ON THE ADDITIVITY OF EXPERT CONTRIBUTIONS IN MIXTURE-OF-EXPERTS MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Sparse Mixture-of-Experts (MoE) language models activate a small subset of experts for each token, yet how strongly these experts interact in their effects on the loss remains less well understood. First, we evaluate six open-weight MoE language models on MMLU and compare the loss changes caused by removing experts individually and in pairs within the same layer. We find that removing two experts together changes the loss by approximately the sum of the changes caused by removing each separately, so the effects of experts on the loss are nearly additive. Second, we examine how deviations from this additive relation, which we call interaction, vary with domain overlap and layer depth. Pairs with greater domain overlap exhibit larger absolute interaction when their individual effects on the loss are comparable, while absolute interaction generally decreases in later layers. Finally, we study a single-layer MoE with squared loss, where interaction equals the inner product between expert contributions and vanishes when their outputs are orthogonal. We examine this connection on data generated by orthogonal linear teachers. Training from Gaussian initialization produces near-orthogonal expert outputs and reduces interaction relative to the loss changes caused by removing experts individually. We further prove that training started near the teacher solution recovers it at an exponential rate, yielding orthogonal outputs and vanishing interaction. These results establish this recovery as a sufficient mechanism for additivity in the controlled model.

1 INTRODUCTION

Sparse Mixture-of-Experts (MoE) architectures are widely adopted in recent large language models (DeepSeek-AI et al., 2026; Qwen Team, 2026; Gemma Team, 2026). An MoE layer replaces a single feed-forward network with a collection of expert networks and an input-dependent router. For each token, the router selects a small subset of experts, whose outputs are combined through a weighted sum. The model can thus draw on the parameters of many experts while evaluating only a few of them per token, which provides a flexible way to balance model capacity and computational cost. Accordingly, different experts are expected to specialize in different inputs (Shazeer et al., 2017; Dai et al., 2024), while the experts selected for a token collaborate to produce its output.

Research into the internal behavior of MoE models has so far focused on expert specialization, that is, which experts are selected for which inputs. For example, Xue et al. (2024) analyze token-to-expert routing and identify largely context-independent preferences established early in training. Hu et al. (2026) distinguish experts associated with particular domains from those exerting a strong causal influence on model predictions. Moving beyond individual experts, Tang et al. (2026) identify semantically coherent expert groups through cross-layer coactivation. These studies investigate expert specialization and organization, but how the contributions of experts in the same layer combine to affect the loss is less understood. Although these contributions add exactly within a layer, the layers after it and the loss are nonlinear, so the effect of one expert on the loss may depend on which other experts are present. We therefore ask:

How strongly does each expert interact with the others in affecting the loss?

First, we examine whether the combined effect of two experts on the loss equals the sum of their individual effects. We compare the loss change caused by removing two experts together

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.

(double knockout) with the sum of the changes caused by removing each expert separately (single knockout). The difference measures their interaction, with zero interaction corresponding to exact additivity. We conduct this analysis on six open-weight MoE language models from five model families, evaluated on MMLU (Hendrycks et al., 2021), covering every pair of routed experts in every MoE layer. Treating MMLU subjects as domains, we find that domain-averaged double-knockout effects closely track summed single-knockout effects, so the effects of experts on the loss are nearly additive (Section 3.1).

Second, we examine how interaction varies with domain overlap and layer depth. Since two experts can interact only on inputs to which both contribute, their domain preferences may affect how much they interact. We find that pairs with greater overlap in their activation profiles exhibit larger absolute interaction at comparable single-knockout effect sizes (Section 3.2). We also compare layers, since expert contributions from earlier layers pass through more computation before reaching the loss. Absolute interaction generally decreases with depth and is lowest in the last MoE layer across all six models (Section 3.3). Because only the output normalization, unembedding, and softmax follow the last MoE layer, this pattern suggests that the interaction mainly forms as expert outputs propagate through the intermediate layers.

Finally, we examine how training can produce additive expert effects in a single-layer MoE. In this model, squared loss acts directly on the layer output, so no downstream computation follows and interaction equals the inner product of the two experts' contributions, which isolates the role of output geometry. On data generated by orthogonal linear teachers, training from Gaussian initialization produces near-orthogonal expert outputs and reduces interaction relative to single-knockout effects (Section 4.1). We further prove that, when initialized sufficiently close to a teacher-equivalent solution, the router and experts converge to it under population gradient flow and gradient descent, which drives output cosine and both absolute and normalized interaction to zero (Section 4.2).

Overall, each expert affects the loss nearly independently of the others, even though the computation after its layer is nonlinear. Two experts interact mainly on the domains they share, so the domains an expert serves indicate not only its specialization but also which experts it interacts with. In the single-layer MoE, we show that training can make the effects of experts on the loss additive by driving their outputs toward orthogonality.

2 Background and Related Work

Sparse MoE models select a subset of experts for each token and combine their outputs using input-dependent routing weights (Shazeer et al., 2017; Fedus et al., 2022). Whether this routing produces domain-specialized experts depends on the model. Mixtral reports routing patterns more closely associated with syntax than domain (Jiang et al., 2024), and OpenMoE finds that routing depends largely on token identity and is learned early in training (Xue et al., 2024). ST-MoE similarly observes specialization by token categories such as numbers and proper nouns, but little specialization by language (Zoph et al., 2022). In contrast, OLMoE documents stronger domain and vocabulary specialization than Mixtral (Muennighoff et al., 2025). Multilingual analyses further reveal layer-dependent structure, with more language-specific routing in early and late layers and stronger cross-language sharing in middle layers (Bandarkar et al., 2026). Recent work connects expert-usage similarity to hidden-state geometry, cautioning against interpreting routing preferences alone as domain expertise (Wang et al., 2026b). Together, these studies characterize which inputs experts receive, but not necessarily the functions experts perform on these inputs.

The degree of specialization also depends on architecture and training. DeepSeekMoE combines fine-grained experts with shared experts that capture common knowledge (Dai et al., 2024), and global-batch load balancing permits stronger domain specialization than balancing within micro-batches (Qiu et al., 2025). Expert Choice routing instead lets experts select tokens, controlling expert capacity and mitigating under- or over-specialization caused by imbalanced routing (Zhou et al., 2022). Memory-aware routing uses historical token–expert assignments to stabilize specialization while maintaining balanced expert utilization (Hou et al., 2026). EMO restricts the tokens of each document to a common subset of experts (Wang et al., 2026a). