakanth Pasunuru, Giridharan Anantharaman, Xian Li, Shuohui Chen, Halil Akin, Mandeep Baines, Louis Martin, Xing Zhou, Punit Singh Koura, Brian O'Horo, Jeffrey Wang, Luke Zettlemoyer, Mona Diab, Zornitsa Kozareva, and Veselin Stoyanov. Efficient large scale language modeling with mixtures of experts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 11699–11732. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.804.

Jun Bai, Minghao Tong, Yang Liu, Zixia Jia, and Zilong Zheng. Understanding and leveraging the expert specialization of context faithfulness in mixture-of-experts LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21927–21942. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1114.

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

Anzhe Cheng, Shukai Duan, Shixuan Li, Chenzhong Yin, Mingxi Cheng, Heng Ping, Tamoghna Chattopadhyay, Sophia I. Thomopoulos, Shahin Nazarian, Paul Thompson, and Paul Bogdan. ERMoE: Eigen-reparameterized mixture-of-experts for stable routing and interpretable specialization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12997–13006, 2026.

Zewen Chi, Li Dong, Shaohan Huang, Damai Dai, Shuming Ma, Barun Patra, Saksham Singhal, Payal Bajaj, Xia Song, Xian-Ling Mao, Heyan Huang, and Furu Wei. On the representation collapse of sparse mixture of experts. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8493–8502. Association for Computational Linguistics, 2022a. doi: 10.18653/v1/2022.acl-long.581.

Damai Dai, Li Dong, Shuming Ma, Bo Zheng, Zhifang Sui, Baobao Chang, and Furu Wei. StableMoE: Stable routing strategy for mixture of experts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7085–7095. Association for Computational Linguistics, 2022b. doi: 10.18653/v1/2022.acl-long.489.

Damai Dai, Chengqi Deng, Chenggang Zhao, R.X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y.K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1280–1297. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.70.

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten P. Bosma, Zongwei Zhou, Tao Wang, Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, Kathleen Meier-Hellstern, Toju Duke, Lucas Dixon, Kun Zhang, Quoc V. Le, Yonghui Wu, Zhifeng Chen, and Claire Cui. GLaM: Efficient scaling of language models with mixture-of-experts. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5547–5569. PMLR, 2022.

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

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2242–2251. PMLR, 2019.

Amirata Ghorbani and James Y. Zou. Neuron shapley: Discovering the responsible neurons. In *Advances in Neural Information Processing Systems*, volume 33, pp. 5922–5932, 2020.

Granite Team, IBM. Granite-3.1-3b-a800m-instruct model card. Hugging Face Model Card, 2024. URL https://huggingface.co/ibm-granite/granite-3.1-3b-a800m-instruct.

Babak Hassibi and David G. Stork. Second order derivatives for network pruning: Optimal brain surgeon. In *Advances in Neural Information Processing Systems*, volume 5, 1992.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

Weizhong Huang, Yuxin Zhang, Xiawu Zheng, Fei Chao, Rongrong Ji, and Liujuan Cao. Discovering important experts for mixture-of-experts models pruning through a theoretical perspective. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-4538.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. doi: 10.1162/neco.1991.3.1.79.

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

Joseph D. Janizek, Pascal Sturmfels, and Su-In Lee. Explaining explanations: Axiomatic feature interactions for deep networks. *Journal of Machine Learning Research*, 22(104):1–54, 2021.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021. doi: 10.3390/app11146421.

Michael I. Jordan and Robert A. Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural Computation*, 6(2):181–214, 1994. doi: 10.1162/neco.1994.6.2.181.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1885–1894. PMLR, 2017.

Yann LeCun, John S. Denker, and Sara A. Solla. Optimal brain damage. In *Advances in Neural Information Processing Systems*, 1989.

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

Mike Lewis, Shruti Bhosale, Tim Dettmers, Naman Goyal, and Luke Zettlemoyer. BASE layers: Simplifying training of large, sparse models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6265–6274. PMLR, 2021.

Junzhuo Li, Bo Wang, Xiuze Zhou, Peijie Jiang, Jia Liu, and Xuming Hu. Decoding knowledge attribution in mixture-of-experts: A framework of basic-refinement collaboration and efficiency analysis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 22431–22446. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1093.

Meng Li, Hengyang Sun, Yanjun Huang, and Hong Chen. Shapley value: From cooperative game to explainable artificial intelligence. *Autonomous Intelligent Systems*, 4(1):2, 2024. doi: 10.1007/s43684-023-00060-8.

Ka Man Lo, Zeyu Huang, Zihan Qiu, Zili Wang, and Jie Fu. A closer look into mixture-of-experts in large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 4427–4447. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-naacl.251.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 2017.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations*, 2017.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, Yuling Gu, Shane Arora, Akshita Bhagia, Dustin Schwenk, David Wadden, Alexander Wettig, Binyuan Hui, Tim Dettmers, Douwe Kiela, Ali Farhadi, Noah A. Smith, Pang Wei Koh, Amanpreet Singh, and Hannaneh Hajishirzi. OLMoE: Open mixture-of-experts language models. In *International Conference on Learning Representations*, 2025.

Matthew Lyle Olson, Neale Ratzlaff, Musashi Hinck, Man Luo, Sungduk Yu, Chendi Xue, and Vasudev Lal. Probing semantic routing in large mixture-of-expert models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 18263–18278. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.991.

OpenAI. gpt-oss-120b & gpt-oss-20b model card. OpenAI Model Card, 2025. URL `https://openai.com/index/gpt-oss-model-card/`.

Lloyd S. Shapley. A value for n-person games. In Harold W. Kuhn and Albert W. Tucker (eds.), *Contributions to the Theory of Games II*, number 28 in Annals of Mathematics Studies, pp. 307–317. Princeton University Press, Princeton, NJ, 1953.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Zunhai Su, Qingyuan Li, Hao Zhang, YuLei Qian, Yuchen Xie, and Kehong Yuan. Unveiling super experts in mixture-of-experts large language models. *arXiv preprint arXiv:2507.23279*, 2025. URL `https://arxiv.org/abs/2507.23279`.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3319–3328. PMLR, 2017.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The shapley taylor interaction index. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9259–9268. PMLR, 2020.

Tho Tran Huu, Huu-Tuan Nguyen, Thien-Hai Nguyen, Nhat-Tri Ho, Viet-Hoang Tran, Tho Quan, and Tan Minh Nguyen. Geometric and stochastic analysis of discontinuities in sparse mixture-of-experts. *arXiv preprint arXiv:2606.19036*, 2026. URL https://arxiv.org/abs/2606.19036.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992. doi: 10.1007/BF00992696.

Yanyue Xie, Zhi Zhang, Ding Zhou, Cong Xie, Ziang Song, Xin Liu, Yanzhi Wang, Xue Lin, and An Xu. MoE-Pruner: Pruning mixture-of-experts large language model using the hints from its router. *arXiv preprint arXiv:2410.12013*, 2024. doi: 10.48550/arXiv.2410.12013. URL https://arxiv.org/abs/2410.12013.

Chu Xu, Xinke Jiang, Rihong Qiu, Jiaran Gao, and Junfeng Zhao. MODEL SHAPLEY: Find your ideal parameter player via one gradient backpropagation. In *Advances in Neural Information Processing Systems*, volume 38, pp. 79240–79290, 2025. doi: 10.52202/085713-2393.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Yuhao Zhang. SHAPE: Coalition-aware expert pruning for sparse mixture-of-experts LLMs. *arXiv preprint arXiv:2606.09886*, 2026. URL https://arxiv.org/abs/2606.09886.

Chen Zheng, Yuhang Cai, Deyi Liu, Jin Ma, Yiyuan Ma, Yuan Yang, Jing Liu, Yutao Zeng, Xun Zhou, and Siyuan Qiao. GatePro: Parameter-free expert selection optimization for mixture-of-experts models. *arXiv preprint arXiv:2510.13079*, 2025. doi: 10.48550/arXiv.2510.13079. URL https://arxiv.org/abs/2510.13079.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Y. Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V. Le, and James Laudon. Mixture-of-experts with expert choice routing. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Yixiao Zhou, Ziyu Zhao, Dongzhou Cheng, Zhiliang Wu, Jie Gui, Yi Yang, Fei Wu, Yu Cheng, and Hehe Fan. Dropping experts, recombining neurons: Retraining-free pruning for sparse mixture-of-experts LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 15169–15186. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.820.

# Appendix Contents


**Appendix A Mathematical Notation** . . . 16

**Appendix B Theoretical Derivations** . . . 16

*B.1* Fixed-route game, loss slice, and exact path identity . . . 16

*B.2* Exact and quadratic conditional expert marginals . . . 18

*B.3* Closed-form expert Shapley theorem . . . 19

*B.4* Pairwise interaction as a joint-removal residual . . . 20

*B.5* Nearest top-$k$ switching boundary and CCGE construction . . . 20

*B.6* Exact path and quadratic router Shapley value . . . 22

*B.7* All-active fixed-gate reduction . . . 23

*B.8* Shapley axioms for the quadratic expert game . . . 23

*B.9* Approximation and curvature-surrogate error . . . 23

*B.10* Theory-to-intervention alignment . . . 25

**Appendix C Estimator Details and Complexity** . . . 26

*C.1* Matrix-free output-space curvature products . . . 26

*C.2* Sparse support of the required products . . . 28

*C.3* Streaming aggregation . . . 28

*C.4* Computational complexity . . . 28

**Appendix D Detailed Experimental Protocols** . . . 29

*D.1* Backbones and model-specific routing structure . . . 29

*D.2* Data separation, attribution loss, and scoring . . . 29

*D.3* RQ1: usage–contribution separation and percentage removal . . . 30

*D.4* RQ2: task-specific cross-task intervention . . . 31

*D.5* RQ3: router estimators and boundary-exchange protocol . . . 31

*D.6* RQ4: co-routed interaction ablation . . . 33

*D.7* RQ5: sparse expert fine-tuning . . . 33

*D.8* Supporting probe-size stability . . . 33

*D.9* Matched GSM8K case-study selection . . . 34

**Appendix E Supporting Results in RQ Order** . . . 34

*E.1* RQ1: percentage-removal robustness and baseline definitions . . . 34

*E.2* Matched GSM8K qualitative cases . . . 36

*E.3* RQ2: task-specific cross-task matrices and heatmaps . . . 38
*E.4* RQ3: router attribution analysis . . . 42
*E.5* RQ5: gains over unmodified checkpoint . . . 42
*E.6* Supporting usage–value and stability diagnostics . . . 45
**Appendix F Use of Existing Assets** . . . 46
**Appendix G Related Work** . . . 46

# A Mathematical Notation

Table 7 summarizes the symbols used by the generic Shapley construction, the fixed-route expert game, the boundary router game, and the practical estimator. Token- and layer-specific arguments are suppressed when they are fixed by context. An *expert slot* means a layer-local expert, i.e., the pair consisting of a layer and its local expert index. Whole lowercase or Greek vectors are bold italic, matrices are uppercase bold italic, and their explicitly expanded scalar coordinates are ordinary italic. Scalar variables and fixed scalar constants are likewise ordinary italic. Global index sets such as $\mathcal{E}$ and $\mathcal{S}$ are calligraphic; locally introduced coalitions and auxiliary sets may use ordinary uppercase letters. Named operators and semantic labels are upright.

# B Theoretical Derivations

The main text states the two counterfactual games compactly. Here we separate three logically distinct objects that are easy to conflate: (i) the exact nonlinear characteristic function induced by a functional intervention, (ii) the quadratic Taylor game used to obtain a closed-form allocation, and (iii) the practical curvature surrogate used to estimate that quadratic game. The first specifies *what is attributed*; the second supplies a tractable local approximation; the third affects numerical estimation but not the game semantics.

**Color guide.** To keep the two derivations visually distinct, orange marks fixed-route expert displacements such as $\boldsymbol{\delta}_e$, while blue marks the matched router-exchange displacement $\boldsymbol{\delta}_r$. Indexed variants inherit the same color. The colors are expository only: they do not alter any definition and may be ignored in grayscale.

## B.1 Fixed-route game, downstream loss slice, and exact path identity

Fix one probe example $x$, one token representation $\boldsymbol{x}$, and one MoE layer. The example-level scalar loss is $\mathcal{L}_x$ from Equation (78). All non-intervened upstream inputs and model parameters are held fixed, while the downstream computation graph is recomputed normally after substituting the attributed MoE output. This defines the intervention slice $\mathcal{L}_{\boldsymbol{x}} : \mathbb{R}^{d_y} \to \mathbb{R}$ of $\mathcal{L}_x$ even when the supervised loss contains multiple gold target positions. Because the downstream graph may itself contain hard top-$k$ routers, $\mathcal{L}_{\boldsymbol{x}}$ is generally piecewise smooth as a function of the substituted output. The exact characteristic function below does not require smoothness. Whenever we invoke a gradient, Hessian, or path-Taylor identity, we condition on the relevant output segment remaining inside a single downstream routing region and assume that the remaining downstream computation is $C^2$ on that segment. If a substitution crosses a downstream routing boundary, the segment must instead be partitioned into smooth pieces. The full loss change is then the sum of the smooth-segment integrals and any jump terms at discontinuous switches; the jump terms vanish when the downstream map is continuous at the boundaries. The single-region identity below must not be applied across such a switch without this piecewise accounting. Let $\mathcal{S} = \mathcal{S}_k(\boldsymbol{x})$ be the active support recorded in the hard forward pass. The factual expert contributions and coalition outputs are

$$\begin{aligned}
\boldsymbol{\delta}_e &:= g_e \boldsymbol{f}_e, \\
\boldsymbol{y}^\star &= \sum_{e \in \mathcal{S}} \boldsymbol{\delta}_e, \\
\boldsymbol{y}_C &:= \sum_{e \in C} \boldsymbol{\delta}_e = \boldsymbol{y}^\star - \sum_{m \in \mathcal{S} \setminus C} \boldsymbol{\delta}_m, \qquad C \subseteq \mathcal{S}, \\
\boldsymbol{d}_C &:= \boldsymbol{y}_C - \boldsymbol{y}^\star = - \sum_{m \in \mathcal{S} \setminus C} \boldsymbol{\delta}_m.
\end{aligned} \tag{12}$$

The exact expert characteristic function is therefore

$$\mathcal{U}_{\boldsymbol{x}}^{\exp}(C) := -\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}_C). \tag{13}$$

This definition contains no Taylor approximation and exactly matches the fixed-route output-zero intervention.

We next record an exact calculus identity used by both games. Assume $\mathcal{L}_{\boldsymbol{x}}$ is twice continuously differentiable on the segment from $\boldsymbol{y}^\star$ to $\boldsymbol{y}^\star + \boldsymbol{d}$. Define $\boldsymbol{u}_\star := \nabla_{\boldsymbol{y}} \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star)$ and $\boldsymbol{H}(\boldsymbol{y}) := \nabla^2_{\boldsymbol{y}} \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y})$.

Table 7: Mathematical notation used by **CONTRAIL**.

| Symbol | Meaning |
|---|---|
| $N = \{1, \dots, M\}$ | Generic player set for the Shapley definition. |
| $C \subseteq N$ | Generic coalition; $\pi$ is a uniformly random player permutation. |
| $P_i^\pi$ | Players preceding $i$ in permutation $\pi$. |
| $x,\ \mathcal{T}_x$ | Probe example and the set of its supervised gold-target positions. |
| $\boldsymbol{\theta},\ x_{<t}$ | Fixed model parameters and token prefix before target position $t$. |
| $m_t,\ c_t$ | Binary supervision mask and gold token at supervised position $t$. |
| $\mathcal{L}_x$ | Factual scalar summed teacher-forced causal cross-entropy for probe example $x$. |
| $\mathcal{L}_{x,i}(\boldsymbol{y}_i),\ \mathcal{L}_{x,i,t}(\boldsymbol{y}_i)$ | Example loss slice and target-position term after substituting attributed output row $i$ by $\boldsymbol{y}_i$. |
| $\boldsymbol{x} \in \mathbb{R}^{d_x}$ | Token representation entering the attributed MoE layer; $d_x$ is its input width. |
| $\mathcal{E} = \{1, \dots, E\}$ | Expert index set. |
| $\boldsymbol{\ell}(\boldsymbol{x}) \in \mathbb{R}^E$ | Router logit vector; $\boldsymbol{e}_j \in \mathbb{R}^E$ is the $j$-th standard basis vector. |
| $\mathcal{S} = \mathcal{S}_k(\boldsymbol{x})$ | Realized hard top-$k$ expert support. |
| $g_e(\boldsymbol{x}) \geq 0$ | Recorded post-routing gate weight; $g_e = 0$ outside $\mathcal{S}$. |
| $\boldsymbol{g},\ \widetilde{\boldsymbol{g}} \in \mathbb{R}^E$ | Gate-weight vector and the local CCGE surrogate gate map. |
| $\boldsymbol{f}_e(\boldsymbol{x}) \in \mathbb{R}^{d_y}$ | Expert output at the factual token input. |
| $\boldsymbol{\delta}_e := g_e \boldsymbol{f}_e$ | Factual gated output contributed by active expert $e$. |
| $\boldsymbol{y}^\star = \sum_{e \in \mathcal{S}} \boldsymbol{\delta}_e$ | Factual MoE output. |
| $\boldsymbol{y}_C = \sum_{e \in C} \boldsymbol{\delta}_e$ | Fixed-route coalition output for $C \subseteq \mathcal{S}$. |
| $\boldsymbol{d}_C := \boldsymbol{y}_C - \boldsymbol{y}^\star$ | Coalition displacement from the factual output. |
| $\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y})$ | Intervention slice of $\mathcal{L}_x$ as a function of the MoE output $\boldsymbol{y}$. |
| $\mathcal{U}_{\boldsymbol{x}}^{\exp}(C)$ | Exact expert characteristic function, $-\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}_C)$. |
| $\mathcal{U}_{\boldsymbol{x},2}^{\exp}(C)$ | Quadratic expert characteristic function, $-q_\star(\boldsymbol{d}_C)$. |
| $\boldsymbol{u}_\star = \nabla_{\boldsymbol{y}} \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star)$ | Output gradient at the factual point. |
| $\boldsymbol{H}_\star = \nabla_{\boldsymbol{y}}^2 \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star)$ | Exact symmetric output Hessian at the factual point. |
| $\overline{\boldsymbol{H}}(\boldsymbol{d})$ | Exact path-averaged Hessian for displacement $\boldsymbol{d}$. |
| $\widehat{\boldsymbol{H}}$ | Practical curvature operator: an empirical-Fisher or GGN surrogate, or the exact Hessian accessed through HVPs. |
| $q_\star(\boldsymbol{d})$ | Second-order Taylor model around $\boldsymbol{y}^\star$. |
| $\boldsymbol{\Delta}_{\mathcal{S}} = \sum_{e \in \mathcal{S}} \boldsymbol{\delta}_e$ | Total factual routed expert output; equals $\boldsymbol{y}^\star$ in the MoE block. |
| $\boldsymbol{\Delta}_{\mathcal{S},e}^{\text{keep}}$ | Sum of the focal displacement and partner displacements retained in the sparse score. |
| $\phi_e^{\text{nl}}, \phi_e^{(2)}$ | Exact nonlinear and quadratic-game token-level expert Shapley values. |
| $\phi_e^{(2),\text{full}}, \phi_e^{(2),\text{sparse}}$ | Full quadratic expert score and its pair-truncated counterpart. |
| $\phi_{e,\text{self}}^{(2)}$ | Direct plus self-curvature part of $\phi_e^{(2)}$, excluding co-routed pairs. |
| $D_2(T)$ | Quadratic loss increase from jointly removing expert set $T$. |
| $\psi_{em}^{(2)}$ | Pairwise joint-removal residual, $\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_m$. |
| $\Phi_e^{\text{corpus}}, \Phi_e^{\text{cond}}$ | Corpus-average and active-token-conditional expert values. |
| $\rho_e = n_e / N_{\text{tok}}$ | Routing frequency of expert $e$ on the probe set. |
| $a, b$ | Lowest-logit selected and highest-logit excluded experts at the top-$k$ boundary. |
| $\boldsymbol{v} = \boldsymbol{e}_a - \boldsymbol{e}_b$ | Selected–excluded logit exchange direction. |
| $\mathbf{1} \in \mathbb{R}^E$ | All-ones vector used to state gate-mass conservation. |
| $\gamma = \ell_a - \ell_b > 0$ | Strict factual top-$k$ boundary margin. |
| $\boldsymbol{\ell}^0 = \boldsymbol{\ell} - \frac{\gamma}{2}\boldsymbol{v}$ | Orthogonal projection of the factual logits onto the nearest switching boundary. |
| $\boldsymbol{\ell}^- = \boldsymbol{\ell} - \gamma \boldsymbol{v}$ | Reflection of the factual logits across the nearest switching hyperplane. |
| $\mu = g_a > 0$ | Factual gate mass transferred from $a$ to $b$. |
| $\boldsymbol{\delta}_r = \mu(\boldsymbol{f}_a - \boldsymbol{f}_b)$ | Factual-minus-one-swap routing displacement. |
| $\boldsymbol{y}^{\text{common}}$ | Output of the unchanged selected experts in the one-swap construction. |
| $\boldsymbol{J}^{\text{CCGE}}$ | Mass-conserving local Jacobian $\frac{\mu}{2\gamma}\boldsymbol{v}\boldsymbol{v}^\top$. |
| $\phi_r^{\text{nl}}, \phi_r^{(2)}$ | Exact binary router value and its quadratic approximation. |
| $\widehat{\phi}$ | Practical score after replacing exact curvature by its estimator; $\widehat{\phi}_r^{(2)}$ is the router score. |
| $i,\ x(i)$ | Attributed unit (one token row, one MoE layer) and its probe example. |
| $\mathcal{B},\ n_{\mathcal{B}} = \lvert\mathcal{B}\rvert$ | Mini-batch of attributed units and its size. |
| $\boldsymbol{p}_i, \boldsymbol{w}_i,\ \widehat{K}_{\mathcal{B}}(\boldsymbol{p}, \boldsymbol{w})$ | Output-space displacements and their batch-averaged bilinear surrogate. |
| $\boldsymbol{F}_i,\ \boldsymbol{J}_i,\ \boldsymbol{\Delta\ell}_i$ | Expert-output matrix, gate Jacobian, and reflected-logit displacement used in the router dimension chain. |
| $O_e$ | Set of partner indices whose pair residuals are omitted by the sparse diagnostic score. |
| $\lambda,\ \nu,\ N_\lambda,\ Q_\lambda^{(\nu)}$ | MoE-layer index, router-estimator index, unit count, and mean layer score. |
| $L,\ P_{\text{obs}},\ C_{\exp}$ | Number of attributed MoE layers, number of observed layer-local expert pairs, and cost of one excluded-expert forward in the complexity analysis. |
| $\text{sim}_{\lambda e}(A, B)$ | Signed rank-normalized cell-level contribution similarity between tasks $A$ and $B$. |
| $\Omega_y,\ R_y$ | Output-space neighborhood and max displacement used in error bounds. |
| $M_3,\ \varepsilon_H$ | Third-derivative bound and Hessian-surrogate operator-norm error. |

For $h(t) := \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star + t\boldsymbol{d})$, the fundamental theorem of calculus gives $h'(t) = h'(0) + \int_0^t h''(s)\,ds$. Integrating this identity once more and exchanging the order on the triangle $0 \le s \le t \le 1$ yields

$$\begin{aligned}
\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star + \boldsymbol{d}) - \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star) &= \int_0^1 h'(t)\,dt \\
&= h'(0) + \int_0^1 \int_0^t h''(s)\,ds\,dt \\
&= h'(0) + \int_0^1 (1-s) h''(s)\,ds \\
&= \boldsymbol{u}_\star^\top \boldsymbol{d} + \int_0^1 (1-s)\boldsymbol{d}^\top \boldsymbol{H}(\boldsymbol{y}^\star + s\boldsymbol{d})\boldsymbol{d}\,ds \\
&= \boldsymbol{u}_\star^\top \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^\top \overline{\boldsymbol{H}}(\boldsymbol{d})\boldsymbol{d}.
\end{aligned} \tag{14}$$

where the *path-averaged Hessian*

$$\overline{\boldsymbol{H}}(\boldsymbol{d}) := 2\int_0^1 (1-s)\boldsymbol{H}(\boldsymbol{y}^\star + s\boldsymbol{d})\,ds, \tag{15}$$

is defined with the factor 2 so that $\overline{\boldsymbol{H}}(\boldsymbol{d}) = \boldsymbol{H}_\star$ whenever curvature is constant along the segment. This normalization is important: since $\int_0^1 (1-s)ds = 1/2$, omitting the factor 2 would introduce a spurious factor-of-two error in the quadratic term.

For clarity, the output-space objects have shapes $\boldsymbol{y}^\star, \boldsymbol{d}, \boldsymbol{u}_\star \in \mathbb{R}^{d_y}$ and $\boldsymbol{H}(\boldsymbol{y}), \overline{\boldsymbol{H}}(\boldsymbol{d}), \boldsymbol{H}_\star \in \mathbb{R}^{d_y \times d_y}$. Hence the two terms in the path identity are scalars, with the component-level expansion

$$\begin{aligned}
\boldsymbol{u}_\star^\top \boldsymbol{d} &= \sum_{p=1}^{d_y} u_{\star,p} d_p, \\
\boldsymbol{d}^\top \overline{\boldsymbol{H}}(\boldsymbol{d})\boldsymbol{d} &= \sum_{p=1}^{d_y}\sum_{q=1}^{d_y} d_p\, \overline{H}_{pq}(\boldsymbol{d})\, d_q \in \mathbb{R}.
\end{aligned} \tag{16}$$

The first contraction is coordinatewise; the second additionally exposes all cross-coordinate curvature couplings through the off-diagonal Hessian entries.

The local quadratic model used by **CONTRAIL** is obtained by freezing the path curvature at the factual point,

$$q_\star(\boldsymbol{d}) := \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star) + \boldsymbol{u}_\star^\top \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^\top \boldsymbol{H}_\star \boldsymbol{d}, \qquad \boldsymbol{H}_\star := \boldsymbol{H}(\boldsymbol{y}^\star). \tag{17}$$

It induces the finite quadratic expert game

$$\mathcal{U}_{\boldsymbol{x},2}^{\exp}(C) := -q_\star(\boldsymbol{d}_C), \qquad C \subseteq \mathcal{S}. \tag{18}$$

The additive constant $-\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star)$ is retained for semantic clarity but cancels from every marginal. All closed forms below are exact Shapley values for $\mathcal{U}_{\boldsymbol{x},2}^{\exp}$; their error relative to the nonlinear game is treated in Section B.9.

### B.2 Exact and quadratic conditional expert marginals

Fix $e \in \mathcal{S}$ and a predecessor coalition $C \subseteq \mathcal{S} \setminus \{e\}$. Let $R_C := \mathcal{S} \setminus (C \cup \{e\})$ be the experts that remain absent after $e$ is restored. Immediately after restoring $e$ the displacement is $\boldsymbol{d}_C^+ = -\sum_{m \in R_C} \boldsymbol{\delta}_m$; immediately before restoring it, $\boldsymbol{d}_C^- = \boldsymbol{d}_C^+ - \boldsymbol{\delta}_e$.

Under the single-region $C^2$ condition in Section B.1, the *exact nonlinear* marginal can be written without approximation. Along $\boldsymbol{\eta}_C(t) := \boldsymbol{y}^\star + \boldsymbol{d}_C^- + t\boldsymbol{\delta}_e$, $t \in [0,1]$,

$$\begin{aligned}
\Delta\mathcal{U}_{e,\mathrm{nl}}^{(C)} &:= \mathcal{U}_{\boldsymbol{x}}^{\exp}(C \cup \{e\}) - \mathcal{U}_{\boldsymbol{x}}^{\exp}(C) \\
&= \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star + \boldsymbol{d}_C^-) - \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star + \boldsymbol{d}_C^+) \\
&= -\int_0^1 \nabla_{\boldsymbol{y}} \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{\eta}_C(t))^\top \boldsymbol{\delta}_e\,dt.
\end{aligned} \tag{19}$$

This identity makes the counterfactual direction explicit: the marginal is the negative line integral of loss gradient while the factual expert contribution is restored. Applying Equation (14) separately to the two coalition endpoints gives the equivalent exact decomposition

$$\Delta\mathcal{U}^{(C)}_{e,\mathrm{nl}} = -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}(\boldsymbol{d}_C^-)^\top \overline{\boldsymbol{H}}(\boldsymbol{d}_C^-)\boldsymbol{d}_C^- - \frac{1}{2}(\boldsymbol{d}_C^+)^\top \overline{\boldsymbol{H}}(\boldsymbol{d}_C^+)\boldsymbol{d}_C^+. \tag{20}$$

This form is exact and shows precisely where coalition dependence enters the nonlinear game: the two endpoint displacements generally have different path-averaged Hessians. The quadratic approximation replaces both of them by the common factual Hessian $\boldsymbol{H}_\star$.

For the quadratic game, substitute $q_\star$ at the two endpoints:

$$\begin{aligned}
\Delta\mathcal{U}^{(C)}_{e,2} &:= q_\star(\boldsymbol{d}_C^-) - q_\star(\boldsymbol{d}_C^+) \\
&= -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}\Big[(\boldsymbol{d}_C^+ - \boldsymbol{\delta}_e)^\top \boldsymbol{H}_\star(\boldsymbol{d}_C^+ - \boldsymbol{\delta}_e) - (\boldsymbol{d}_C^+)^\top \boldsymbol{H}_\star \boldsymbol{d}_C^+\Big] \\
&= -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e - (\boldsymbol{d}_C^+)^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e \\
&= -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e + \sum_{m\in R_C} \boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_m.
\end{aligned} \tag{21}$$

The direct and self-curvature terms do not depend on coalition order. A pair term $(e, m)$ appears exactly when $m$ is still absent at the moment $e$ enters.

### B.3 Closed-form expert Shapley theorem

**Theorem 1** (Quadratic fixed-route expert Shapley value)**.** *Let $\pi$ be a uniformly random permutation of the active set $\mathcal{S}$ and let $P_e^\pi$ be the experts preceding $e$. The Shapley value of expert $e$ in the quadratic game $\mathcal{U}^{\exp}_{\boldsymbol{x},2}$ induced by $q_\star$ is*

$$\phi_e^{(2)} = -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e + \frac{1}{2}\sum_{m\in\mathcal{S}\setminus\{e\}} \boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_m. \tag{22}$$

*Equivalently, with $\boldsymbol{\Delta}_\mathcal{S} := \sum_{m\in\mathcal{S}} \boldsymbol{\delta}_m = \boldsymbol{y}^\star$,*

$$\phi_e^{(2)} = -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\Delta}_\mathcal{S}. \tag{23}$$

Writing output coordinates explicitly, the compact theorem is equivalently

$$\phi_e^{(2)} = -\sum_{p=1}^{d_y} u_{\star,p}\delta_{e,p} + \frac{1}{2}\sum_{p=1}^{d_y}\sum_{q=1}^{d_y} \delta_{e,p}\, H_{\star,pq} \left(\sum_{m\in\mathcal{S}} \delta_{m,q}\right). \tag{24}$$

This form makes the scalar output of the Shapley rule explicit: coordinate $p$ of the focal expert is coupled through $H_{\star,pq}$ to coordinate $q$ of the total routed output.

*Proof.* Define

$$A_e := -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e, \qquad \kappa_{em} := \boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_m. \tag{25}$$

By Equation (21),

$$\Delta\mathcal{U}^{(P_e^\pi)}_{e,2} = A_e + \sum_{m\neq e} \kappa_{em}\, \mathbf{1}\{m \text{ appears after } e \text{ in } \pi\}. \tag{26}$$

Here $\mathbf{1}\{\cdot\}$ denotes the indicator function. For each fixed $m \neq e$, exactly half of all permutations place $m$ after $e$; therefore

$$\begin{aligned}
\phi_e^{(2)} &= \mathbb{E}_\pi\left[\Delta\mathcal{U}^{(P_e^\pi)}_{e,2}\right] \\
&= A_e + \frac{1}{2}\sum_{m\neq e} \kappa_{em} \\
&= -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e + \frac{1}{2}\sum_{m\neq e} \boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_m.
\end{aligned} \tag{27}$$

Collecting the self term and pair sum yields $\frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \sum_{m\in\mathcal{S}} \boldsymbol{\delta}_m$, proving Equation (23). □

For $e \notin \mathcal{S}$, $g_e = 0$ and hence $\boldsymbol{\delta}_e = 0$. Extending the player set to all experts therefore makes every inactive expert a null player with zero value. The exact nonlinear expert Shapley value is $\phi_e^{\mathrm{nl}} = \mathbb{E}_\pi[\Delta\mathcal{U}_{e,\mathrm{nl}}^{(P_e^\pi)}]$. If $\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star + \boldsymbol{d}_C) = q_\star(\boldsymbol{d}_C)$ at every coalition output—in particular, if the loss is quadratic throughout a neighborhood containing those outputs—then the exact and quadratic games coincide and this value reduces to the displayed closed form. For a general nonlinear loss, that reduction is not guaranteed.

### B.4 Pairwise interaction as a joint-removal residual

Define the quadratic loss increase caused by removing a fixed-route set $T \subseteq \mathcal{S}$ from the factual output,

$$D_2(T) := q_\star\left(-\sum_{j\in T}\boldsymbol{\delta}_j\right) - q_\star(0). \tag{28}$$

For a pair $e \neq m$,

$$\begin{aligned} D_2(\{e,m\}) &= -\boldsymbol{u}_\star^\top(\boldsymbol{\delta}_e+\boldsymbol{\delta}_m) + \frac{1}{2}(\boldsymbol{\delta}_e+\boldsymbol{\delta}_m)^\top \boldsymbol{H}_\star(\boldsymbol{\delta}_e+\boldsymbol{\delta}_m),\\ D_2(\{e\}) + D_2(\{m\}) &= -\boldsymbol{u}_\star^\top(\boldsymbol{\delta}_e+\boldsymbol{\delta}_m) + \frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star\boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_m{}^\top \boldsymbol{H}_\star\boldsymbol{\delta}_m,\\ \psi_{em}^{(2)} &:= D_2(\{e,m\}) - D_2(\{e\}) - D_2(\{m\})\\ &= \boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star\boldsymbol{\delta}_m = \psi_{me}^{(2)}. \end{aligned} \tag{29}$$

Thus $\psi_{em}^{(2)} > 0$ means the pair is more damaging to remove jointly than predicted by adding its two separate removal effects, while $\psi_{em}^{(2)} < 0$ indicates sub-additivity. The expert Shapley value can be written equivalently as

$$\phi_e^{(2)} = D_2(\{e\}) + \frac{1}{2}\sum_{m\in\mathcal{S}\setminus\{e\}} \psi_{em}^{(2)}. \tag{30}$$

This removal-game identity fixes the sign convention: $\psi_{em}^{(2)}$ is a *joint-removal residual*, not the Harsanyi dividend of a separately normalized restoration game.

### B.5 Nearest top-$k$ switching boundary and CCGE construction

This subsection establishes the estimand used by router attribution: it identifies the nearest top-$k$ switching boundary, defines a matched finite one-swap endpoint, and derives the unique constant Jacobian within a specified pair-supported, mass-conserving linear class whose integral reaches that endpoint. It does not define the RQ3 baseline or layer-selection protocol; those comparisons treat the construction proved here as fixed and are specified separately in Section D.5.

**Nearest switching geometry.** Let $\boldsymbol{\ell} \in \mathbb{R}^E$ be the factual logit vector, assume $1 \le k < E$ and that the top-$k$ support is unique, and write $\mathcal{S} = \mathcal{S}_k(\boldsymbol{\ell})$. For selected $i \in \mathcal{S}$ and excluded $j \notin \mathcal{S}$, define $\boldsymbol{v}_{ij} := \boldsymbol{e}_i - \boldsymbol{e}_j$ and the switching hyperplane $\mathcal{H}_{ij} := \{\boldsymbol{z} : \boldsymbol{v}_{ij}^\top \boldsymbol{z} = 0\}$. Since $\boldsymbol{v}_{ij}^\top\boldsymbol{\ell} = \ell_i - \ell_j > 0$,

$$\begin{aligned} \operatorname{dist}(\boldsymbol{\ell},\mathcal{H}_{ij}) &= \frac{\ell_i-\ell_j}{\|\boldsymbol{e}_i-\boldsymbol{e}_j\|_2} = \frac{\ell_i-\ell_j}{\sqrt{2}},\\ a &:= \arg\min_{i\in\mathcal{S}} \ell_i, \qquad b := \arg\max_{j\notin\mathcal{S}} \ell_j,\\ \gamma &:= \ell_a - \ell_b > 0,\\ \min_{i\in\mathcal{S},\, j\notin\mathcal{S}} \operatorname{dist}(\boldsymbol{\ell},\mathcal{H}_{ij}) &= \operatorname{dist}(\boldsymbol{\ell},\mathcal{H}_{ab}) = \frac{\gamma}{\sqrt{2}}. \end{aligned} \tag{31}$$

Indeed, $\min_{i\in\mathcal{S},\, j\notin\mathcal{S}}(\ell_i-\ell_j) = \min_{i\in\mathcal{S}}\ell_i - \max_{j\notin\mathcal{S}}\ell_j = \ell_a - \ell_b = \gamma$, which proves the minimizing pair and the distance claim simultaneously.

Let $\boldsymbol{v} := \boldsymbol{e}_a - \boldsymbol{e}_b$. The orthogonal projection of $\boldsymbol{\ell}$ onto the nearest boundary is

$$\boldsymbol{\ell}^0 = \boldsymbol{\ell} - \frac{\boldsymbol{v}^\top\boldsymbol{\ell}}{\|\boldsymbol{v}\|_2^2}\boldsymbol{v} = \boldsymbol{\ell} - \frac{\gamma}{2}\boldsymbol{v}. \tag{32}$$

Reflecting the factual logits across the same hyperplane gives

$$\boldsymbol{\ell}^- := 2\boldsymbol{\ell}^0 - \boldsymbol{\ell} = \boldsymbol{\ell} - \gamma \boldsymbol{v}. \tag{33}$$

Consequently $\ell_a^- = \ell_b$ and $\ell_b^- = \ell_a$. Moreover, every unchanged selected logit satisfies $\ell_i = \ell_i^- \geq \ell_a = \ell_b^-$ for $i \in \mathcal{S} \setminus \{a\}$, whereas every unchanged excluded logit satisfies $\ell_j = \ell_j^- \leq \ell_b = \ell_a^-$ for $j \in \mathcal{E} \setminus (\mathcal{S} \cup \{b\})$. Hence, under the strict-margin and fixed tie-breaking assumptions, the original $k-1$ unchanged selected experts remain, $b$ replaces $a$, and no other membership changes. This is the chosen nearest-boundary pair exchange, mirrored at equal Euclidean distance on the two sides of the boundary. If several pairs attain the same minimum gap, the deterministic tie-breaking rule selects one of them.

**Matched functional endpoint.** The reflected logits identify which membership change is nearest, but they do not by themselves define the functional intervention. In particular, re-evaluating the native gate normalization at $\boldsymbol{\ell}^-$ can alter every gate weight and need not preserve the factual mass assigned to $a$. We instead hold the $k-1$ common expert contributions fixed and transfer exactly $\mu := g_a > 0$ from $a$ to $b$. Let

$$\begin{aligned} \boldsymbol{y}_{\text{common}} &:= \sum_{e \in \mathcal{S} \setminus \{a\}} g_e \boldsymbol{f}_e = \boldsymbol{y}^\star - \mu \boldsymbol{f}_a, \qquad \boldsymbol{\delta}_r := \mu(\boldsymbol{f}_a - \boldsymbol{f}_b), \\ \boldsymbol{y}^- &:= \boldsymbol{y}^\star - \boldsymbol{\delta}_r = \boldsymbol{y}_{\text{common}} + \mu \boldsymbol{f}_b. \end{aligned} \tag{34}$$

The two endpoints therefore differ only in whether the boundary mass $\mu$ multiplies $\boldsymbol{f}_a$ or $\boldsymbol{f}_b$. Geometry selects the pair $(a, b)$; mass transfer defines the estimand.

**Deriving the CCGE Jacobian from the endpoint.** We derive the local gate operator after imposing explicit design axioms. Treat $\mu$ and $\gamma$ as stop-gradient constants and restrict attention to a constant linear map $\boldsymbol{J} \in \mathbb{R}^{E \times E}$ from logit displacement to gate displacement. The intended one-swap intervention imposes three structural conditions: (i) pair support, so rows and columns outside $\{a, b\}$ vanish; (ii) mass conservation, $\mathbf{1}^\top \boldsymbol{J} = 0$; and (iii) invariance to a common shift of the pair logits, $\boldsymbol{J}\mathbf{1} = 0$. Consequently the only possibly nonzero block must have the form

$$\boldsymbol{J}_{\{a,b\},\{a,b\}} = \alpha \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \qquad \boldsymbol{J} = \alpha \boldsymbol{v}\boldsymbol{v}^\top, \tag{35}$$

for a single scalar $\alpha$. The reflected logit displacement is $\boldsymbol{\Delta\ell} := \boldsymbol{\ell}^- - \boldsymbol{\ell} = -\gamma \boldsymbol{v}$; on the pair coordinates it is $[-\gamma, \gamma]^\top$. Endpoint consistency requires this finite logit exchange to induce the gate displacement $[-\mu, \mu]^\top = -\mu \boldsymbol{v}$. Substituting Equation (35) gives

$$\boldsymbol{J}\boldsymbol{\Delta\ell} = -\alpha\gamma \boldsymbol{v}(\boldsymbol{v}^\top \boldsymbol{v}) = -2\alpha\gamma \boldsymbol{v} \overset{!}{=} -\mu \boldsymbol{v}, \qquad \Longrightarrow \qquad \alpha = \frac{\mu}{2\gamma}. \tag{36}$$

Hence, within this constant pair-supported linear class, the structural and endpoint conditions uniquely determine

$$\boldsymbol{J}^{\text{CCGE}} := \frac{\mu}{2\gamma} \boldsymbol{v}\boldsymbol{v}^\top = \frac{\mu}{2\gamma} (\boldsymbol{e}_a - \boldsymbol{e}_b)(\boldsymbol{e}_a - \boldsymbol{e}_b)^\top. \tag{37}$$

The factor $2\gamma$ has a direct finite-difference interpretation: the logit difference $\ell_a - \ell_b$ moves from $+\gamma$ to $-\gamma$, a total change of $2\gamma$, while gate mass $\mu$ is transferred. Thus $\mu/(2\gamma)$ is the endpoint-calibrated transfer rate, not a free straight-through scale.

The full $E \times E$ operator is represented completely by its nonzero block,

$$\boldsymbol{J}^{\text{CCGE}}_{\{a,b\},\{a,b\}} = \frac{\mu}{2\gamma} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \qquad [\boldsymbol{J}^{\text{CCGE}}]_{ij} = \frac{\mu}{2\gamma} \begin{cases} 1, & (i,j) \in \{(a,a),(b,b)\}, \\ -1, & (i,j) \in \{(a,b),(b,a)\}, \\ 0, & \text{otherwise.} \end{cases} \tag{38}$$

The endpoint calibration is visible directly in the nonzero block:

$$\frac{\mu}{2\gamma} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} -\gamma \\ \gamma \end{bmatrix} = \begin{bmatrix} -\mu \\ \mu \end{bmatrix}. \tag{39}$$

Its remaining structural properties follow immediately:

$$\begin{aligned} \mathbf{1}^\top \boldsymbol{J}^{\mathrm{CCGE}} &= 0, & \boldsymbol{J}^{\mathrm{CCGE}}\mathbf{1} &= 0, \\ \boldsymbol{z}^\top \boldsymbol{J}^{\mathrm{CCGE}} \boldsymbol{z} &= \frac{\mu}{2\gamma}(\boldsymbol{v}^\top \boldsymbol{z})^2 \geq 0, & \operatorname{rank}(\boldsymbol{J}^{\mathrm{CCGE}}) &= 1. \end{aligned} \tag{40}$$

The first identity conserves total gate mass, the second removes the uninformative common-logit direction, and the last two show that the only nonzero eigendirection is the selected–excluded contrast $\boldsymbol{v}$, with eigenvalue $\mu/\gamma$ because $\|\boldsymbol{v}\|_2^2 = 2$.

**An integrable local gate map.** The constant Jacobian above is realized by the affine gate map

$$\widetilde{\boldsymbol{g}}(\boldsymbol{z}) := \boldsymbol{g}^{\mathrm{common}} + \frac{\mu}{2}\left(1 + \frac{\boldsymbol{v}^\top \boldsymbol{z}}{\gamma}\right)\boldsymbol{e}_a + \frac{\mu}{2}\left(1 - \frac{\boldsymbol{v}^\top \boldsymbol{z}}{\gamma}\right)\boldsymbol{e}_b, \tag{41}$$

where $\boldsymbol{g}^{\mathrm{common}}$ contains the unchanged factual gates of $\mathcal{S}\setminus\{a\}$ and is zero at $a, b$. Because $\boldsymbol{v}^\top \boldsymbol{\ell} = \gamma$ and $\boldsymbol{v}^\top \boldsymbol{\ell}^- = -\gamma$, this map is anchored at both finite endpoints:

$$\widetilde{\boldsymbol{g}}(\boldsymbol{\ell}) = \boldsymbol{g}^{\mathrm{common}} + \mu \boldsymbol{e}_a, \quad \widetilde{\boldsymbol{g}}(\boldsymbol{\ell}^-) = \boldsymbol{g}^{\mathrm{common}} + \mu \boldsymbol{e}_b, \quad \frac{\partial \widetilde{\boldsymbol{g}}}{\partial \boldsymbol{z}} = \boldsymbol{J}^{\mathrm{CCGE}}. \tag{42}$$

Both pair weights remain in $[0, \mu]$ throughout the local slab $|\boldsymbol{v}^\top \boldsymbol{z}| \leq \gamma$. Outside this slab the affine extension may produce negative coordinates and is not claimed to be a valid forward router. The token input and expert outputs $\boldsymbol{f}_e$ are held fixed along this local path.

Finally, parameterize the reflected segment as $\boldsymbol{\ell}(t) = \boldsymbol{\ell} - t\gamma \boldsymbol{v}$, $t \in [0, 1]$. Since $\boldsymbol{v}^\top \boldsymbol{\ell}(t) = \gamma(1 - 2t)$, the pair weights are exactly $\widetilde{\boldsymbol{g}}_a(\boldsymbol{\ell}(t)) = \mu(1 - t)$ and $\widetilde{\boldsymbol{g}}_b(\boldsymbol{\ell}(t)) = \mu t$. The induced output is therefore

$$\boldsymbol{y}^{\mathrm{CCGE}}(t) = \boldsymbol{y}_{\mathrm{common}} + \mu(1 - t)\boldsymbol{f}_a + \mu t \boldsymbol{f}_b = \boldsymbol{y}^\star - t\boldsymbol{\delta}_r. \tag{43}$$

Equivalently, with $\boldsymbol{F} := [\boldsymbol{f}_1, \ldots, \boldsymbol{f}_E] \in \mathbb{R}^{d_y \times E}$, its constant velocity follows from the explicit chain

$$\begin{aligned} \frac{\mathrm{d}\boldsymbol{y}^{\mathrm{CCGE}}(t)}{\mathrm{d}t} &= \underbrace{\boldsymbol{F}}_{d_y \times E}\ \underbrace{\boldsymbol{J}^{\mathrm{CCGE}}}_{E \times E}\ \underbrace{(-\gamma \boldsymbol{v})}_{E \times 1} \\ &= -\mu \boldsymbol{F}\boldsymbol{v} = -\mu(\boldsymbol{f}_a - \boldsymbol{f}_b) = -\boldsymbol{\delta}_r \in \mathbb{R}^{d_y}. \end{aligned} \tag{44}$$

Integrating from $t = 0$ to $t = 1$ yields $\boldsymbol{y}^{\mathrm{CCGE}}(1) - \boldsymbol{y}^{\mathrm{CCGE}}(0) = -\boldsymbol{\delta}_r$, so the local differential operator reaches exactly the matched finite endpoint $\boldsymbol{y}^- = \boldsymbol{y}^\star - \boldsymbol{\delta}_r$; it introduces no additional surrogate-endpoint error. The path crosses the switching boundary at $t = \frac{1}{2}$.

If $\gamma = 0$, the factual logits already lie on a switching boundary and the normalization in Equation (37) is undefined. The route value remains a finite endpoint utility difference once a deterministic alternative is specified, but CCGE as written is a strict-margin local construction. Exact ties are therefore outside the differentiable CCGE case.

### B.6 Exact path and quadratic router Shapley value

The local router game contains one player $r$. Its empty state is the matched one-swap output $\boldsymbol{y}^- = \boldsymbol{y}^\star - \boldsymbol{\delta}_r$ and its factual state is $\boldsymbol{y}^\star$:

$$V_r(\varnothing) := -\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^-), \qquad V_r(\{r\}) := -\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star). \tag{45}$$

Hence the one-player Shapley value is the exact endpoint utility difference. Under the single-region $C^1$ condition from Section B.1, it also admits the path-integral representation

$$\begin{aligned} \phi_r^{\mathrm{nl}} &= V_r(\{r\}) - V_r(\varnothing) \\ &= \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^-) - \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star) \\ &= -\int_0^1 \nabla_{\boldsymbol{y}} \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star - t\boldsymbol{\delta}_r)^\top \boldsymbol{\delta}_r \, dt. \end{aligned} \tag{46}$$

If $\mathcal{L}_{\boldsymbol{x}}$ is $C^2$ on the same segment, applying Equation (14) to the displacement $\boldsymbol{d} = -\boldsymbol{\delta}_r$ gives the exact second-order path representation

$$\phi_r^{\mathrm{nl}} = -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_r + \int_0^1 (1 - t)\boldsymbol{\delta}_r{}^\top \boldsymbol{H}(\boldsymbol{y}^\star - t\boldsymbol{\delta}_r)\boldsymbol{\delta}_r \, dt. \tag{47}$$

Freezing curvature at $\boldsymbol{y}^\star$ yields the quadratic approximation

$$\begin{aligned}\phi_r^{(2)} &= -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_r + \frac{1}{2}\boldsymbol{\delta}_r{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_r \\ &= -\mu \boldsymbol{u}_\star^\top \boldsymbol{f}_a + \mu \boldsymbol{u}_\star^\top \boldsymbol{f}_b + \frac{\mu^2}{2}\boldsymbol{f}_a^\top \boldsymbol{H}_\star \boldsymbol{f}_a + \frac{\mu^2}{2}\boldsymbol{f}_b^\top \boldsymbol{H}_\star \boldsymbol{f}_b - \mu^2 \boldsymbol{f}_a^\top \boldsymbol{H}_\star \boldsymbol{f}_b.\end{aligned} \tag{48}$$

The factor $1/2$ is the integral $\int_0^1 (1-t)dt$ for constant curvature; it is not an additional heuristic weight. The selected and excluded experts enter with opposite first-order signs, and their cross-curvature is negative because the counterfactual exchanges one expert output for the other.

### B.7 All-active fixed-gate reduction

Suppose $E = k$, so every expert is active for every token and the gate weights are fixed constants. There is no selected–excluded membership decision and hence no router boundary game. With effective additive blocks $\bar{\boldsymbol{f}}_e := g_e \boldsymbol{f}_e$, the expert formula becomes

$$\phi_e^{(2)} = -\boldsymbol{u}_\star^\top \bar{\boldsymbol{f}}_e + \frac{1}{2}\bar{\boldsymbol{f}}_e^\top \boldsymbol{H}_\star \bar{\boldsymbol{f}}_e + \frac{1}{2}\sum_{m\neq e} \bar{\boldsymbol{f}}_e^\top \boldsymbol{H}_\star \bar{\boldsymbol{f}}_m = -\boldsymbol{u}_\star^\top \bar{\boldsymbol{f}}_e + \frac{1}{2}\bar{\boldsymbol{f}}_e^\top \boldsymbol{H}_\star \sum_m \bar{\boldsymbol{f}}_m. \tag{49}$$

Thus **CONTRAIL** reduces to the standard quadratic Shapley allocation of additive functional blocks when conditional routing disappears.

### B.8 Shapley axioms for the quadratic expert game

Consider the finite cooperative game $\mathcal{U}_{\boldsymbol{x},2}^{\exp}$ defined in Equation (18). Since Equation (22) was obtained by the exact Shapley expectation for this game, all four standard axioms hold exactly for the quadratic characteristic function.

**Null player.** If $\mathcal{U}_{\boldsymbol{x},2}^{\exp}(C \cup \{e\}) = \mathcal{U}_{\boldsymbol{x},2}^{\exp}(C)$ for every $C \subseteq \mathcal{S} \setminus \{e\}$, every marginal in **??** is zero and therefore $\phi_e^{(2)} = 0$. The structural condition $\boldsymbol{\delta}_e = 0$ is a sufficient special case.

**Symmetry.** If experts $e$ and $m$ are exchangeable in the characteristic function, i.e., $\mathcal{U}_{\boldsymbol{x},2}^{\exp}(C \cup \{e\}) = \mathcal{U}_{\boldsymbol{x},2}^{\exp}(C \cup \{m\})$ for every coalition excluding both, then their permutation marginals match under the label swap and $\phi_e^{(2)} = \phi_m^{(2)}$.

**Linearity.** For any two characteristic functions $\mathcal{U}_1, \mathcal{U}_2$ on the same player set and scalars $\alpha, \beta$, each coalition marginal of $\alpha\mathcal{U}_1 + \beta\mathcal{U}_2$ is the same linear combination of the two marginals; therefore the Shapley expectation is linear.

**Efficiency.** Let $\boldsymbol{\Delta}_{\mathcal{S}} := \sum_{e\in\mathcal{S}} \boldsymbol{\delta}_e$. Summing the closed form gives

$$\begin{aligned}\sum_{e\in\mathcal{S}} \phi_e^{(2)} &= -\boldsymbol{u}_\star^\top \boldsymbol{\Delta}_{\mathcal{S}} + \frac{1}{2}\sum_e \boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e + \frac{1}{2}\sum_e \sum_{m\neq e} \boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_m \\ &= -\boldsymbol{u}_\star^\top \boldsymbol{\Delta}_{\mathcal{S}} + \frac{1}{2}\boldsymbol{\Delta}_{\mathcal{S}}^\top \boldsymbol{H}_\star \boldsymbol{\Delta}_{\mathcal{S}} \\ &= q_\star(-\boldsymbol{\Delta}_{\mathcal{S}}) - q_\star(0) \\ &= \mathcal{U}_{\boldsymbol{x},2}^{\exp}(\mathcal{S}) - \mathcal{U}_{\boldsymbol{x},2}^{\exp}(\varnothing).\end{aligned} \tag{50}$$

This is the standard baseline-correct efficiency identity; no assumption that $\mathcal{U}_{\boldsymbol{x},2}^{\exp}(\varnothing) = 0$ is made.

### B.9 Approximation and curvature-surrogate error

The exact intervention games and the practical score differ for conceptually separate reasons. First, replacing the nonlinear loss by $q_\star$ incurs a Taylor remainder. Second, replacing the exact Hessian $\boldsymbol{H}_\star$ by a surrogate $\widehat{\boldsymbol{H}}$ incurs curvature error. The compact full **CONTRAIL** score does not require pair

truncation; a third term is relevant only for optional sparse pair-diagnostic variants that intentionally omit individual residuals.

Assume $\mathcal{L}_{\boldsymbol{x}} \in C^3$ on a convex neighborhood $\Omega_y$ containing all fixed-route coalition outputs and the router segment. We additionally require $\Omega_y$ to lie inside one downstream hard-routing region, with the remaining downstream computation $C^3$ throughout $\Omega_y$. Thus the relevant output paths avoid downstream routing switches while retaining the regularity needed for the third-order bound. This is the third-order counterpart of the local $C^2$ condition in Section B.1; the exact intervention game remains defined even when this local smoothness condition fails. Let

$$M_3 := \sup_{\boldsymbol{z}\in\Omega_y} \|\nabla_{\boldsymbol{y}}^3 \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{z})\|_{\mathrm{op}}, \qquad R_y := \max_{C\subseteq\mathcal{S}} \|\boldsymbol{d}_C\|_2. \tag{51}$$

For the third derivative, the operator norm means

$$\|\nabla_{\boldsymbol{y}}^3 \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{z})\|_{\mathrm{op}} := \sup_{\|\boldsymbol{\alpha}\|_2=\|\boldsymbol{\beta}\|_2=\|\boldsymbol{\zeta}\|_2=1} \left|\nabla_{\boldsymbol{y}}^3 \mathcal{L}_{\boldsymbol{x}}(\boldsymbol{z})[\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\zeta}]\right|, \tag{52}$$

where the bracket denotes evaluation of the trilinear third-derivative tensor. Taylor's theorem gives, for every expert coalition endpoint,

$$|\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{y}^\star + \boldsymbol{d}_C) - q_\star(\boldsymbol{d}_C)| \le \frac{M_3}{6}\|\boldsymbol{d}_C\|_2^3 \le \frac{M_3}{6} R_y^3. \tag{53}$$

An expert marginal subtracts two such endpoints, while its Shapley value is a convex average of marginals. Therefore

$$\left|\phi_e^{\mathrm{nl}} - \phi_e^{(2)}\right| \le \frac{M_3}{3} R_y^3. \tag{54}$$

For the router game only the nonzero endpoint $\boldsymbol{d} = -\boldsymbol{\delta}_r$ is approximated, so

$$\left|\phi_r^{\mathrm{nl}} - \phi_r^{(2)}\right| \le \frac{M_3}{6}\|\boldsymbol{\delta}_r\|_2^3. \tag{55}$$

These are local bounds: they become informative when the relevant functional output displacements stay within a region of controlled third derivative. No separate smallness assumption is imposed in parameter space; locality is instead controlled by the output displacement and the derivatives along the corresponding output-space region.

Now let $\widehat{\boldsymbol{H}}$ be a symmetric curvature surrogate satisfying

$$\|\boldsymbol{H}_\star - \widehat{\boldsymbol{H}}\|_{\mathrm{op}} \le \varepsilon_H. \tag{56}$$

For any displacement $\boldsymbol{d}$,

$$\left|\frac{1}{2}\boldsymbol{d}^\top(\boldsymbol{H}_\star - \widehat{\boldsymbol{H}})\boldsymbol{d}\right| \le \frac{1}{2}\varepsilon_H\|\boldsymbol{d}\|_2^2. \tag{57}$$

The closed form permits a sharper expert-specific bound than separately bounding two coalition endpoints. If $\widehat{\phi}_e^{(2)}$ denotes Equation (23) with $\boldsymbol{H}_\star$ replaced by $\widehat{\boldsymbol{H}}$, then

$$\left|\phi_e^{(2)} - \widehat{\phi}_e^{(2)}\right| = \frac{1}{2}\left|\boldsymbol{\delta}_e{}^\top(\boldsymbol{H}_\star - \widehat{\boldsymbol{H}})\boldsymbol{\Delta}_{\mathcal{S}}\right| \le \frac{1}{2}\varepsilon_H\|\boldsymbol{\delta}_e\|_2\|\boldsymbol{\Delta}_{\mathcal{S}}\|_2. \tag{58}$$

For the router endpoint, let $\widehat{\phi}_r^{(2)}$ denote Equation (48) with $\boldsymbol{H}_\star$ replaced by $\widehat{\boldsymbol{H}}$. The corresponding curvature-surrogate bound is $\frac{1}{2}\varepsilon_H\|\boldsymbol{\delta}_r\|_2^2$. These are conditional error decompositions: they do not assert that the practical rank-one surrogate satisfies a small $\varepsilon_H$ for a general downstream network. A weaker, task-specific guarantee would only need to control the finitely many bilinear directions appearing in the expert and router scores, rather than the full operator norm.

Finally, for an optional sparse pair-diagnostic variant, suppose the diagnostic for expert $e$ intentionally omits a set $O_e \subseteq \mathcal{S}\setminus\{e\}$ of pair residuals. This case is separate from the compact full score in Equation (23), which already sums all interactions without enumerating pairs. Write $\phi_e^{(2),\mathrm{full}} := \phi_e^{(2)}$. Then

$$\begin{aligned}
\phi_e^{(2),\mathrm{sparse}} &:= -\boldsymbol{u}_\star^\top\boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e{}^\top\boldsymbol{H}_\star\boldsymbol{\delta}_e + \frac{1}{2}\sum_{m\in\mathcal{S}\setminus(\{e\}\cup O_e)} \boldsymbol{\delta}_e{}^\top\boldsymbol{H}_\star\boldsymbol{\delta}_m, \\
\phi_e^{(2),\mathrm{full}} - \phi_e^{(2),\mathrm{sparse}} &= \frac{1}{2}\sum_{m\in O_e}\psi_{em}^{(2)}, \\
\left|\phi_e^{(2),\mathrm{full}} - \phi_e^{(2),\mathrm{sparse}}\right| &\le \frac{1}{2}\sum_{m\in O_e}|\psi_{em}^{(2)}|.
\end{aligned} \tag{59}$$

To combine pair omission with curvature substitution, define the displacement sum retained by the sparse score as

$$\boldsymbol{\Delta}_{\mathcal{S},e}^{\text{keep}} := \boldsymbol{\Delta}_{\mathcal{S}} - \sum_{m \in O_e} \boldsymbol{\delta}_m. \tag{60}$$

With this notation, $\phi_e^{(2),\text{sparse}} = -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2}\boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\Delta}_{\mathcal{S},e}^{\text{keep}}$. The exact-Hessian sparse score and its surrogate counterpart differ by $\frac{1}{2}\boldsymbol{\delta}_e{}^\top (\boldsymbol{H}_\star - \widehat{\boldsymbol{H}})\boldsymbol{\Delta}_{\mathcal{S},e}^{\text{keep}}$. Consequently, the corresponding curvature-substitution error is at most $\frac{1}{2}\varepsilon_H \|\boldsymbol{\delta}_e\|_2 \|\boldsymbol{\Delta}_{\mathcal{S},e}^{\text{keep}}\|_2$. The retained sum, rather than the full $\boldsymbol{\Delta}_{\mathcal{S}}$, is required when pair omission and curvature substitution are applied simultaneously.

Let $\widehat{\phi}_e^{\textsc{Contrail}}$ denote the quadratic score after replacing $\boldsymbol{H}_\star$ by the chosen surrogate and, if applicable, omitting the pairs in $O_e$. The three error sources separate through the exact algebraic identity

$$\begin{aligned}\phi_e^{\text{nl}} - \widehat{\phi}_e^{\textsc{Contrail}} = \big(\phi_e^{\text{nl}} - \phi_e^{(2),\text{full}}\big) + \big(\phi_e^{(2),\text{full}} - \phi_e^{(2),\text{sparse}}\big) \\ + \big(\phi_e^{(2),\text{sparse}} - \widehat{\phi}_e^{\textsc{Contrail}}\big).\end{aligned} \tag{61}$$

Applying the triangle inequality and the three bounds above yields the conservative expert-score bound

$$\left|\phi_e^{\text{nl}} - \widehat{\phi}_e^{\textsc{Contrail}}\right| \le \frac{M_3}{3} R_y^3 + \frac{1}{2}\varepsilon_H \|\boldsymbol{\delta}_e\|_2 \|\boldsymbol{\Delta}_{\mathcal{S},e}^{\text{keep}}\|_2 + \frac{1}{2}\sum_{m \in O_e} |\psi_{em}^{(2)}|. \tag{62}$$

For the router score, with no pair truncation,

$$\left|\phi_r^{\text{nl}} - \widehat{\phi}_r^{(2)}\right| \le \frac{M_3}{6}\|\boldsymbol{\delta}_r\|_2^3 + \frac{1}{2}\varepsilon_H \|\boldsymbol{\delta}_r\|_2^2. \tag{63}$$

The first term is model nonlinearity and the second is curvature estimation. The third term in the expert bound applies only to the optional sparse pair-diagnostic variant and is not incurred by the compact full **CONTRAIL** score. When $O_e = \varnothing$, $\boldsymbol{\Delta}_{\mathcal{S},e}^{\text{keep}} = \boldsymbol{\Delta}_{\mathcal{S}}$ and the result reduces to the full-score curvature bound in Equation (58). Keeping these terms separate prevents estimator error from being conflated with the definition of the counterfactual game.

### B.10 THEORY-TO-INTERVENTION ALIGNMENT

The intervention design uses endpoints from the same counterfactual families as the two games. Table 8 makes this correspondence explicit. For expert attribution, the empirical removal of a ranked set instantiates a coalition endpoint of the fixed-route game and evaluates the ranking; it does not directly enumerate every marginal entering the Shapley average. The router intervention directly evaluates the binary game's matched endpoint.

Table 8: Theory-to-intervention alignment. Empirical interventions use matched endpoints from the corresponding counterfactual game; expert removal evaluates ranked coalition endpoints rather than enumerating all Shapley marginals.

| Object | Theoretical counterfactual | Empirical intervention | Main evidence |
|---|---|---|---|
| Expert value | $C \mapsto \boldsymbol{y}_C = \sum_{e \in C} \boldsymbol{\delta}_e$ | ranked-coalition output-zero with route and gates fixed | RQ1 / Table 1 |
| Expert interaction | $D_2(\{e,m\}) - D_2(\{e\}) - D_2(\{m\})$ | Full-vs-Self ranking under matched coalition output-zero | RQ4 / Table 5 |
| Router value | $\boldsymbol{y}^\star \mapsto \boldsymbol{y}^\star - \boldsymbol{\delta}_r$ | nearest-boundary one-swap exchange | RQ3 / Table 4 |
| Estimator stability | fixed game, increasing probe support | probe-size ranking convergence | Appendix / Figure 6 |

For the expert-game evaluations that use output-zero removal (RQ1 Top 1%/3%, RQ2, and RQ4), the selected expert set and recorded gates are held fixed exactly as in Equation (1). The RQ1 Bottom 10%/20% logit-mask intervention is deliberately support-changing and is used as a separate expendability test rather than as a direct realization of the fixed-route characteristic function. For the router experiment, the selected–excluded pair and transferred gate mass are those used by Equation (5). This separation keeps each empirical test aligned with the counterfactual question it is intended to answer.

# C Estimator Details and Complexity

## C.1 Matrix-free output-space curvature products

For attributed unit $i$ (a token row at a scored MoE layer), let $\mathcal{S}_i$ denote its factual support, $\boldsymbol{\delta}_{i,e}$ the factual gated output of expert $e$, $\boldsymbol{y}_i^\star := \sum_{e\in\mathcal{S}_i} \boldsymbol{\delta}_{i,e}$ the factual MoE output, and $x(i)$ the containing probe example. We write $\mathcal{L}_{x(i),i}(\boldsymbol{y}_i)$ for the intervention slice obtained by varying row $\boldsymbol{y}_i$ while holding all other inputs fixed. For router attribution, let $\boldsymbol{\delta}_{i,r}$ denote the indexed nearest-boundary one-swap displacement defined by the construction in Section B.5. Define $\boldsymbol{u}_i := \nabla_{\boldsymbol{y}_i}\mathcal{L}_{x(i),i}(\boldsymbol{y}_i^\star)$ and let $\boldsymbol{p}_i, \boldsymbol{w}_i \in \mathbb{R}^{d_y}$ be arbitrary functional displacements; $\boldsymbol{w}_i$ avoids overloading $r$, which denotes the router player. The exact theory uses $\boldsymbol{H}_i = \nabla^2_{\boldsymbol{y}_i}\mathcal{L}_{x(i),i}(\boldsymbol{y}_i^\star)$, and we write $\widehat{\boldsymbol{H}}_i$ for the chosen per-unit curvature surrogate.

In the implementation, the mini-batch loss is the sum of the per-sequence losses from Equation (78). Because a sum reduction does not mix gradients across batch elements, a backward hook on the gated expert-output rows exposes each $\boldsymbol{u}_i$ directly during a single ordinary backward pass; no extra forward or backward pass is required for curvature. A practical positive-semidefinite gradient-outer-product surrogate is

$$\widehat{\boldsymbol{H}}_i^{\mathrm{EF}} := \boldsymbol{u}_i\boldsymbol{u}_i^\top. \tag{64}$$

We call this empirical-Fisher-style only because it has the usual outer-product form. It is the outer product of the aggregate sequence-objective gradient, not an identity with the exact Hessian and not the tokenwise empirical Fisher $\sum_t \boldsymbol{u}_{i,t}\boldsymbol{u}_{i,t}^\top$, where $\boldsymbol{u}_{i,t} := \nabla_{\boldsymbol{y}_i}\mathcal{L}_{x(i),i,t}(\boldsymbol{y}_i^\star)$. Indeed, because $\boldsymbol{u}_i = \sum_t \boldsymbol{u}_{i,t}$ under the summed objective, $\boldsymbol{u}_i\boldsymbol{u}_i^\top$ also contains cross-target terms $\sum_{t\neq t'} \boldsymbol{u}_{i,t}\boldsymbol{u}_{i,t'}^\top$. This choice is therefore a practical rank-one curvature surrogate.

**Scalar collapse of rank-one bilinear forms.** Every curvature term required by the two games is a bilinear form $\boldsymbol{p}_i^\top \widehat{\boldsymbol{H}}_i^{\mathrm{EF}} \boldsymbol{w}_i$ evaluated at factual displacements. For the rank-one surrogate this form collapses exactly to a product of two scalar projections,

$$\boldsymbol{p}_i^\top \widehat{\boldsymbol{H}}_i^{\mathrm{EF}} \boldsymbol{w}_i = \boldsymbol{p}_i^\top (\boldsymbol{u}_i\boldsymbol{u}_i^\top)\boldsymbol{w}_i = (\boldsymbol{u}_i^\top \boldsymbol{p}_i)\,(\boldsymbol{u}_i^\top \boldsymbol{w}_i). \tag{65}$$

The identity follows from associativity: the middle contraction $\boldsymbol{u}_i^\top \boldsymbol{w}_i$ is a scalar, so the outer product is never formed. Equivalently, write the gradient and its outer product explicitly as

$$\boldsymbol{u}_i = \begin{bmatrix} u_{i,1} \\ \vdots \\ u_{i,d_y} \end{bmatrix}, \qquad \widehat{\boldsymbol{H}}_i^{\mathrm{EF}} = \underbrace{\begin{bmatrix} u_{i,1} \\ \vdots \\ u_{i,d_y} \end{bmatrix}}_{d_y\times 1} \underbrace{\begin{bmatrix} u_{i,1} & \cdots & u_{i,d_y} \end{bmatrix}}_{1\times d_y} = \underbrace{\begin{bmatrix} u_{i,1}^2 & \cdots & u_{i,1}u_{i,d_y} \\ \vdots & \ddots & \vdots \\ u_{i,d_y}u_{i,1} & \cdots & u_{i,d_y}^2 \end{bmatrix}}_{d_y\times d_y}. \tag{66}$$

Its action on a displacement is therefore

$$\begin{aligned} \widehat{\boldsymbol{H}}_i^{\mathrm{EF}} \boldsymbol{w}_i &= \begin{bmatrix} u_{i,1}^2 & \cdots & u_{i,1}u_{i,d_y} \\ \vdots & \ddots & \vdots \\ u_{i,d_y}u_{i,1} & \cdots & u_{i,d_y}^2 \end{bmatrix} \begin{bmatrix} w_{i,1} \\ \vdots \\ w_{i,d_y} \end{bmatrix} \\ &= \begin{bmatrix} u_{i,1}\sum_{q=1}^{d_y} u_{i,q}w_{i,q} \\ \vdots \\ u_{i,d_y}\sum_{q=1}^{d_y} u_{i,q}w_{i,q} \end{bmatrix} = \boldsymbol{u}_i\Big(\sum_{q=1}^{d_y} u_{i,q}w_{i,q}\Big) = \boldsymbol{u}_i(\boldsymbol{u}_i^\top \boldsymbol{w}_i). \end{aligned} \tag{67}$$

Left-multiplication by $\boldsymbol{p}_i^\top$ then gives the coordinate expansion

$$\begin{aligned} \boldsymbol{p}_i^\top \widehat{\boldsymbol{H}}_i^{\mathrm{EF}} \boldsymbol{w}_i &= \sum_{p=1}^{d_y}\sum_{q=1}^{d_y} p_{i,p}u_{i,p}u_{i,q}w_{i,q} \\ &= \Big(\sum_{p=1}^{d_y} p_{i,p}u_{i,p}\Big)\Big(\sum_{q=1}^{d_y} u_{i,q}w_{i,q}\Big) = (\boldsymbol{u}_i^\top \boldsymbol{p}_i)(\boldsymbol{u}_i^\top \boldsymbol{w}_i). \end{aligned} \tag{68}$$

The double sum factorizes because each matrix entry $u_{i,p}u_{i,q}$ is a product of a row-only factor and a column-only factor. A general Hessian does not separate in this way, which is precisely why its bilinear forms cannot in general be reduced to two scalar projections.

The collapse is exact rather than a further approximation. Provided $\boldsymbol{u}_i \neq 0$, the surrogate has rank one, image $\mathrm{span}\{\boldsymbol{u}_i\}$, and kernel $\boldsymbol{u}_i^{\perp}$, so the curvature-relevant content of any displacement $\boldsymbol{w}_i$ is exactly its projection $\boldsymbol{u}_i^{\top}\boldsymbol{w}_i$; components orthogonal to $\boldsymbol{u}_i$ are annihilated by the surrogate itself (if $\boldsymbol{u}_i = 0$, all surrogate quadratic terms vanish, although the exact Hessian need not be zero). The only approximation in this subsection is therefore the surrogate substitution $\boldsymbol{H}_i \approx \boldsymbol{u}_i\boldsymbol{u}_i^{\top}$ of Equation (64); given that substitution, replacing every quadratic form by products of projections via Equation (65) is an equality chain.

**Active-expert curvature matrix.** The same collapse can be displayed simultaneously for all active experts. Choose an ordering $(e_1, \ldots, e_k)$ of the factual support $\mathcal{S}_i$ and collect its functional displacements and scalar projections as

$$\boldsymbol{D}_i := [\boldsymbol{\delta}_{i,e_1} \quad \cdots \quad \boldsymbol{\delta}_{i,e_k}] \in \mathbb{R}^{d_y \times k}, \qquad \boldsymbol{s}_i := \boldsymbol{D}_i^{\top}\boldsymbol{u}_i = \begin{bmatrix} s_{i,e_1} \\ \vdots \\ s_{i,e_k} \end{bmatrix} \in \mathbb{R}^k. \tag{69}$$

Here $\boldsymbol{D}_i$ is notation local to this derivation. Pulling the $d_y \times d_y$ curvature surrogate back to the $k$ active-expert coordinates gives

$$\begin{aligned} \boldsymbol{D}_i^{\top}\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}\boldsymbol{D}_i = \boldsymbol{D}_i^{\top}(\boldsymbol{u}_i\boldsymbol{u}_i^{\top})\boldsymbol{D}_i = (\boldsymbol{D}_i^{\top}\boldsymbol{u}_i)(\boldsymbol{D}_i^{\top}\boldsymbol{u}_i)^{\top} = \boldsymbol{s}_i\boldsymbol{s}_i^{\top} \\ = \begin{bmatrix} s_{i,e_1}^2 & s_{i,e_1}s_{i,e_2} & \cdots & s_{i,e_1}s_{i,e_k} \\ s_{i,e_2}s_{i,e_1} & s_{i,e_2}^2 & \cdots & s_{i,e_2}s_{i,e_k} \\ \vdots & \vdots & \ddots & \vdots \\ s_{i,e_k}s_{i,e_1} & s_{i,e_k}s_{i,e_2} & \cdots & s_{i,e_k}^2 \end{bmatrix}. \end{aligned} \tag{70}$$

Thus the diagonal entries are the expert self-curvature terms $\boldsymbol{\delta}_{i,e}{}^{\top}\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}\boldsymbol{\delta}_{i,e} = s_{i,e}^2$, while the off-diagonal entries are the pair interactions $\widehat{\psi}_{i,em} = \boldsymbol{\delta}_{i,e}{}^{\top}\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}\boldsymbol{\delta}_{i,m} = s_{i,e}s_{i,m}$. The curvature term in the compact expert score is the row sum of this matrix:

$$\boldsymbol{\delta}_{i,e_j}{}^{\top}\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}\Big(\sum_{m\in\mathcal{S}_i}\boldsymbol{\delta}_{i,m}\Big) = \sum_{\ell=1}^{k} s_{i,e_j}s_{i,e_\ell} = s_{i,e_j}\, s_{i,\Sigma}, \qquad s_{i,\Sigma} = \sum_{\ell=1}^{k} s_{i,e_\ell}. \tag{71}$$

Consequently, neither the $d_y \times d_y$ output-space matrix in Equation (66) nor the $k \times k$ matrix in Equation (70) must be materialized: the complete expert score needs only the $k$ entries of $\boldsymbol{s}_i$ and their sum. The router curvature is the one-column special case, $\boldsymbol{\delta}_{i,r}{}^{\top}\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}\boldsymbol{\delta}_{i,r} = s_{i,r}^2$.

For a mini-batch $\mathcal{B}$ with $n_{\mathcal{B}} := |\mathcal{B}|$, write $\boldsymbol{p} = (\boldsymbol{p}_i)_{i\in\mathcal{B}}$ and $\boldsymbol{w} = (\boldsymbol{w}_i)_{i\in\mathcal{B}}$. The average bilinear form is

$$\widehat{K}_{\mathcal{B}}(\boldsymbol{p}, \boldsymbol{w}) := \frac{1}{n_{\mathcal{B}}}\sum_{i\in\mathcal{B}}\boldsymbol{p}_i^{\top}\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}\boldsymbol{w}_i = \frac{1}{n_{\mathcal{B}}}\sum_{i\in\mathcal{B}}(\boldsymbol{u}_i^{\top}\boldsymbol{p}_i)(\boldsymbol{u}_i^{\top}\boldsymbol{w}_i), \tag{72}$$

where the second equality applies Equation (65) unit by unit.

**Collapsed game scores.** Define the scalar projections $s_{i,e} := \boldsymbol{u}_i^{\top}\boldsymbol{\delta}_{i,e}$, $s_{i,r} := \boldsymbol{u}_i^{\top}\boldsymbol{\delta}_{i,r}$, and $s_{i,\Sigma} := \boldsymbol{u}_i^{\top}\sum_{m\in\mathcal{S}_i}\boldsymbol{\delta}_{i,m}$. Substituting the surrogate into the compact expert Shapley form of Equation (3) and the router form of Equation (9) gives, term by term,

$$\boldsymbol{u}_i^{\top}\boldsymbol{\delta}_{i,e} = s_{i,e}, \qquad \boldsymbol{\delta}_{i,e}{}^{\top}\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}\Big(\sum_{m\in\mathcal{S}_i}\boldsymbol{\delta}_{i,m}\Big) = s_{i,e}\, s_{i,\Sigma}, \qquad \boldsymbol{\delta}_{i,r}{}^{\top}\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}\boldsymbol{\delta}_{i,r} = s_{i,r}^2, \tag{73}$$

where the second identity uses bilinearity to move the support sum into either slot of Equation (65), and the third is the same identity with $\boldsymbol{p}_i = \boldsymbol{w}_i = \boldsymbol{\delta}_{i,r}$. The full per-unit scores therefore reduce to

$$\widehat{\phi}_{i,e} = -s_{i,e} + \frac{1}{2}s_{i,e}s_{i,\Sigma}, \qquad \widehat{\phi}_{i,r} = -s_{i,r} + \frac{1}{2}s_{i,r}^2. \tag{74}$$

If pair diagnostics are required, then $\widehat{\psi}_{i,em} = s_{i,e}s_{i,m}$ by Equation (65). Because the surrogate is positive semidefinite, $\boldsymbol{w}_i^\top \widehat{\boldsymbol{H}}_i^{\mathrm{EF}} \boldsymbol{w}_i = (\boldsymbol{u}_i^\top \boldsymbol{w}_i)^2 \geq 0$ for every displacement, and the sign of $\widehat{\psi}_{i,em} = s_{i,e}s_{i,m}$ is the product of the signs of the two projections, so the super- versus sub-additive reading of the joint-removal residual in the main text can be read off directly from the scalar projections. Exact Hessian–vector products or a generalized Gauss–Newton operator can replace $\widehat{\boldsymbol{H}}_i^{\mathrm{EF}}$ in the same formulas; only the bilinear operator in Equation (65) changes, not the characteristic functions, so the game structure and its axioms are unaffected by the choice of surrogate.

### C.2 Sparse support of the required products

For a token with $k$ active experts, evaluating the *full expert Shapley score* requires $k$ active expert displacements, their $k$ direct projections, and one projection of the total factual MoE output. Thus the summed interaction contribution in Equation (3) does not require storing a dense expert–expert matrix. Explicit pair products are needed only for analyses that request individual $\psi_{em}$ values, such as the interaction diagnostics; in that case there are at most $k(k-1)/2$ co-routed pairs for the token.

Across a corpus, expert-level scores are stored only for experts actually observed in the factual route. Pair-level diagnostic statistics are likewise stored only for observed co-routing pairs. Router attribution adds one expert evaluation per token and layer: the highest-logit excluded expert $b$. The boundary displacement $\boldsymbol{\delta}_r = g_a(\boldsymbol{f}_a - \boldsymbol{f}_b)$ is then scored with the same output-gradient machinery.

### C.3 Streaming aggregation

All quantities in this subsection are computed separately for each MoE layer; thus $e$ denotes a layer-local expert slot and the layer index is suppressed. Let $N_{\mathrm{tok}}$ be the number of attributed probe tokens at that layer and let $n_e$ be the number on which expert $e$ is active. In this subsection only, $\phi_e(\boldsymbol{x}_i)$ denotes whichever token-level expert score is being aggregated (exact, quadratic, or its practical estimate). We distinguish contribution from usage by using separate symbols:

$$\Phi_e^{\mathrm{corpus}} := \frac{1}{N_{\mathrm{tok}}} \sum_{i=1}^{N_{\mathrm{tok}}} \phi_e(\boldsymbol{x}_i), \qquad \Phi_e^{\mathrm{cond}} := \frac{1}{n_e} \sum_{i:e\in\mathcal{S}_k(\boldsymbol{x}_i)} \phi_e(\boldsymbol{x}_i), \qquad \rho_e := \frac{n_e}{N_{\mathrm{tok}}}. \tag{75}$$

For $n_e > 0$, inactive tokens contribute $\phi_e(\boldsymbol{x}_i) = 0$ by the null-player extension, so

$$\Phi_e^{\mathrm{corpus}} = \rho_e \, \Phi_e^{\mathrm{cond}}. \tag{76}$$

If $n_e = 0$, we set $\rho_e = 0$ and $\Phi_e^{\mathrm{corpus}} = 0$, while the conditional mean is undefined and is omitted. For observed experts, this identity makes the usage–contribution distinction explicit: corpus value combines how often an expert is selected with its conditional value when selected, whereas $\Phi_e^{\mathrm{cond}}$ isolates the latter.

For the RQ4 interaction ablation we also define the quadratic self-only token score and its conditional mean,

$$\phi_{e,\mathrm{self}}^{(2)}(\boldsymbol{x}) := -\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e + \frac{1}{2} \boldsymbol{\delta}_e{}^\top \boldsymbol{H}_\star \boldsymbol{\delta}_e, \qquad \Phi_{e,\mathrm{self}}^{\mathrm{cond}} := \frac{1}{n_e} \sum_{i:e\in\mathcal{S}_i} \phi_{e,\mathrm{self}}^{(2)}(\boldsymbol{x}_i). \tag{77}$$

The matched corpus-level ablation score is therefore $\rho_e \Phi_{e,\mathrm{self}}^{\mathrm{cond}}$: it retains exactly the same routing-frequency factor as the full corpus contribution while deleting only the co-routed pair term. We denote this baseline by Self$\times\rho$. Accordingly, native routing frequency is denoted by $\rho_e$ throughout and is never assigned the Shapley symbol $\phi_e$.

Router scores are aggregated over the same attributed units at the layer or routing-block level. The associated causal intervention uses the same token-wise boundary pair and gate-mass exchange, so scoring and evaluation share the same local counterfactual.

### C.4 Computational complexity

Let $L$ be the number of attributed MoE layers, $E$ the number of experts per layer, and $d_y$ the MoE output width. The selected expert outputs are already available from the factual sparse forward pass.

At one token and one layer, forming the $k$ gated displacements and their projected gradients costs $\mathcal{O}(kd_y)$ arithmetic. The total projection $s_{i,\Sigma}$ is obtained by summing the $k$ scalar projections or directly projecting $\boldsymbol{y}_i^\star$, so the *full Shapley score for all active experts* adds only $\mathcal{O}(k)$ scalar work after projection. Scoring all attributed layers therefore costs $\mathcal{O}(Lkd_y)$ projection arithmetic per token, in addition to the ordinary forward and backward passes.

If all individual pair diagnostics are requested, their scalar products add $\mathcal{O}(k^2)$ work and at most $k(k-1)/2$ temporary pair values per token and layer. This quadratic-in-$k$ cost is diagnostic rather than intrinsic to the full expert value. Let $P_{\mathrm{obs}}$ be the number of distinct layer-local expert pairs observed across the corpus. Stored aggregate statistics then cost $\mathcal{O}(LE + P_{\mathrm{obs}})$; this is sparse in practice but has the worst-case bound $P_{\mathrm{obs}} = \mathcal{O}(LE^2)$ if every pair is observed.

Router attribution first identifies the highest-logit excluded expert. This search costs $\mathcal{O}(E)$ per token and layer unless the router already exposes a cached top-$(k+1)$ order statistic. It then evaluates that excluded expert once; writing $C_{\mathrm{exp}}$ for the cost of one expert forward, the additional cost across all attributed layers is $\mathcal{O}(L(E+C_{\mathrm{exp}}+d_y))$, or $\mathcal{O}(L(C_{\mathrm{exp}}+d_y))$ when boundary lookup is cached. The final $d_y$ term forms and projects the one-swap displacement. The rank-one implementation requires one ordinary backward pass for the probe batch, no expert-parameter Jacobians, and no materialized Hessian/Fisher blocks. Exact-Hessian or GGN variants may require additional Hessian–vector products, but they leave the counterfactual games and closed-form Shapley algebra unchanged.

# D Detailed Experimental Protocols

This section records the percentage-based intervention protocols and the full supporting grids used by the revised empirical study. Expert sets are always constructed within each MoE layer.

## D.1 Backbones and model-specific routing structure

We use OLMoE-1B-7B-Instruct (Muennighoff et al., 2025), Qwen3-30B-A3B-Instruct-2507 (Yang et al., 2025), gpt-oss-20b (OpenAI, 2025), and Granite-3.1-3B-A800M-Instruct (Granite Team, IBM, 2024). OLMoE has 16 MoE layers with 64 experts and top-8 activation; Qwen3 has 48 layers with 128 experts and top-8 activation; gpt-oss has 24 layers with 32 experts and top-4 activation; Granite has 32 sparse layers with 40 experts and top-8 activation. OLMoE keeps its original non-renormalized top-8 gate weights, whereas Granite normalizes over its selected top-8 set.

Table 9: MoE backbones. Total/active parameters and routing structure.

| Model | Total params | Active params | MoE layers | Experts/layer | Active experts |
|---|---|---|---|---|---|
| Qwen3-30B-A3B-Instruct-2507 | 30.5B | 3.3B | 48 | 128 | 8 |
| gpt-oss-20b | 21B | 3.6B | 24 | 32 | 4 |
| OLMoE-1B-7B-Instruct | 6.9B | 1.3B | 16 | 64 | 8 |
| Granite-3.1-3B-A800M-Instruct | 3.0B | 0.8B | 32 | 40 | 8 |

## D.2 Data separation, attribution loss, and scoring

WikiText-2 validation text is used for attribution and test text for intervention PPL. GSM8K uses training examples for attribution and the full 1,319-example test set for zero-shot chat exact-match evaluation. MMLU uses validation examples for attribution and its 14,042-example test split for answer-choice accuracy. MedQA uses its training split for attribution and 1,273 test questions for accuracy. The attribution and evaluation splits are therefore disjoint. GSM8K uses greedy decoding with `max_new_tokens=768`; gpt-oss uses the Harmony template with low reasoning effort.

**Differentiable attribution loss.** All probe gradients use exactly the same type of scalar objective for every task: summed teacher-forced causal cross-entropy on gold target tokens. For a probe example $x$, let $m_t \in \{0,1\}$ be the supervision mask, $\mathcal{T}_x := \{t : m_t = 1\}$ its supervised positions, and $c_t$ the gold token at target position $t$. Let $\boldsymbol{\theta}$ denote the fixed model parameters. The factual

scalar objective is

$$\mathcal{L}_x := -\sum_t m_t \log p_{\boldsymbol{\theta}}(c_t \mid x_{<t}) = -\sum_{t \in \mathcal{T}_x} \log p_{\boldsymbol{\theta}}(c_t \mid x_{<t})\,. \tag{78}$$

For a fixed attributed unit $i$, let $p_{\boldsymbol{\theta}}(\cdot \mid x_{<t}; \boldsymbol{y}_i)$ denote the next-token distribution after substituting only its MoE output row by $\boldsymbol{y}_i$. This defines the intervention slice $\mathcal{L}_{x,i}(\boldsymbol{y}_i)$, whose factual value at $\boldsymbol{y}_i^\star$ equals $\mathcal{L}_x$. The reduction is always a *sum*, never a mean over supervised tokens. Thus, for a unit $i$ belonging to example $x(i)$,

$$\boldsymbol{u}_i := \nabla_{\boldsymbol{y}_i} \mathcal{L}_{x(i),i}(\boldsymbol{y}_i^\star) = \sum_{t \in \mathcal{T}_{x(i)}} \nabla_{\boldsymbol{y}_i} \mathcal{L}_{x(i),i,t}(\boldsymbol{y}_i^\star), \qquad \mathcal{L}_{x(i),i,t}(\boldsymbol{y}_i) := -\log p_{\boldsymbol{\theta}}(c_t \mid x(i)_{<t}; \boldsymbol{y}_i). \tag{79}$$

Terms that are not causally downstream of $\boldsymbol{y}_i$ are zero. In particular, an intermediate token representation can influence its own and later supervised targets through causal attention, so we do not assume a one-to-one position–loss correspondence. Using a token mean would multiply every such unit gradient from an example by $1/|\mathcal{T}_x|$; the summed objective instead attributes the total gold-continuation negative log-likelihood without a length-normalization factor. This reduction is part of the estimator definition, not an innocuous display convention: scaling $\mathcal{L}_x$ by a constant $c$ scales the direct term by $c$ but the gradient-outer-product curvature term by $c^2$. Accordingly, sum and mean reductions need not induce identical rankings, and we hold the summed reduction fixed across every reported method and task. For a mini-batch, the backward scalar is likewise the sum of its per-example $\mathcal{L}_x$ objectives. A backward hook on a MoE output tensor therefore returns each row gradient $\boldsymbol{u}_i$ in Equation (79); it does not require a separate backward pass or a one-to-one assignment between hidden rows and target positions.

The supervised target set is task-specific while the loss form is shared.

- **WikiText-2.** Tokenized text is divided into fixed-length contiguous chunks with no padding. Every shifted next-token target in the chunk is supervised, equivalently every input position except the last predicts its gold successor.
- **GSM8K.** The prompt tokens are masked from the loss. We teacher-force the complete gold completion—the full reasoning chain and final answer span (including the standard `#### N` answer format)—and exclude the terminal EOS token from the target mask.
- **MMLU/MedQA.** The prompt requests a single answer label and the target set contains only the gold answer-letter token. The resulting teacher-forced token cross-entropy is the differentiable classification objective; each model-specific prompt is checked so that the serialized answer label is exactly one supervised tokenizer token. For gpt-oss, the Harmony final-channel suffix `<|channel|>final<|message|>` is appended immediately before the answer label so that the supervised token is scored in the intended output channel; channel-control tokens themselves are masked.

ACC is therefore a final downstream evaluation metric only and never participates in gradient extraction or any other backpropagation. Intervention evaluation uses PPL for WikiText-2, greedy-decoding exact-match ACC for GSM8K, and answer-label ACC for MMLU/MedQA, where the prediction is the largest logit among the valid answer labels. Metric changes are always computed against the baseline in the same run, but we use two explicitly labeled accuracy conventions. RQ2 reports accuracy damage, $\Delta\mathrm{ACC}_{\mathrm{damage}} := \mathrm{ACC}_{\mathrm{base}} - \mathrm{ACC}_{\mathrm{int}}$, so larger positive values mean greater damage. RQ3 reports signed accuracy change, $\Delta\mathrm{ACC}_{\mathrm{change}} := \mathrm{ACC}_{\mathrm{int}} - \mathrm{ACC}_{\mathrm{base}}$, so more negative values mean greater damage. Whenever a perplexity change is reported, $\Delta\mathrm{PPL} := \mathrm{PPL}_{\mathrm{int}} - \mathrm{PPL}_{\mathrm{base}}$, so larger positive values mean greater damage. Other intervention tables report absolute post-intervention ACC or PPL as stated in their captions.

### D.3 RQ1: USAGE–CONTRIBUTION SEPARATION AND PERCENTAGE REMOVAL

RQ1 combines a structural comparison with matched causal intervention. We first compare conditional expert value with native routing frequency using Spearman and Pearson correlation; we denote the Spearman rank correlation by $r_s$. A *rare high-value* slot has routing frequency below the within-probe median and conditional value at or above the 90th percentile. This diagnostic asks whether usage and contribution induce the same ranking before any expert is removed.

Throughout the intervention and adaptation protocols, *Contribution* without a qualifier denotes the practical estimate of the signed corpus score $\Phi_e^{\mathrm{corpus}}$ from Equation (75): Top groups sort it in descending order and Bottom groups in ascending order; we do not rank by absolute magnitude. The conditional score $\Phi_e^{\mathrm{cond}}$ is used only when the goal is explicitly to separate value-on-activation from usage, namely the usage–value diagnostic above and the case-study contrast sets in Section D.9.

For the causal test, each layer with $E$ experts contributes $\lceil pE \rceil$ experts to a percentage group. Top 1% and Top 3% groups are removed by output-zero, which keeps the factual route and gate weights fixed. Bottom 10% and Bottom 20% groups are removed by logit masking, which forces rerouting. We compare **CONTRAIL** (Ours) with native routing frequency $\rho_e$, mean gate weight, gradient score, SHAPE, Shapley-MoE, and same-budget random selection with five seeds. Every method therefore uses the same Top 1%/3% and Bottom 10%/20% intervention budget. The percentage definition approximately normalizes intervention scale across architectures with different expert counts; the ceiling operation can make the realized fraction slightly larger at small budgets.

### D.4 RQ2: TASK-SPECIFIC CROSS-TASK INTERVENTION

For each model, we independently compute task-conditioned Contribution rankings for MMLU, GSM8K, and MedQA. For each $p \in \{1\%, 3\%\}$ and selection task $X$, the top-$p$ set is output-zeroed and evaluated on every task $Y \in \{\mathrm{MMLU}, \mathrm{GSM8K}, \mathrm{MedQA}\}$. This yields a $3 \times 3$ causal matrix per model and budget. Diagonal dominance is evaluated within each evaluation task. Jaccard overlap between the selected expert sets and layer-by-expert value heatmaps are reported as structural diagnostics.

### D.5 RQ3: ROUTER ESTIMATORS AND BOUNDARY-EXCHANGE PROTOCOL

**Fixed estimand and comparison axis.** This subsection does not redefine the routing counterfactual. It imports from Section B.5 the nearest boundary pair, reflected logit exchange, and mass-matched functional endpoint, and asks only how alternative backward estimators rank the sensitivity of that fixed intervention across layers. For attributed unit $i$ at MoE layer $\lambda$, the indexed quantities established there are

$$\begin{gathered} \boldsymbol{v}_i := \boldsymbol{e}_{a_i} - \boldsymbol{e}_{b_i}, \qquad \gamma_i := \ell_{i,a_i} - \ell_{i,b_i} > 0, \qquad \boldsymbol{\Delta\ell}_i := \boldsymbol{\ell}_i^- - \boldsymbol{\ell}_i = -\gamma_i \boldsymbol{v}_i, \\ \boldsymbol{\delta}_{i,r} := \mu_i(\boldsymbol{f}_{i,a_i} - \boldsymbol{f}_{i,b_i}), \qquad \boldsymbol{y}_i^- := \boldsymbol{y}_i^\star - \boldsymbol{\delta}_{i,r}. \end{gathered} \tag{80}$$

Here $a_i$ is the lowest-logit selected expert, $b_i$ is the highest-logit excluded expert, and $\mu_i$ is the factual gate mass of $a_i$. These objects are held common across all methods; in particular, estimator-specific scores are never used to change the boundary pair or the intervention magnitude.

**Generic first-order score.** A local backward rule is represented at the factual point by a pseudo-Jacobian $\boldsymbol{J}_i \in \mathbb{R}^{E\times E}$ from logit displacement to gate displacement. This includes STEs (Bengio et al., 2013): although their hard forward map is not differentiated across the support change, their chosen backward linear operator is still expressible as a pseudo-Jacobian. Let $\boldsymbol{F}_i := [\boldsymbol{f}_{i,1}, \ldots, \boldsymbol{f}_{i,E}] \in \mathbb{R}^{d_y \times E}$ collect the expert outputs conceptually. We evaluate every first-order rule on the same finite reflected direction $\boldsymbol{\Delta\ell}_i$ via

$$\chi_i(\boldsymbol{J}_i) := \boldsymbol{u}_i^\top \boldsymbol{F}_i \boldsymbol{J}_i \boldsymbol{\Delta\ell}_i. \tag{81}$$

The dimensions contract from right to left as

$$\underbrace{\boldsymbol{u}_i^\top}_{1\times d_y}\ \underbrace{\boldsymbol{F}_i}_{d_y\times E}\ \underbrace{\boldsymbol{J}_i}_{E\times E}\ \underbrace{\boldsymbol{\Delta\ell}_i}_{E\times 1} \in \mathbb{R}. \tag{82}$$

Thus $\boldsymbol{J}_i\boldsymbol{\Delta\ell}_i$ is the gate displacement implied by the backward rule, $\boldsymbol{F}_i\boldsymbol{J}_i\boldsymbol{\Delta\ell}_i$ is its induced functional displacement, and the final contraction is the corresponding first-order loss change. This common functional score is the experimental comparison axis; it should not be confused with the theoretical derivation of $\boldsymbol{J}_i^{\mathrm{CCGE}}$ in Section B.5. Only columns of $\boldsymbol{F}_i$ touched by $\boldsymbol{J}_i$ are needed in implementation. All five estimators preserve the same factual hard forward pass and differ only in the backward operator or in whether the common output-space curvature correction is added:

1. **Hard-route gradient.** $\boldsymbol{J}_i^{\mathrm{hard}}$ is the native Jacobian of the post-routing gate vector with the realized top-$k$ support treated as fixed by ordinary autodiff. Its token score is $\chi_i^{\mathrm{hard}} := \chi_i(\boldsymbol{J}_i^{\mathrm{hard}})$; the excluded expert has no membership derivative under this backward pass.
2. **Pairwise Identity straight-through estimator (STE).** On the boundary pair, the hard membership switch is assigned a unit-slope symmetric transfer in backward,
$$\boldsymbol{J}_i^{\mathrm{id}} := \frac{\mu_i}{2}\boldsymbol{v}_i\boldsymbol{v}_i^\top, \qquad \chi_i^{\mathrm{id}} := \chi_i(\boldsymbol{J}_i^{\mathrm{id}}). \tag{83}$$
The forward route remains hard. This is the classical STE prescription: a generic unit slope at the discontinuity, chosen without reference to the routing margin $\gamma_i$.
3. **Pairwise Sigmoid STE.** We replace the pairwise switch in backward by the unit-temperature sigmoid transfer $\sigma(\xi) := (1+\mathrm{e}^{-\xi})^{-1}$ and $\xi_i := \boldsymbol{v}_i^\top \boldsymbol{\ell}_i = \gamma_i$ at the factual point. Its local Jacobian is
$$\boldsymbol{J}_i^{\mathrm{sig}} := \mu_i\sigma(\gamma_i)\big(1-\sigma(\gamma_i)\big)\boldsymbol{v}_i\boldsymbol{v}_i^\top, \qquad \chi_i^{\mathrm{sig}} := \chi_i(\boldsymbol{J}_i^{\mathrm{sig}}). \tag{84}$$
4. **CCGE-linear.** We use the margin-normalized Jacobian from Equation (37). Since $\boldsymbol{J}_i^{\mathrm{CCGE}}\boldsymbol{\Delta\ell}_i = -\mu_i\boldsymbol{v}_i$, its token score is
$$\chi_i^{\mathrm{lin}} := \chi_i(\boldsymbol{J}_i^{\mathrm{CCGE}}) = -\boldsymbol{u}_i^\top\boldsymbol{\delta}_{i,r}. \tag{85}$$
5. **Full CCGE.** The full route score adds the same local output-space curvature correction used by CONTRAIL:
$$\chi_i^{\mathrm{full}} := -\boldsymbol{u}_i^\top\boldsymbol{\delta}_{i,r} + \frac{1}{2}\boldsymbol{\delta}_{i,r}{}^\top\widehat{\boldsymbol{H}}_i\boldsymbol{\delta}_{i,r}. \tag{86}$$
Under the empirical-Fisher-style surrogate this becomes $\chi_i^{\mathrm{full}} = -s_{i,r} + \frac{1}{2}s_{i,r}^2$, with $s_{i,r} := \boldsymbol{u}_i^\top\boldsymbol{\delta}_{i,r}$.

Table 10: Unified view of the five router estimators used in RQ3. All methods use the same factual boundary pair $(a_i, b_i)$ and the same reflected-logit direction $\boldsymbol{\Delta\ell}_i = -\gamma_i\boldsymbol{v}_i$. The third column shows the output-space displacement induced by the backward operator along that common direction; full CCGE uses the same endpoint-aligned first-order displacement as CCGE-linear and additionally includes a second-order curvature correction.

| Method | Backward Jacobian $\boldsymbol{J}_i$ | Induced output displacement $\boldsymbol{F}_i\boldsymbol{J}_i\boldsymbol{\Delta\ell}_i$ | Token score |
|---|---|---|---|
| Hard-route gradient | $\boldsymbol{J}_i^{\mathrm{hard}}$ | $\boldsymbol{F}_i\boldsymbol{J}_i^{\mathrm{hard}}\boldsymbol{\Delta\ell}_i$ (native fixed-support autodiff) | $\chi_i^{\mathrm{hard}} = \chi_i(\boldsymbol{J}_i^{\mathrm{hard}})$ |
| Identity STE | $\frac{\mu_i}{2}\boldsymbol{v}_i\boldsymbol{v}_i^\top$ | $-\gamma_i\boldsymbol{\delta}_{i,r}$ | $\chi_i^{\mathrm{id}} = \boldsymbol{u}_i^\top(-\gamma_i\boldsymbol{\delta}_{i,r})$ |
| Sigmoid STE | $\mu_i\sigma(\gamma_i)(1-\sigma(\gamma_i))\boldsymbol{v}_i\boldsymbol{v}_i^\top$ | $-2\gamma_i\sigma(\gamma_i)(1-\sigma(\gamma_i))\boldsymbol{\delta}_{i,r}$ | $\chi_i^{\mathrm{sig}} = \boldsymbol{u}_i^\top[-2\gamma_i\sigma(\gamma_i)(1-\sigma(\gamma_i))\boldsymbol{\delta}_{i,r}]$ |
| CCGE-linear | $\frac{\mu_i}{2\gamma_i}\boldsymbol{v}_i\boldsymbol{v}_i^\top$ | $-\boldsymbol{\delta}_{i,r}$ | $\chi_i^{\mathrm{lin}} = -\boldsymbol{u}_i^\top\boldsymbol{\delta}_{i,r}$ |
| CCGE (Ours) | $\frac{\mu_i}{2\gamma_i}\boldsymbol{v}_i\boldsymbol{v}_i^\top$ | $-\boldsymbol{\delta}_{i,r}$ | $\chi_i^{\mathrm{full}} = -\boldsymbol{u}_i^\top\boldsymbol{\delta}_{i,r} + \frac{1}{2}\boldsymbol{\delta}_{i,r}{}^\top\widehat{\boldsymbol{H}}_i\boldsymbol{\delta}_{i,r}$ |

**Induced-scale comparison.** The table compares the estimators as alternative scoring rules. The three pairwise backward rules share the rank-one family $\boldsymbol{J}_i(\alpha_i) = \alpha_i\boldsymbol{v}_i\boldsymbol{v}_i^\top$ derived in Equation (35); within the RQ3 protocol, their only difference is the scale $\alpha_i$ assigned to the common exchange direction. Since $\boldsymbol{v}_i^\top\boldsymbol{v}_i = 2$, define the dimensionless induced-endpoint multiplier $\beta_i(\alpha_i) := 2\alpha_i\gamma_i/\mu_i$. Then
$$\begin{aligned}\boldsymbol{F}_i\boldsymbol{J}_i(\alpha_i)\boldsymbol{\Delta\ell}_i &= -2\alpha_i\gamma_i(\boldsymbol{f}_{i,a_i} - \boldsymbol{f}_{i,b_i}) = -\beta_i(\alpha_i)\boldsymbol{\delta}_{i,r},\\ \chi_i(\boldsymbol{J}_i(\alpha_i)) &= -\beta_i(\alpha_i)s_{i,r}, \qquad s_{i,r} := \boldsymbol{u}_i^\top\boldsymbol{\delta}_{i,r}.\end{aligned} \tag{87}$$
Thus $\beta_i = 1$ means that the backward rule is calibrated to the finite one-swap endpoint, while $\beta_i \neq 1$ introduces a token- and layer-dependent rescaling before aggregation. Substituting the three prescribed Jacobians gives
$$\beta_i^{\mathrm{id}} = \gamma_i, \qquad \beta_i^{\mathrm{sig}} = 2\gamma_i\sigma(\gamma_i)(1-\sigma(\gamma_i)), \qquad \beta_i^{\mathrm{CCGE}} = 1. \tag{88}$$
The Identity STE therefore weights the same functional direction by the raw routing margin. The Sigmoid STE introduces both the margin factor and sigmoid saturation, downweighting large-margin decisions. By contrast, the CCGE value $\beta_i = 1$ is not re-derived here: it is the endpoint

calibration proved in Equations (36) and (44). RQ3 tests whether this theoretically fixed calibration produces a better ranking than the two generic STE scales.

The hard-route gradient lies outside the pairwise family because it differentiates only the realized fixed-support gate map: the excluded expert has no membership derivative, so its induced functional perturbation need not be collinear with $-\boldsymbol{\delta}_{i,r}$. Full CCGE uses exactly the same endpoint-aligned first-order displacement as CCGE-linear and adds $\frac{1}{2}\boldsymbol{\delta}_{i,r}{}^{\top}\widehat{\boldsymbol{H}}_i\boldsymbol{\delta}_{i,r}$. Their comparison therefore changes only the order of the local downstream loss model—linear versus quadratic—while holding the boundary pair, finite endpoint, and first-order displacement fixed. This isolates the empirical contribution of output-space curvature.

**Aggregation and matched causal evaluation.** Let $\mathcal{M} := \{\mathrm{hard}, \mathrm{id}, \mathrm{sig}, \mathrm{lin}, \mathrm{full}\}$ index the five estimators, and let $N_\lambda$ be the number of attributed units scored at layer $\lambda$. Write $\chi_{i,\lambda}^{(\nu)}$ for estimator $\nu$'s score on attributed unit $i$ at that layer. For every $\nu \in \mathcal{M}$, token scores are aggregated by the same signed arithmetic mean,

$$Q_\lambda^{(\nu)} := \frac{1}{N_\lambda}\sum_{i=1}^{N_\lambda}\chi_{i,\lambda}^{(\nu)}. \tag{89}$$

The three layers with the largest $Q_\lambda^{(\nu)}$ are selected. We do not take absolute values and do not use intervention outcomes to choose layers. At the selected layers we then apply the identical nearest-boundary one-swap intervention: the lowest-logit selected expert is exchanged with the highest-logit excluded expert and the factual gate mass is transferred to the replacement. This common endpoint isolates ranking quality from intervention strength. The resulting causal comparison is reported in Table 4.

### D.6 RQ4: CO-ROUTED INTERACTION ABLATION

The interaction ablation compares the full corpus Contribution score $\Phi_e^{\mathrm{corpus}}$ with Self$\times\rho$, i.e., $\rho_e\Phi_{e,\mathrm{self}}^{\mathrm{cond}}$ from Equation (77). The latter removes the co-routed pair term while retaining the same routing-frequency factor. We sweep Top 1%, 3%, and 9% groups under output-zero. The 1% and 3% Contribution/output-zero cells reuse the matched RQ1 groups; 9% increases coalition size so that pair terms can accumulate. The reported comparison is the matched intervention damage in Table 5; we do not infer a set-membership change rate without separately reporting the corresponding overlaps.

### D.7 RQ5: SPARSE EXPERT FINE-TUNING

The sparse-adaptation experiment uses the same per-layer percentage semantics as RQ1. For each downstream task, the router, attention blocks, embeddings, layer norms, and all unselected expert FFNs are frozen; only expert FFNs in the top 1%/3%/6%/10% of the task-conditioned ranking are trainable. Selection methods are **CONTRAIL** (Ours), native routing frequency, random selection, mean gate weight, gradient score, SHAPE, and Shapley-MoE. All methods start from the same unmodified instruction-tuned checkpoint and share the optimizer, learning-rate schedule, examples, update count, batch size, and evaluation protocol. We report task performance and the fraction of trainable expert parameters so that adaptation gain is compared at a matched trainable-expert budget.

### D.8 SUPPORTING PROBE-SIZE STABILITY

Let $\mathcal{R}_n$ denote the expert ranking computed from the first $n$ probe examples and $\mathcal{R}_{\mathrm{full}}$ the ranking from the complete probe, and let $\mathcal{N}$ be the finite set of evaluated prefix sizes. We define

$$n_{0.9} := \min\left\{n \in \mathcal{N} : \tau(\mathcal{R}_n, \mathcal{R}_{\mathrm{full}}) \geq 0.9\right\},$$

where $\tau$ is Kendall rank correlation; if the displayed set is empty, we set $n_{0.9} := \infty$. We report $n_{0.9}$ for both the direct score and the full interaction-aware score. This quantifies the amount of probe data required before the expert ordering stabilizes, separating finite-probe estimation variability from differences between attribution rules.

### D.9 MATCHED GSM8K CASE-STUDY SELECTION

The qualitative cases isolate the frequency–contribution distinction at the level of individual generations. For each model, we construct two contrastive sets of 40 expert slots from the empirical frequency–value plane using the signed conditional score $\Phi_e^{\mathrm{cond}}$: a *rare/high-contribution* set with low routing frequency and high conditional value, and a *frequent/low-contribution* set with high routing frequency and low conditional value. Each set is removed with the same output-zero intervention used by the fixed-route expert game, so the factual route and recorded gate weights are unchanged.

We scan every GSM8K example. Let $d_R(x)$ and $d_F(x)$ denote the absolute per-example PPL changes after deleting the rare/high and frequent/low sets, respectively. We quantify joint impact and matching by

$$\begin{aligned} I(x) &:= \min\{d_R(x), d_F(x)\}, \\ A_{\mathrm{rel}}(x) &:= \frac{|d_R(x) - d_F(x)|}{\max\{d_R(x), d_F(x)\} + 10^{-12}}, \\ S_{\mathrm{case}}(x) &:= \frac{I(x)}{1 + A_{\mathrm{rel}}(x)}. \end{aligned} \tag{90}$$

Here $10^{-12}$ is a numerical stabilizer for the zero-change case. A high $S_{\mathrm{case}}$ therefore requires *both* deletions to have a large effect on the example while penalizing unequal perturbation magnitudes. Within each backbone we rank all GSM8K examples by this score before inspecting the generated responses. The displayed cases are taken from the highest-ranked matched examples while ensuring backbone coverage; answer correctness, whether an answer flips, and the visual appearance of the response are not used in the selection score. This makes the qualitative inspection conditional on a fixed, response-independent selection rule rather than on post-hoc answer behavior.

Matching the perturbation magnitude reduces the confound that one deletion simply creates a much larger generic disturbance. It does not make the two interventions identical in representation space, so the cases are used as qualitative complements to the aggregate causal results rather than as a standalone quantitative estimate (Section E.2).

## E SUPPORTING RESULTS IN RQ ORDER

### E.1 RQ1: PERCENTAGE-REMOVAL ROBUSTNESS AND BASELINE DEFINITIONS

The main paper compares Random, Routing Frequency, Shapley-MoE, and **CONTRAIL**. Here we report the complete cross-method percentage-removal matrix under the same causal budget, adding mean gate weight, gradient score, and SHAPE. Random selection uses seeds 42–46. High-value removal is evaluated by how strongly it damages the task metric, while low-value removal is evaluated by how closely it preserves the pretrained model.

All baselines use the same layer-wise percentage budgets and intervention semantics; they differ only in the ranking score. Routing frequency is $\rho_e$ from Equation (75). Mean gate weight is the active-token conditional mean $n_e^{-1} \sum_{i:e\in\mathcal{S}_i} g_{i,e}$. Gradient score is the corpus-aggregated signed first-order term $-\boldsymbol{u}_i^\top \boldsymbol{\delta}_{i,e}$, using the same inactive-token zeros and normalization as $\Phi_e^{\mathrm{corpus}}$. SHAPE (Zhang, 2026) and Shapley-MoE (Huang et al., 2025) use their published Shapley-style ranking rules with the same probes and intervention budgets. This common causal operator isolates the quality of the ranking itself.

The complete comparison in Table 11 exposes three levels of information in an expert ranking. Routing frequency and mean gate weight primarily measure allocation: they identify experts that receive tokens or gate mass. Gradient and SHAPE add local sensitivity, so their rankings more often concentrate damaging experts near the top. Shapley-MoE and **CONTRAIL** further account for the utility of expert computation in its routed context, which tends to concentrate the most damaging removals near the top of the ranking across the model–task grid.

The bottom-removal blocks probe a complementary property of the ranking: whether experts assigned low fixed-route contribution remain comparatively lower-impact once the router is allowed to compensate. They test whether the low end of a ranking remains benign under a support-changing

Table 11: Complete RQ1 percentage-removal comparison. All expert-ranking baselines use the same layer-wise Top 1%/3% output-zero and Bottom 10%/20% logit-mask interventions. Random reports the five-seed mean; $\Delta\%$ is relative to the unmodified checkpoint.

| Method | OLMoE | | | Qwen3 | | | gpt-oss | | | Granite | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC | WT-2 PPL | GSM8K ACC | MMLU ACC |
| ***Unmodified checkpoint*** | | | | | | | | | | | | |
| Unmodified checkpoint | 13.73 | 0.525 | 0.487 | 7.23 | 0.933 | 0.760 | 25.57 | 0.894 | 0.698 | 7.41 | 0.602 | 0.505 |
| ***Top 1%: output-zero (stronger damage: PPL↑, ACC↓)*** | | | | | | | | | | | | |
| **CONTRAIL** (Ours) | 19.76 Δ +43.9% | 0.140 Δ -73.3% | 0.405 Δ -16.8% | 8.54 Δ +18.1% | 0.793 Δ -15.0% | 0.736 Δ -3.1% | 35.51 Δ +38.9% | 0.878 Δ -1.8% | 0.686 Δ -1.7% | 8.79 Δ +18.6% | 0.024 Δ -96.0% | 0.403 Δ -20.2% |
| Shapley-MoE | 18.06 Δ +31.5% | 0.255 Δ -51.4% | 0.428 Δ -12.1% | 8.49 Δ +17.4% | 0.798 Δ -14.5% | 0.738 Δ -2.9% | 34.47 Δ +34.8% | 0.881 Δ -1.4% | 0.687 Δ -1.5% | 8.52 Δ +15.0% | 0.185 Δ -69.3% | 0.425 Δ -15.7% |
| SHAPE | 17.03 Δ +24.1% | 0.324 Δ -38.2% | 0.442 Δ -9.2% | 8.33 Δ +15.2% | 0.815 Δ -12.7% | 0.741 Δ -2.5% | 33.85 Δ +32.4% | 0.883 Δ -1.2% | 0.688 Δ -1.4% | 8.36 Δ +12.8% | 0.281 Δ -53.3% | 0.439 Δ -13.1% |
| Gradient score | 16.58 Δ +20.8% | 0.355 Δ -32.4% | 0.448 Δ -8.0% | 8.25 Δ +14.0% | 0.824 Δ -11.7% | 0.743 Δ -2.2% | 33.57 Δ +31.3% | 0.884 Δ -1.1% | 0.688 Δ -1.4% | 8.29 Δ +11.8% | 0.324 Δ -46.2% | 0.445 Δ -11.9% |
| Mean gate weight | 15.78 Δ +15.0% | 0.409 Δ -22.1% | 0.459 Δ -5.8% | 8.15 Δ +12.7% | 0.834 Δ -10.6% | 0.745 Δ -1.9% | 33.09 Δ +29.4% | 0.886 Δ -0.9% | 0.689 Δ -1.3% | 8.16 Δ +10.1% | 0.399 Δ -33.7% | 0.456 Δ -9.8% |
| Routing frequency | 15.12 Δ +10.1% | 0.255 Δ -51.4% | 0.352 Δ -27.7% | 8.29 Δ +14.6% | 0.817 Δ -12.5% | 0.745 Δ -1.9% | 24.06 Δ -5.9% | 0.854 Δ -4.5% | 0.687 Δ -1.6% | 8.45 Δ +14.0% | 0.236 Δ -60.8% | 0.427 Δ -15.4% |
| Random | 14.08 Δ +2.5% | 0.524 Δ -0.2% | 0.482 Δ -1.0% | 7.35 Δ +1.6% | 0.928 Δ -0.6% | 0.755 Δ -0.6% | 32.05 Δ +25.3% | 0.889 Δ -0.6% | 0.690 Δ -1.1% | 7.89 Δ +6.5% | 0.560 Δ -7.0% | 0.478 Δ -5.3% |
| ***Top 3%: output-zero (stronger damage: PPL↑, ACC↓)*** | | | | | | | | | | | | |
| **CONTRAIL** (Ours) | 21.85 Δ +59.1% | 0.036 Δ -93.1% | 0.384 Δ -21.1% | 10.02 Δ +38.5% | 0.060 Δ -93.6% | 0.675 Δ -11.1% | 28.64 Δ +12.0% | 0.743 Δ -16.9% | 0.659 Δ -5.6% | 14.18 Δ +91.4% | 0.011 Δ -98.2% | 0.282 Δ -44.2% |
| Shapley-MoE | 19.56 Δ +42.5% | 0.180 Δ -65.8% | 0.413 Δ -15.2% | 9.92 Δ +37.2% | 0.107 Δ -88.5% | 0.682 Δ -10.2% | 26.69 Δ +4.4% | 0.783 Δ -12.4% | 0.663 Δ -5.0% | 12.51 Δ +68.8% | 0.145 Δ -75.9% | 0.329 Δ -34.8% |
| SHAPE | 18.19 Δ +32.5% | 0.266 Δ -49.3% | 0.430 Δ -11.7% | 9.59 Δ +32.6% | 0.229 Δ -75.4% | 0.695 Δ -8.5% | 25.52 Δ -0.2% | 0.807 Δ -9.7% | 0.666 Δ -4.6% | 11.51 Δ +55.3% | 0.225 Δ -62.6% | 0.357 Δ -29.2% |
| Gradient score | 17.58 Δ +28.1% | 0.304 Δ -42.0% | 0.438 Δ -10.1% | 9.42 Δ +30.2% | 0.296 Δ -68.3% | 0.702 Δ -7.5% | 25.00 Δ -2.2% | 0.817 Δ -8.6% | 0.667 Δ -4.5% | 11.06 Δ +49.3% | 0.261 Δ -56.7% | 0.370 Δ -26.7% |
| Mean gate weight | 16.52 Δ +20.3% | 0.371 Δ -29.3% | 0.451 Δ -7.4% | 9.21 Δ +27.4% | 0.375 Δ -59.8% | 0.712 Δ -6.3% | 24.09 Δ -5.8% | 0.836 Δ -6.5% | 0.669 Δ -4.2% | 10.28 Δ +38.7% | 0.323 Δ -46.3% | 0.392 Δ -22.4% |
| Routing frequency | 17.14 Δ +24.8% | 0.035 Δ -93.3% | 0.332 Δ -31.8% | 9.53 Δ +31.7% | 0.295 Δ -68.4% | 0.710 Δ -6.5% | 26.35 Δ +3.1% | 0.712 Δ -20.4% | 0.634 Δ -9.2% | 16.76 Δ +126.2% | 0.017 Δ -97.2% | 0.337 Δ -33.3% |
| Random | 14.23 Δ +3.6% | 0.515 Δ -1.9% | 0.480 Δ -1.4% | 7.46 Δ +3.1% | 0.930 Δ -0.4% | 0.753 Δ -0.9% | 22.14 Δ -13.4% | 0.876 Δ -2.0% | 0.673 Δ -3.6% | 8.61 Δ +16.2% | 0.457 Δ -24.1% | 0.439 Δ -13.1% |
| ***Bottom 10%: logit-mask (better preservation: smaller absolute change from checkpoint)*** | | | | | | | | | | | | |
| **CONTRAIL** (Ours) | 13.60 Δ -0.9% | 0.529 Δ +0.8% | 0.454 Δ -6.8% | 7.23 Δ -0.05% | 0.927 Δ -0.7% | 0.758 Δ -0.2% | 25.37 Δ -0.8% | 0.892 Δ -0.2% | 0.675 Δ -3.3% | 7.68 Δ +3.6% | 0.593 Δ -1.5% | 0.435 Δ -13.9% |
| Shapley-MoE | 14.49 Δ +5.6% | 0.459 Δ -12.5% | 0.448 Δ -8.0% | 7.30 Δ +0.9% | 0.924 Δ -1.0% | 0.753 Δ -0.9% | 26.45 Δ +3.4% | 0.865 Δ -3.2% | 0.669 Δ -4.2% | 8.16 Δ +10.1% | 0.487 Δ -19.1% | 0.429 Δ -15.1% |
| SHAPE | 14.79 Δ +7.7% | 0.430 Δ -18.0% | 0.445 Δ -8.6% | 7.40 Δ +2.3% | 0.918 Δ -1.6% | 0.747 Δ -1.7% | 26.77 Δ +4.7% | 0.853 Δ -4.6% | 0.666 Δ -4.6% | 8.38 Δ +13.2% | 0.437 Δ -27.4% | 0.426 Δ -15.6% |
| Gradient score | 14.94 Δ +8.8% | 0.416 Δ -20.8% | 0.444 Δ -8.9% | 7.46 Δ +3.2% | 0.914 Δ -2.1% | 0.743 Δ -2.2% | 26.92 Δ +5.3% | 0.847 Δ -5.3% | 0.665 Δ -4.8% | 8.50 Δ +14.7% | 0.412 Δ -31.6% | 0.425 Δ -15.9% |
| Mean gate weight | 15.20 Δ +10.7% | 0.391 Δ -25.6% | 0.441 Δ -9.4% | 7.50 Δ +3.7% | 0.912 Δ -2.3% | 0.740 Δ -2.5% | 27.20 Δ +6.4% | 0.836 Δ -6.5% | 0.662 Δ -5.2% | 8.70 Δ +17.3% | 0.368 Δ -38.8% | 0.422 Δ -16.4% |
| Routing frequency | 13.86 Δ +0.9% | 0.530 Δ +1.0% | 0.463 Δ -4.9% | 7.24 Δ +0.09% | 0.932 Δ -0.09% | 0.751 Δ -1.1% | 25.22 Δ -1.4% | 0.887 Δ -0.8% | 0.671 Δ -3.9% | 7.91 Δ +6.7% | 0.603 Δ +0.2% | 0.438 Δ -13.3% |
| Random | 15.72 Δ +14.5% | 0.340 Δ -35.2% | 0.441 Δ -9.4% | 7.89 Δ +9.1% | 0.885 Δ -5.2% | 0.720 Δ -5.2% | 27.76 Δ +8.6% | 0.814 Δ -8.9% | 0.658 Δ -5.7% | 9.09 Δ +22.7% | 0.281 Δ -53.3% | 0.456 Δ -9.7% |
| ***Bottom 20%: logit-mask (better preservation: smaller absolute change from checkpoint)*** | | | | | | | | | | | | |
| **CONTRAIL** (Ours) | 14.02 Δ +2.1% | 0.510 Δ -2.9% | 0.437 Δ -10.3% | 7.22 Δ -0.2% | 0.927 Δ -0.7% | 0.756 Δ -0.5% | 26.56 Δ +3.9% | 0.884 Δ -1.1% | 0.652 Δ -6.6% | 8.57 Δ +15.7% | 0.554 Δ -8.0% | 0.362 Δ -28.3% |
| Shapley-MoE | 15.15 Δ +10.4% | 0.403 Δ -23.2% | 0.411 Δ -15.6% | 7.39 Δ +2.2% | 0.914 Δ -2.0% | 0.746 Δ -1.8% | 28.52 Δ +11.5% | 0.831 Δ -7.1% | 0.629 Δ -10.0% | 9.64 Δ +30.1% | 0.426 Δ -29.3% | 0.356 Δ -29.5% |
| SHAPE | 15.69 Δ +14.2% | 0.353 Δ -32.8% | 0.398 Δ -18.2% | 7.61 Δ +5.2% | 0.895 Δ -4.1% | 0.734 Δ -3.4% | 29.45 Δ +15.2% | 0.806 Δ -9.8% | 0.617 Δ -11.5% | 10.14 Δ +36.8% | 0.366 Δ -39.3% | 0.353 Δ -30.1% |
| Gradient score | 15.95 Δ +16.2% | 0.328 Δ -37.5% | 0.392 Δ -19.4% | 7.75 Δ +7.1% | 0.883 Δ -5.4% | 0.726 Δ -4.4% | 29.91 Δ +17.0% | 0.794 Δ -11.2% | 0.612 Δ -12.3% | 10.39 Δ +40.2% | 0.335 Δ -44.3% | 0.352 Δ -30.4% |
| Mean gate weight | 16.42 Δ +19.6% | 0.284 Δ -45.9% | 0.382 Δ -21.7% | 7.83 Δ +8.3% | 0.876 Δ -6.1% | 0.721 Δ -5.0% | 30.71 Δ +20.1% | 0.772 Δ -13.7% | 0.602 Δ -13.7% | 10.83 Δ +46.2% | 0.283 Δ -53.1% | 0.349 Δ -30.9% |
| Routing frequency | 14.36 Δ +4.6% | 0.507 Δ -3.4% | 0.432 Δ -11.3% | 7.37 Δ +1.9% | 0.927 Δ -0.7% | 0.744 Δ -2.1% | 25.52 Δ -0.2% | 0.887 Δ -0.8% | 0.648 Δ -7.2% | 8.95 Δ +20.8% | 0.529 Δ -12.1% | 0.358 Δ -29.1% |
| Random | 17.35 Δ +26.4% | 0.196 Δ -62.7% | 0.360 Δ -26.1% | 8.62 Δ +19.2% | 0.800 Δ -14.3% | 0.680 Δ -10.5% | 32.33 Δ +26.4% | 0.728 Δ -18.6% | 0.583 Δ -16.5% | 11.71 Δ +58.0% | 0.177 Δ -70.6% | 0.376 Δ -25.5% |

intervention. A small bottom-removal effect therefore indicates that the router can replace those low-ranked computations with limited task damage, whereas a large effect shows that the ranking is not robust to rerouting compensation.

### E.2 MATCHED GSM8K QUALITATIVE CASES

Section D.9 describes the matched-effect selection procedure. We show the full displayed reasoning chain for each response. The only compression is for pathological repeated-token spans: the first few repeated tokens are shown, followed by an explicit repetition marker. Correct final answers are highlighted in green and incorrect or missing answers in red.

***Case 1: OLMoE, GSM8K idx 108.*** **Prompt.** Henry is making cookies for a local baking competition. He wants to make twice as many as he did last year. When he finishes baking, he realizes he actually baked 15 more cookies than he meant to. He drops 5 of his cookies as he is putting them out to cool, and now has a total of 110 cookies. How many cookies did Henry bake last year?

| **Base** | **Remove 40 rare / high-contribution** | **Remove 40 frequent / low-contribution** |
|---|---|---|
| Let's assume the number of cookies Henry baked last year is $x$.<br>He wants to make twice as many this year, so he will make $2x$ cookies.<br>However, he accidentally baked 15 more cookies than he meant to, so he actually baked $2x + 15$ cookies.<br>After dropping 5 cookies, he has a total of 110 cookies, so $2x + 15 - 5 = 110$.<br>Simplifying the equation, we get $2x + 10 = 110$.<br>Subtracting 10 from both sides, we get $2x = 100$.<br>Dividing both sides by 2, we get $x = 50$.<br>Therefore, Henry baked 50 cookies last year.<br>**✓ The answer is: 50** | Henry wants to make twice as many cookies as he did last year, so if he baked $x$ cookies last year, he will bake $2x$ cookies this year.<br>He also realizes he actually baked 15 more cookies than he meant to, so $2x - 15 = 110$.<br>To find the value of $x$, we add 15 to both sides: $2x - 15 + 15 = 110 + 15$.<br>Simplifying, $2x = 125$.<br>Dividing both sides by 2, $x = 62.5$.<br>So, Henry baked 62.5 cookies last year.<br>He dropped 5 cookies, so the total number of cookies he baked is $62.5 - 5 = 57.5$.<br>Since he wants to make twice as many cookies as last year, Henry actually baked $57.5 \times 2 = 115$ cookies last year.<br>**✗ The answer is: 115** | Let's assume Henry baked $x$ cookies last year.<br>He wants to make twice as many cookies this year, so he baked $2x$ cookies this year.<br>However, he accidentally baked 15 more cookies than he meant to, so he actually baked $2x + 15$ cookies.<br>After dropping 5 cookies, he has a total of 110 cookies, so $2x + 15 - 5 = 110$.<br>Simplifying the equation, we have $2x + 10 = 110$.<br>Subtracting 10 from both sides, we have $2x = 100$.<br>Dividing both sides by 2, we have $x = 50$.<br>Therefore, Henry baked 50 cookies last year.<br>**✓ The answer is: 50** |

***Case 2: Granite, GSM8K idx 117.*** **Prompt.** John buys 2 pairs of shoes for each of his 3 children. They cost $60 each. How much did he pay?

| **Base** | **Remove 40 rare / high-contribution** | **Remove 40 frequent / low-contribution** |
|---|---|---|
| John has 3 children, and he buys 2 pairs of shoes for each child. So, he buys a total of $3 \times 2 = 6$ pairs of shoes.<br>Each pair of shoes costs $60, so the total cost for 6 pairs is $6 \times \$60 = \$360$.<br>Therefore, John paid $360 for the shoes.<br>**✓ Answer: $360** | John has 3 children and buys 2 pairs of shoes for each, "so so so so so . . ."<br>The generation then continues with repeated "so" and "at" tokens: "at at at at at . . ." *[repeats until termination].*<br>**✗ No valid answer** | John has 3 children, and he buys 2 pairs of shoes for each child.<br>So, the total number of pairs of shoes he buys is $3 \times 2 = 6$ pairs.<br>Each pair of shoes costs $60.<br>To find the total cost, multiply the number of pairs by the cost per pair: $6 \times \$60 = \$360$.<br>**✓ John paid $360 for the shoes.** |

***Case 3: Granite, GSM8K idx 220.*** **Prompt.** It takes Carmen 10 minutes to finish a crossword puzzle and 5 minutes to finish a sudoku puzzle. Over the weekend she solved 3 crossword puzzles and 8 sudoku puzzles. How much time did she spend playing these games?

| **Base** | **Remove 40 rare / high-contribution** | **Remove 40 frequent / low-contribution** |
|---|---|---|
| Carmen spends 10 minutes to finish one crossword puzzle and 5 minutes to finish one sudoku puzzle.<br>For the crossword puzzles, she solved 3, so the total time is $3 \times 10 = 30$ minutes.<br>For the sudoku puzzles, she solved 8, so the total time is $8 \times 5 = 40$ minutes.<br>In total, Carmen spent $30 + 40 = 70$ minutes playing these games.<br>So, Carmen spent 70 minutes playing crossword puzzles and sudoku puzzles over the weekend.<br>**✓ Final answer: 70 minutes** | Carmen spent 10 minutes per crossword puzzle and solved 3 of them, "so so so so so . . . " *[repeats]*, that she spent:<br>3 crossword puzzles $\times 10$ minutes per puzzle $= 30$ minutes.<br>For the sudoku puzzles, she spent 5 minutes per puzzle and solved 8 of them, "so so so so so . . . " *[repeats]*, that she spent:<br>8 sudoku puzzles $\times 5$ minutes per puzzle $= 40$ minutes.<br>To find the total time spent, add the time for crossword puzzles and sudoku puzzles: $30 + 40 = 70$ minutes.<br>Carmen spent a total of 70 minutes playing these games.<br>**✓ Answer: 70 minutes**<br>*Correct final answer, but visibly degraded reasoning.* | Carmen spent $10 \times 3 = 30$ minutes on crossword puzzles.<br>She also spent $5 \times 8 = 40$ minutes on sudoku puzzles.<br>In total, Carmen spent $30 + 40 = 70$ minutes playing these games.<br>**✓ The answer is: 70** |

***Case 4: gpt-oss, GSM8K idx 147.*** **Prompt.** Debra sees 30 bees leave the hive in the first 6 hours, half that many return in the next 6 hours, and twice the original number leave in the third 6 hours. In the last 6 hours, every bee still outside returns. How many bees return in the final block?

| **Base** | **Remove 40 rare / high-contribution** | **Remove 40 frequent / low-contribution** |
|---|---|---|
| Let's break the day into the four 6-hour blocks Debra watches.<br>1st block: 30 bees leave the hive.<br>2nd block: half of the bees that left in the first block return, so $\frac{1}{2} \times 30 = 15$ bees return.<br>3rd block: twice as many bees leave as in the first block, so $2 \times 30 = 60$ bees leave.<br>4th block: every bee that had left before this block and had not yet returned comes back.<br>Now count the bees still outside: from the first block, $30 - 15 = 15$ remain; from the third block, all 60 remain.<br>Thus, $15 + 60 = 75$ bees return in the last 6 hours.<br>**✓ Answer: 75** | *Reasoning.* First 6h: 30 leave. Next 6h: half that many return, so 15 return. Then twice as many as first leave, so $2 \times 30 = 60$ leave in the third 6h.<br>The response then asks which bees "left before that" and considers only the first 30. Of those, $30 - 15 = 15$ had not returned, so it concludes that 15 return in the last 6h.<br>**✗ Answer: 15**<br>*Trailing repeated special tokens are omitted.* | Let's keep track of the bees in each 6-hour block.<br>1st (0–6 h): 30 leave.<br>2nd (6–12 h): half of 30 return, so 15 return.<br>3rd (12–18 h): $2 \times 30 = 60$ leave.<br>Step 1 – bees still outside after the first three blocks: from the first block, $30 - 15 = 15$ remain; from the third block, all 60 remain.<br>Total still outside: $15 + 60 = 75$.<br>Step 2 – the problem states that every bee that left before and had not already returned comes back in the final block. Thus all 75 return.<br>**✓ Answer: 75** |

***Case 5: Qwen3, GSM8K idx 125.*** **Prompt.** Julia buys a new package of spoons. Her husband gives her 5 more spoons, she uses 3 to sample her stew, and she later has 12 spoons. How many spoons were in Julia's package?

| **Base** | **Remove 40 rare / high-contribution** | **Remove 40 frequent / low-contribution** |
|---|---|---|
| Let's break down the problem step by step.<br>1. Julia bought a package of spoons; call the number in that package $x$.<br>2. Her husband bought 5 new spoons and gave them to her, so before using any she had $x + 5$.<br>3. She used 3 spoons, leaving $x + 5 - 3 = x + 2$.<br>4. When she went to set the table, she had 12 spoons, so $x + 2 = 12$.<br>5. Solving gives $x = 12 - 2 = 10$.<br>✓ **Answer: 10 spoons**<br>**Final answer:** $\boxed{10}$. | Let's break down the problem step by step.<br>Julia bought a package of spoons; call its size $x$.<br>Her husband bought 5 new spoons and gave them to her, so the total before using any was $x + 5$.<br>Julia used 3 spoons, leaving $(x + 5) - 3 = x + 2$.<br>Later she had a total of 12 spoons, so the number after using 3 was 12. Therefore $x + 2 = 12$.<br>Solving gives $x = 12 - 2 = 10$.<br>✓ **Answer: 10 spoons**<br>**Check:** Julia bought 10; her husband added 5 to make 15; she used 3 and had 12 left. Correct.<br>† *Raw evaluator metadata records* `pred=12`*, although the displayed reasoning and answer are 10.* | We are given the following information: Julia bought a package of $x$ spoons; her husband added 5; Julia used 3; and 12 spoons remained.<br>We need to find how many spoons were in Julia's original package.<br>Step 1: Julia starts with $x$ spoons.<br>Step 2: her husband adds 5, giving $x + 5$.<br>Step 3: she uses 3, leaving $x + 5 - 3 = x + 2$.<br>Step 4: she has 12 left, so $x + 2 = 12$.<br>Solving gives $x = 12 - 2 = 10$.<br>✓ **Answer: Julia bought a package of 10 spoons.**<br>**Verification:** $10 + 5 = 15$, and $15 - 3 = 12$, matching the problem.<br>**Final answer:** $\boxed{10}$. |

Across the cleanly scored examples, removing the rare/high-contribution set can flip a correct arithmetic solution or destabilize the generation, whereas the matched frequent/low-contribution deletion preserves the reasoning and final answer. The two Granite cases also separate answer correctness from reasoning quality: one rare/high-contribution deletion collapses completely, while the other remains numerically correct but develops strong repetition. The Qwen3 example is retained as a logging discrepancy rather than answer-flip evidence, because its displayed rare/high-contribution response is correct even though the raw evaluator metadata records a different prediction. These cases qualitatively complement the aggregate RQ1 intervention results.

### E.3 RQ2: TASK-SPECIFIC CROSS-TASK MATRICES AND HEATMAPS

Table 12: Top 1% task-specific deletion matrix. Entries are $\Delta\mathrm{ACC}_{\mathrm{damage}}$; rows are evaluation tasks and columns are the task used to select experts.

| Model | Eval task | Select MMLU | Select GSM8K | Select MedQA |
|---|---|---|---|---|
| OLMoE | MMLU | **0.0821** | 0.0103 | 0.0382 |
| | GSM8K | 0.1016 | **0.3844** | 0.1948 |
| | MedQA | 0.0471 | -0.0008 | 0.0338 |
| Qwen3 | MMLU | **0.0280** | 0.0100 | 0.0210 |
| | GSM8K | 0.0610 | **0.1403** | 0.0740 |
| | MedQA | 0.0500 | 0.0180 | **0.0660** |
| gpt-oss | MMLU | 0.0115 | 0.0313 | **0.0318** |
| | GSM8K | -0.0030 | **0.0159** | 0.0099 |
| | MedQA | -0.0024 | 0.0157 | **0.0503** |
| Granite | MMLU | **0.1023** | 0.0575 | 0.0866 |
| | GSM8K | 0.2009 | **0.5785** | 0.1964 |
| | MedQA | 0.0518 | 0.0463 | **0.0660** |

The full cross-task matrices in Tables 12 and 13 show that transfer between task-conditioned rankings is structured and asymmetric. GSM8K has the clearest diagonal: the ranking estimated from

Table 13: Top 3% task-specific deletion matrix. Same convention as Table 12.

| Model | Eval task | Select MMLU | Select GSM8K | Select MedQA |
|---|---|---|---|---|
| OLMoE | MMLU | **0.1030** | 0.0205 | 0.0836 |
| | GSM8K | 0.3260 | **0.4890** | 0.2464 |
| | MedQA | **0.0518** | -0.0008 | 0.0479 |
| Qwen3 | MMLU | **0.0710** | 0.0400 | 0.0560 |
| | GSM8K | 0.3650 | **0.8734** | 0.2950 |
| | MedQA | 0.0840 | 0.0340 | **0.1100** |
| gpt-oss | MMLU | 0.0388 | **0.0775** | 0.0617 |
| | GSM8K | 0.0842 | **0.1509** | 0.0614 |
| | MedQA | 0.0141 | **0.0597** | 0.0299 |
| Granite | MMLU | **0.2228** | 0.1688 | 0.1716 |
| | GSM8K | 0.4693 | **0.5906** | 0.4109 |
| | MedQA | 0.0841 | **0.1579** | 0.0903 |

mathematical reasoning repeatedly causes substantially more GSM8K damage than rankings imported from MMLU or MedQA. MMLU and MedQA transfer more strongly to each other, consistent with a larger shared factual-knowledge component. The few off-diagonal cells that exceed the diagonal are also informative: expert specialization is overlapping rather than a partition into task-exclusive modules, and the causal effect of transferring a ranking from task $A$ to task $B$ need not equal the reverse transfer because the selected sets and their marginal utilities differ. The persistence of these patterns from Top 1% to Top 3% shows that task conditioning changes a region of the expert ranking, not just its single highest-valued element.

The heatmaps provide the depth-resolved counterpart to the intervention matrices. In Figure 3, task changes alter both the experts that receive high value and the layers in which high-value mass is concentrated. The individual/self maps in Figure 4 retain much of the broad support, while the full values redistribute weight inside that support according to co-routing. This separation is consistent with the decomposition used by **CONTRAIL**: individual terms locate experts with strong standalone effects, whereas pair terms change the relative ordering of experts whose utility depends on the partners with which they are executed.

For reproducibility, we define the signed cell-level similarity used below. Let $\Phi^A_{\lambda e}$ denote the corpus contribution $\Phi^{\text{corpus}}_e$ from Equation (75) for layer $\lambda$, expert $e$, and task $A$. Within one backbone and task, let $n_A$ be the number of slots with nonzero $|\Phi^A_{\lambda e}|$, and let $\text{rank}_\uparrow(|\Phi^A_{\lambda e}|)$ be its average rank in ascending magnitude order (smallest rank 1, largest rank $n_A$). Define

$$r^A_{\lambda e} := \frac{\text{rank}_\uparrow(|\Phi^A_{\lambda e}|)}{n_A} \in (0, 1], \qquad \widetilde{\Phi}^A_{\lambda e} := \text{sign}(\Phi^A_{\lambda e}) r^A_{\lambda e}. \tag{91}$$

For nonzero slots this applies when $n_A > 0$; if $n_A = 0$, or for any individual zero slot, set $\widetilde{\Phi}^A_{\lambda e} = 0$. For two tasks $A, B$, define

$$\text{sim}_{\lambda e}(A, B) := \begin{cases} \text{sign}\Big(\widetilde{\Phi}^A_{\lambda e}\widetilde{\Phi}^B_{\lambda e}\Big) \dfrac{\min\{|\widetilde{\Phi}^A_{\lambda e}|, |\widetilde{\Phi}^B_{\lambda e}|\}}{\max\{|\widetilde{\Phi}^A_{\lambda e}|, |\widetilde{\Phi}^B_{\lambda e}|\}}, & \widetilde{\Phi}^A_{\lambda e}\widetilde{\Phi}^B_{\lambda e} \neq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{92}$$

Slots inactive in both tasks are marked separately; all other zero cases are shown as zero similarity. The reported positive-agreement rate is the fraction with $\text{sim}_{\lambda e}(A, B) > 0$ among slots that are nonzero in both tasks. Average ranks make the normalization well-defined under ties.

Figure 5 provides the continuous, cell-level counterpart to the discrete top-expert overlaps in Table 3. Across the four backbones, MMLU and MedQA exhibit dominant co-directional alignment (70.2%–77.3% positive agreement), consistent with the two knowledge tasks recruiting broadly aligned functional structure across depth. Pairing GSM8K with either knowledge task yields a more balanced positive–negative split (42.0%–55.8% positive agreement), consistent with weaker functional alignment and the observed cross-task transfer degradation. The recurrence of this pattern across architectures with 32–128 experts per layer indicates that it is not confined to a single model scale or routing topology.

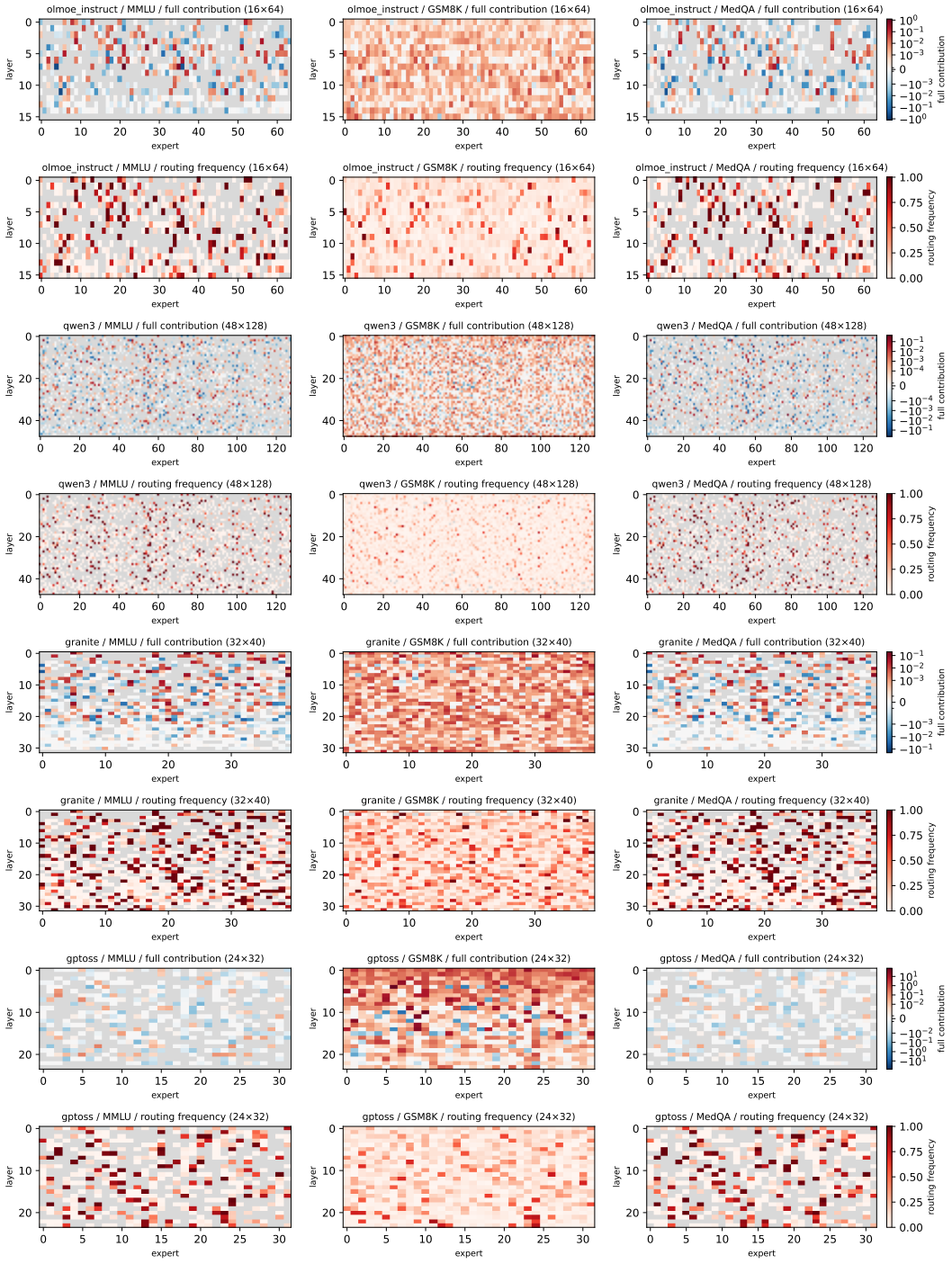


Figure 3: Full-value heatmaps. Additional layer-by-expert visualizations of the full expert value across tasks and backbones.

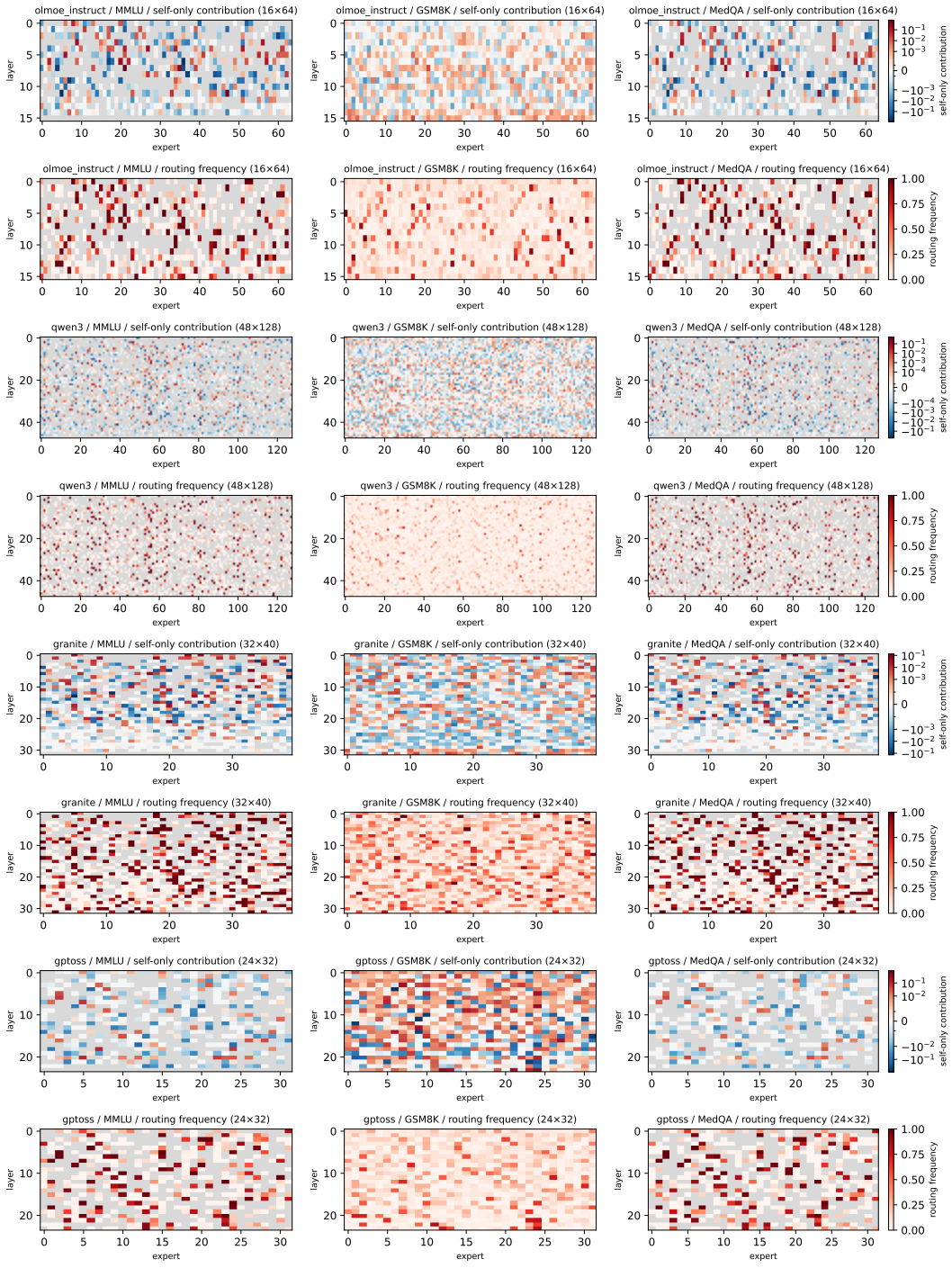


Figure 4: Individual/self-value heatmaps. The same layer-by-expert layout without the full co-routed interaction contribution.

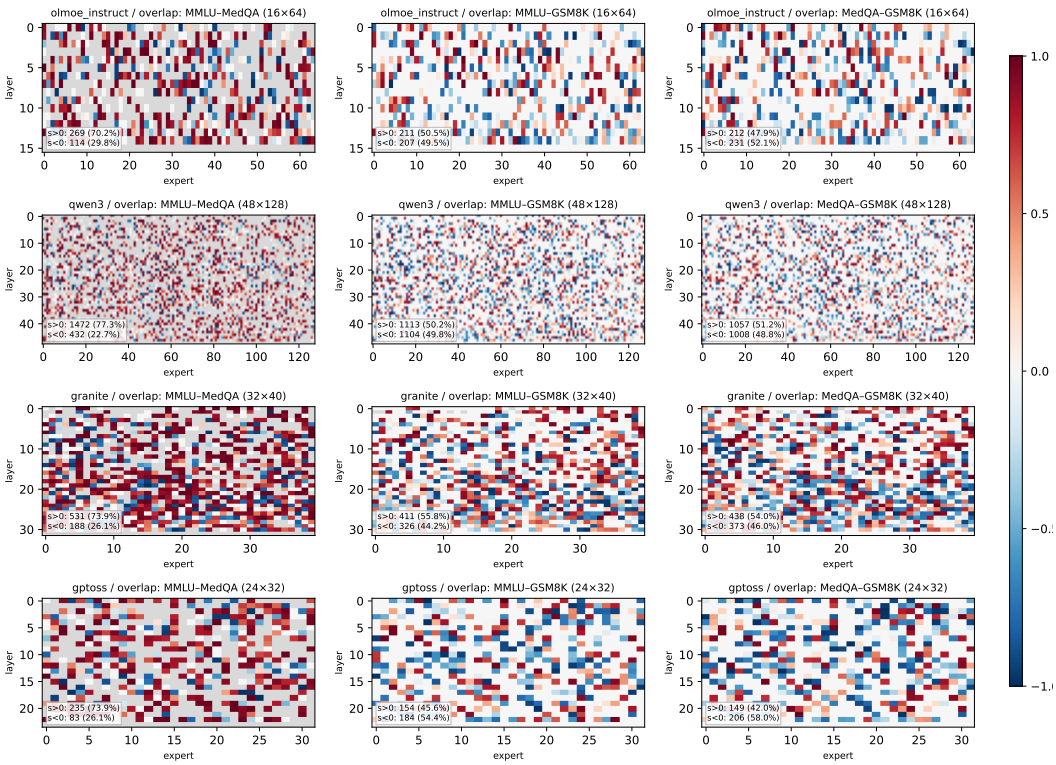


Figure 5: **Pairwise cell-level expert contribution similarity across tasks.** Each heatmap displays $\mathrm{sim}_{\lambda e}(A, B)$ from Equation (92) for every layer–expert slot: MMLU versus MedQA (left), MMLU versus GSM8K (center), and MedQA versus GSM8K (right), across four MoE backbones. **Red** indicates co-directional contribution ($\mathrm{sim} > 0$), **blue** indicates sign conflict ($\mathrm{sim} < 0$), **white** denotes one side zero or inactive, and **gray** denotes inactivity in both tasks.

### E.4 RQ3: ROUTER ATTRIBUTION ANALYSIS

The core matched intervention grid is reported in Table 4. The comparison should be read together with the estimator definitions in Section D.5. The hard-route and STE baselines differ in the backward signal assigned to the same factual boundary pair, whereas CCGE-linear uses the margin-normalized, mass-conserving boundary derivative and full CCGE additionally scores the curvature of the induced output displacement. Because all methods select three layers and are evaluated with the same finite nearest-boundary one-swap endpoint, the 10/12 first-place and 2/12 second-place pattern of full CCGE reflects layer-ranking fidelity rather than a stronger intervention being applied to **CONTRAIL**.

The four second-best/one best placements of CCGE-linear are also informative: boundary alignment already captures a large part of the router signal, while the second-order correction changes the ordering when the downstream loss geometry makes the same output displacement more or less consequential. This is the router-side analogue of the expert decomposition: the finite counterfactual is fixed first, and the local estimator determines how faithfully that counterfactual is ranked from factual-state information.

### E.5 RQ5: SPARSE FINE-TUNING GAINS OVER THE UNMODIFIED CHECKPOINT

The main text compares Routing Frequency and **CONTRAIL** at the 3% and 10% trainable-expert budgets. Here we expand the grid to 1%/3%/6%/10% and add mean gate weight, gradient score, SHAPE, and Shapley-MoE. We also report a direction-normalized summary: relative PPL reduction for WT-2 and relative accuracy increase for GSM8K/MMLU, averaged across the three tasks within each backbone.

The expanded grid separates two factors that are conflated by a single fine-tuning budget. Holding the selector fixed while increasing the trainable fraction traces a *capacity curve*: it measures how much adaptation improves when the optimizer is allowed to modify a larger expert subspace. Holding the budget fixed while changing the selector isolates a *selection effect*: it asks how productively that same parameter budget is allocated. This distinction is especially important in sparse MoEs because a "1% budget" does not merely reduce the number of trainable parameters; it restricts learning to a small set of expert functions that are exposed only on the tokens routed to them.

At small budgets, the identity of those functions therefore determines the available update directions. Usage-based selectors favor experts that are frequently executed, and gate-based selectors refine this with allocation magnitude. Gradient and perturbation scores introduce task-local sensitivity. Coalition-aware scores additionally value an expert conditional on the routed computation around it, so they can prioritize experts whose outputs are not only active but consequential in the factual forward pass. As the budget grows, selectors expose increasingly overlapping expert supports and the optimizer has more alternative directions through which to realize a task update. The expected signature is consequently a strong selector gap under scarce capacity that narrows as trainable support expands.

This experiment also connects intervention and adaptation at the level of the parameter subspace. In RQ1, deletion asks which executed expert computations carry loss-relevant utility. Here all non-selected modules are frozen, so the ranking determines where task-specific parameter movement is permitted. A ranking that succeeds in both settings is identifying expert locations that are important in the current computation and high-leverage sites for changing that computation.

The absolute grid in Table 14 follows this capacity–selection decomposition. For every selector, increasing the trainable fraction improves all three task metrics. At a fixed fraction, stronger contribution-aware selectors obtain more improvement from the same number of trainable expert FFNs, and **CONTRAIL** attains the best absolute result in every reported model–task–budget block. Thus the gain cannot be attributed only to having more editable parameters: at matched capacity, the location of those parameters materially changes the attainable update.

Table 14: Complete RQ5 sparse fine-tuning comparison. Absolute post-training performance for all expert selectors under matched Top 1%/3%/6%/10% trainable-expert budgets. All improvements are measured against the same unmodified checkpoint.

| | OLMoE | | | Qwen3 | | | gpt-oss | | | Granite | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Selection method | WT-2 PPL↓ | GSM8K ACC↑ | MMLU ACC↑ | WT-2 PPL↓ | GSM8K ACC↑ | MMLU ACC↑ | WT-2 PPL↓ | GSM8K ACC↑ | MMLU ACC↑ | WT-2 PPL↓ | GSM8K ACC↑ | MMLU ACC↑ |
| ***Unmodified checkpoint*** | | | | | | | | | | | | |
| Unmodified checkpoint | 13.73 | 0.525 | 0.487 | 7.23 | 0.933 | 0.760 | 25.57 | 0.894 | 0.698 | 7.41 | 0.602 | 0.505 |
| ***Top 1% trainable experts*** | | | | | | | | | | | | |
| Routing frequency | 12.95 Δ -5.7% | 0.562 Δ +7.1% | 0.510 Δ +4.7% | 6.62 Δ -8.4% | 0.949 Δ +1.7% | 0.795 Δ +4.6% | 24.30 Δ -5.0% | 0.907 Δ +1.5% | 0.717 Δ +2.8% | 6.94 Δ -6.4% | 0.639 Δ +6.2% | 0.531 Δ +5.2% |
| Mean gate weight | 12.88 Δ -6.2% | 0.566 Δ +7.7% | 0.512 Δ +5.1% | 6.57 Δ -9.2% | 0.951 Δ +1.9% | 0.797 Δ +4.9% | 24.18 Δ -5.4% | 0.909 Δ +1.6% | 0.719 Δ +3.0% | 6.90 Δ -6.9% | 0.642 Δ +6.7% | 0.533 Δ +5.6% |
| Gradient score | 12.79 Δ -6.8% | 0.570 Δ +8.6% | 0.514 Δ +5.6% | 6.52 Δ -9.9% | 0.952 Δ +2.0% | 0.801 Δ +5.4% | 24.02 Δ -6.1% | 0.910 Δ +1.8% | 0.721 Δ +3.4% | 6.85 Δ -7.6% | 0.646 Δ +7.4% | 0.536 Δ +6.2% |
| SHAPE | 12.70 Δ -7.5% | 0.574 Δ +9.4% | 0.517 Δ +6.1% | 6.47 Δ -10.5% | 0.953 Δ +2.1% | 0.803 Δ +5.7% | 23.86 Δ -6.7% | 0.912 Δ +2.0% | 0.724 Δ +3.7% | 6.79 Δ -8.3% | 0.650 Δ +8.0% | 0.539 Δ +6.8% |
| Shapley-MoE | 12.62 Δ -8.1% | 0.579 Δ +10.2% | 0.520 Δ +6.7% | 6.42 Δ -11.3% | 0.955 Δ +2.3% | 0.806 Δ +6.1% | 23.70 Δ -7.3% | 0.914 Δ +2.2% | 0.726 Δ +4.0% | 6.74 Δ -9.0% | 0.655 Δ +8.7% | 0.542 Δ +7.4% |
| **CONTRAIL** (Ours) | 12.50 Δ -8.9% | 0.584 Δ +11.2% | 0.523 Δ +7.3% | 6.35 Δ -12.3% | 0.957 Δ +2.5% | 0.811 Δ +6.7% | 23.51 Δ -8.1% | 0.916 Δ +2.4% | 0.729 Δ +4.5% | 6.65 Δ -10.3% | 0.662 Δ +9.9% | 0.547 Δ +8.4% |
| ***Top 3% trainable experts*** | | | | | | | | | | | | |
| Routing frequency | 12.50 Δ -8.9% | 0.584 Δ +11.2% | 0.523 Δ +7.3% | 6.36 Δ -12.0% | 0.956 Δ +2.4% | 0.810 Δ +6.6% | 23.51 Δ -8.1% | 0.916 Δ +2.4% | 0.729 Δ +4.5% | 6.68 Δ -9.9% | 0.660 Δ +9.6% | 0.546 Δ +8.1% |
| Mean gate weight | 12.41 Δ -9.6% | 0.588 Δ +12.0% | 0.525 Δ +7.9% | 6.32 Δ -12.6% | 0.957 Δ +2.6% | 0.812 Δ +6.9% | 23.35 Δ -8.7% | 0.918 Δ +2.6% | 0.732 Δ +4.8% | 6.62 Δ -10.6% | 0.664 Δ +10.3% | 0.549 Δ +8.7% |
| Gradient score | 12.33 Δ -10.2% | 0.592 Δ +12.8% | 0.528 Δ +8.4% | 6.26 Δ -13.5% | 0.959 Δ +2.8% | 0.815 Δ +7.3% | 23.19 Δ -9.3% | 0.919 Δ +2.8% | 0.734 Δ +5.2% | 6.57 Δ -11.3% | 0.668 Δ +11.0% | 0.552 Δ +9.3% |
| SHAPE | 12.21 Δ -11.0% | 0.598 Δ +13.9% | 0.531 Δ +9.1% | 6.19 Δ -14.4% | 0.961 Δ +2.9% | 0.819 Δ +7.8% | 22.99 Δ -10.1% | 0.921 Δ +3.1% | 0.737 Δ +5.6% | 6.51 Δ -12.2% | 0.673 Δ +11.8% | 0.555 Δ +10.0% |
| Shapley-MoE | 12.10 Δ -11.9% | 0.603 Δ +14.9% | 0.534 Δ +9.7% | 6.14 Δ -15.1% | 0.962 Δ +3.1% | 0.822 Δ +8.2% | 22.79 Δ -10.9% | 0.923 Δ +3.3% | 0.740 Δ +6.0% | 6.44 Δ -13.1% | 0.678 Δ +12.7% | 0.559 Δ +10.7% |
| **CONTRAIL** (Ours) | 11.99 Δ -12.7% | 0.608 Δ +15.9% | 0.538 Δ +10.4% | 6.09 Δ -15.9% | 0.963 Δ +3.2% | 0.825 Δ +8.7% | 22.59 Δ -11.6% | 0.925 Δ +3.5% | 0.743 Δ +6.4% | 6.36 Δ -14.1% | 0.684 Δ +13.7% | 0.563 Δ +11.6% |
| ***Top 6% trainable experts*** | | | | | | | | | | | | |
| Routing frequency | 12.12 Δ -11.7% | 0.602 Δ +14.7% | 0.534 Δ +9.6% | 6.17 Δ -14.7% | 0.961 Δ +3.0% | 0.821 Δ +8.1% | 22.79 Δ -10.9% | 0.923 Δ +3.3% | 0.740 Δ +6.0% | 6.44 Δ -13.1% | 0.678 Δ +12.7% | 0.559 Δ +10.7% |
| Mean gate weight | 12.04 Δ -12.3% | 0.606 Δ +15.5% | 0.536 Δ +10.1% | 6.12 Δ -15.4% | 0.962 Δ +3.1% | 0.823 Δ +8.4% | 22.63 Δ -11.5% | 0.925 Δ +3.5% | 0.742 Δ +6.4% | 6.39 Δ -13.8% | 0.682 Δ +13.3% | 0.562 Δ +11.3% |
| Gradient score | 11.95 Δ -13.0% | 0.611 Δ +16.3% | 0.539 Δ +10.7% | 6.09 Δ -15.9% | 0.963 Δ +3.2% | 0.825 Δ +8.7% | 22.47 Δ -12.1% | 0.927 Δ +3.7% | 0.745 Δ +6.7% | 6.35 Δ -14.3% | 0.685 Δ +13.9% | 0.564 Δ +11.7% |
| SHAPE | 11.86 Δ -13.6% | 0.615 Δ +17.1% | 0.542 Δ +11.2% | 6.05 Δ -16.4% | 0.964 Δ +3.3% | 0.828 Δ +9.0% | 22.31 Δ -12.7% | 0.928 Δ +3.9% | 0.747 Δ +7.0% | 6.30 Δ -15.0% | 0.690 Δ +14.5% | 0.567 Δ +12.3% |
| Shapley-MoE | 11.77 Δ -14.3% | 0.619 Δ +17.9% | 0.544 Δ +11.7% | 6.01 Δ -16.9% | 0.965 Δ +3.4% | 0.829 Δ +9.1% | 22.16 Δ -13.4% | 0.930 Δ +4.0% | 0.750 Δ +7.4% | 6.24 Δ -15.7% | 0.694 Δ +15.2% | 0.570 Δ +12.9% |
| **CONTRAIL** (Ours) | 11.68 Δ -14.9% | 0.623 Δ +18.8% | 0.547 Δ +12.3% | 5.99 Δ -17.2% | 0.966 Δ +3.5% | 0.831 Δ +9.4% | 22.00 Δ -14.0% | 0.932 Δ +4.2% | 0.752 Δ +7.7% | 6.19 Δ -16.4% | 0.698 Δ +15.9% | 0.573 Δ +13.4% |
| ***Top 10% trainable experts*** | | | | | | | | | | | | |
| Routing frequency | 11.83 Δ -13.8% | 0.616 Δ +17.3% | 0.542 Δ +11.3% | 6.08 Δ -16.0% | 0.964 Δ +3.3% | 0.825 Δ +8.7% | 22.27 Δ -12.9% | 0.929 Δ +3.9% | 0.748 Δ +7.1% | 6.28 Δ -15.2% | 0.691 Δ +14.7% | 0.568 Δ +12.4% |
| Mean gate weight | 11.79 Δ -14.1% | 0.618 Δ +17.7% | 0.544 Δ +11.6% | 6.05 Δ -16.4% | 0.964 Δ +3.3% | 0.828 Δ +9.0% | 22.16 Δ -13.4% | 0.930 Δ +4.0% | 0.750 Δ +7.4% | 6.26 Δ -15.6% | 0.693 Δ +15.1% | 0.569 Δ +12.7% |
| Gradient score | 11.72 Δ -14.6% | 0.621 Δ +18.3% | 0.546 Δ +12.0% | 6.02 Δ -16.7% | 0.965 Δ +3.4% | 0.829 Δ +9.1% | 22.04 Δ -13.8% | 0.931 Δ +4.2% | 0.751 Δ +7.7% | 6.22 Δ -16.1% | 0.696 Δ +15.6% | 0.571 Δ +13.2% |
| SHAPE | 11.66 Δ -15.1% | 0.625 Δ +19.0% | 0.547 Δ +12.4% | 6.00 Δ -17.1% | 0.966 Δ +3.5% | 0.830 Δ +9.3% | 21.92 Δ -14.3% | 0.933 Δ +4.3% | 0.753 Δ +7.9% | 6.18 Δ -16.6% | 0.699 Δ +16.1% | 0.574 Δ +13.6% |
| Shapley-MoE | 11.61 Δ -15.4% | 0.627 Δ +19.4% | 0.549 Δ +12.7% | 5.99 Δ -17.2% | 0.966 Δ +3.5% | 0.831 Δ +9.4% | 21.84 Δ -14.6% | 0.933 Δ +4.4% | 0.754 Δ +8.1% | 6.15 Δ -17.0% | 0.701 Δ +16.4% | 0.575 Δ +13.9% |
| **CONTRAIL** (Ours) | 11.54 Δ -15.9% | 0.630 Δ +20.0% | 0.551 Δ +13.1% | 5.96 Δ -17.6% | 0.967 Δ +3.6% | 0.832 Δ +9.5% | 21.72 Δ -15.1% | 0.935 Δ +4.6% | 0.756 Δ +8.3% | 6.11 Δ -17.5% | 0.704 Δ +16.9% | 0.577 Δ +14.3% |

Table 15 makes the budget dependence of this selection effect easier to see. The gap between **CONTRAIL** and the strongest competing selector is largest at the smallest budgets and generally contracts as the trainable fraction grows. Mechanistically, expert choice is most consequential when the optimizer can modify only a narrow support; once many experts become trainable, different selectors increasingly overlap in the functional subspace available for adaptation. Together with RQ1, this gives the attribution score a symmetric operational interpretation: high-value experts are locations where removing executed computation is costly and where allocating scarce update capacity is especially effective.

Table 15: Average sparse fine-tuning gain over the unmodified checkpoint (%). Each cell averages WT-2 relative PPL reduction and GSM8K/MMLU relative accuracy increase for one backbone. Higher is better; all methods use the same layer-wise trainable-expert budget.

| | **OLMoE** | | | | **Qwen3** | | | | **gpt-oss** | | | | **Granite** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Selection method | 1% | 3% | 6% | 10% | 1% | 3% | 6% | 10% | 1% | 3% | 6% | 10% | 1% | 3% | 6% | 10% |
| Random selection | 2.0 | 4.6 | 7.5 | 10.0 | 1.7 | 3.2 | 5.2 | 6.8 | 1.0 | 2.4 | 4.0 | 5.6 | 2.0 | 4.4 | 7.5 | 10.2 |
| Routing frequency | 5.8 | 9.1 | 12.0 | 14.1 | 4.9 | 7.0 | 8.6 | 9.3 | 3.1 | 5.0 | 6.7 | 8.0 | 5.9 | 9.2 | 12.2 | 14.1 |
| Mean gate weight | 6.3 | 9.8 | 12.6 | 14.5 | 5.3 | 7.4 | 8.9 | 9.5 | 3.3 | 5.4 | 7.1 | 8.3 | 6.4 | 9.9 | 12.8 | 14.5 |
| Gradient score | 7.0 | 10.5 | 13.3 | 15.0 | 5.8 | 7.9 | 9.2 | 9.7 | 3.8 | 5.8 | 7.5 | 8.6 | 7.1 | 10.5 | 13.3 | 15.0 |
| SHAPE | 7.7 | 11.3 | 14.0 | 15.5 | 6.1 | 8.4 | 9.5 | 9.9 | 4.1 | 6.3 | 7.9 | 8.8 | 7.7 | 11.3 | 13.9 | 15.4 |
| Shapley-MoE | 8.3 | 12.2 | 14.6 | 15.8 | 6.6 | 8.8 | 9.8 | 10.0 | 4.5 | 6.7 | 8.3 | 9.0 | 8.4 | 12.2 | 14.6 | 15.8 |
| **CONTRAIL** (Ours) | **9.1** | **13.0** | **15.3** | **16.3** | **7.2** | **9.2** | **10.0** | **10.2** | **5.0** | **7.2** | **8.6** | **9.3** | **9.5** | **13.1** | **15.2** | **16.2** |

### E.6 SUPPORTING USAGE–VALUE AND STABILITY DIAGNOSTICS

Table 16: Usage–value and probe stability. Per-probe correlation, rare high-value share, and the first prefix reaching Kendall $\tau \geq 0.9$.

| Model | Probe | Spearman $r_s$ | Rare high-value | $n_{0.9}$ direct | $n_{0.9}$ full |
|---|---|---|---|---|---|
| OLMoE | WT-2 val | +0.276 | 4.10% | 253 | 253 |
| | GSM8K train | +0.255 | 3.52% | 512 | 512 |
| | MMLU val | +0.159 | 3.22% | 128 | 128 |
| Qwen3 | WT-2 val | +0.272 | 5.34% | 256 | 384 |
| | GSM8K train | +0.224 | 3.58% | 512 | 1024 |
| | MMLU val | +0.065 | 4.60% | 256 | 512 |
| gpt-oss | WT-2 val | +0.214 | 2.22% | 256 | 256 |
| | GSM8K train | +0.311 | 2.08% | 256 | 1024 |
| | MMLU val | -0.082 | 5.23% | 512 | 512 |
| Granite | WT-2 val | -0.074 | 4.49% | 256 | 256 |
| | GSM8K train | +0.166 | 4.25% | 512 | 512 |
| | MMLU val | +0.158 | 3.93% | 128 | 128 |

Across the 12 probes in Table 16, the usage–contribution Spearman correlation ranges from mildly negative to modestly positive, while 2.08%–5.34% of experts fall in the rare/high-value region. This quantifies the structural mismatch behind RQ1: routing frequency provides an occupancy measure, whereas conditional contribution can remain high for an expert that is activated infrequently but performs high-leverage computation on those activations.

The stability columns also reveal the extra sample requirement introduced by interaction estimation. Here "direct" denotes the first-order score $-\boldsymbol{u}_\star^\top \boldsymbol{\delta}_e$, whereas "full" denotes the complete interaction-aware score in Equation (22). Direct rankings typically reach $\tau \geq 0.9$ with 128–512 probe examples. The full score sometimes needs the same prefix, but on Qwen3 and gpt-oss GSM8K it requires 1024 examples because the interaction component is observed only on co-routed tokens. Thus the relevant sample size is controlled by both token count and co-routing coverage. The trajectories in Figure 6 make this distinction visible: the direct ranking often stabilizes first when pair support is sparse, after which the interaction-aware ranking converges as repeated co-routed pairs accumulate.

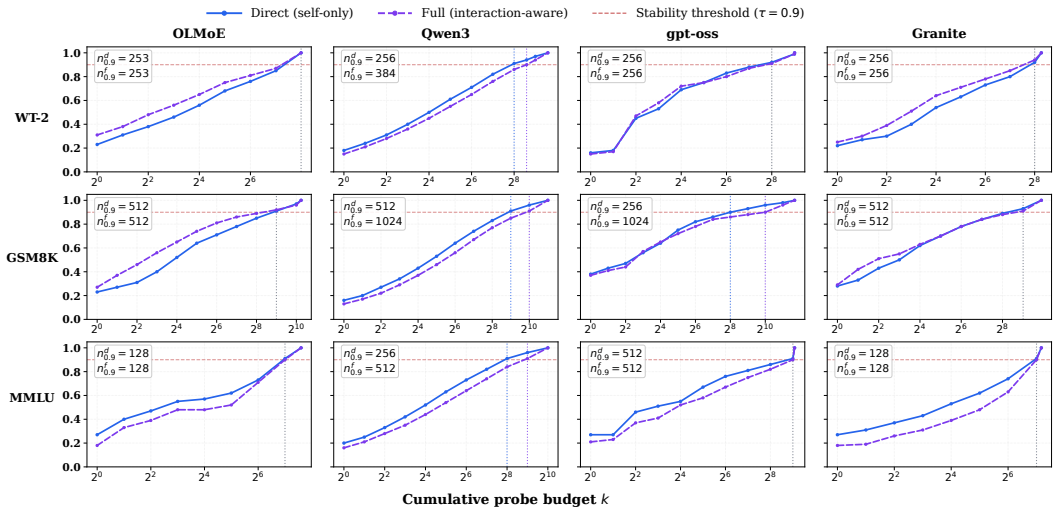


Figure 6: Probe-prefix ranking stability. Kendall $\tau$ convergence for the direct and full expert scores across the measured model/task probes.

# F Use of Existing Assets

All evaluated checkpoints and benchmark datasets are publicly released research assets. We cite the corresponding model reports and benchmark papers in the main text. The manuscript specifies checkpoint families, data splits, prompt conventions, and random-selection seeds.

# G Related Work

## G.1 Sparse MoE routing, scaling, and specialization

Mixture-of-experts models originate from modular conditional computation in which a learned gate partitions inputs among specialized predictors (Jacobs et al., 1991; Jordan & Jacobs, 1994). Modern sparse Transformers turn conditional computation into a scaling mechanism: sparsely gated experts (Shazeer et al., 2017), GShard (Lepikhin et al., 2021), Switch Transformers (Fedus et al., 2022), BASE layers (Lewis et al., 2021), GLaM (Du et al., 2022), and autoregressive MoE studies (Artetxe et al., 2022) scale capacity while keeping per-token computation sparse. Recent architectures continue to refine expert specialization and routing structure (Dai et al., 2024; Muennighoff et al., 2025).

The routing mechanism is itself an active design axis. StableMoE reduces routing fluctuation (Dai et al., 2022b); Expert Choice reverses token-choice routing to obtain balanced expert capacity (Zhou et al., 2022); and analyses of representation collapse show that routing objectives can alter the geometry of token representations (Chi et al., 2022). More recent routing designs explicitly promote diversity among selected experts or align routing scores with expert representation structure (Zheng et al., 2025; Cheng et al., 2026). Empirical studies of trained MoE LLMs further find structure in expert assignments, including context sensitivity (Arnold et al., 2024), expert- and layer-level behavioral differentiation (Lo et al., 2025), and semantic routing (Olson et al., 2025). Together, these studies show that routing traces reveal conditional-computation patterns, specialization, and allocation structure. Their primary object is the selection process—which experts are chosen and how those assignments behave—rather than the downstream task utility of the selected computation.

## G.2 Expert importance, interventions, and cooperative pruning

Methods for expert analysis and compression use router statistics, activation patterns, and related local signals to identify or remove important MoE components (Xie et al., 2024; Zhou et al., 2025; Su et al., 2025). Intervention-based interpretability in dense Transformers similarly localizes function by perturbing internal computations: knowledge neurons connect specific feed-forward units to

factual recall (Dai et al., 2022a), while causal tracing and model editing identify feed-forward computations that mediate factual associations (Meng et al., 2022). MoE-specific studies have begun to analyze knowledge attribution and collaboration between shared and routed experts (Li et al., 2025), and targeted expert fine-tuning has been used to exploit specialization for context faithfulness (Bai et al., 2025).

Cooperative pruning makes interactions between experts explicit. Shapley-MoE (Huang et al., 2025) formulates expert pruning as a game over expert subsets, while SHAPE (Zhang, 2026) restricts coalitions to combinations observed under sparse routing. These approaches establish that expert value can depend on joint coalitions rather than isolated per-expert statistics. Across this literature, importance depends on coalition construction, the counterfactual used for removal, and whether routing is held fixed or recomputed after intervention.

### G.3 Shapley interaction, local curvature, and discrete routing

The Shapley value (Shapley, 1953) provides an axiomatic allocation of a set function and underlies feature, data, neuron, and model attribution (Lundberg & Lee, 2017; Sundararajan et al., 2017; Ghorbani & Zou, 2019; 2020; Li et al., 2024). Extensions such as the Shapley–Taylor index (Sundararajan et al., 2020) and Integrated Hessians (Janizek et al., 2021) make higher-order interactions explicit. Model Shapley (Xu et al., 2025) uses local gradient and curvature statistics to avoid explicit parameter-coalition enumeration; related second-order sensitivity methods appear in pruning and influence estimation (LeCun et al., 1989; Hassibi & Stork, 1992; Koh & Liang, 2017). These methods show how local differential structure can approximate combinatorial attribution while retaining explicit interactions.

Routing introduces a separate technical issue because sparse top-$k$ membership is discrete. Score-function estimators (Williams, 1992), straight-through estimators (Bengio et al., 2013), and continuous relaxations (Jang et al., 2017; Maddison et al., 2017) provide generic tools for differentiating through discrete choices. Recent geometric analyses of sparse MoE routing characterize discontinuities and selected–excluded switching boundaries of top-$k$ regions (Tran Huu et al., 2026). This view makes local membership changes explicit and separates routing-decision geometry from the computations selected on either side of the boundary.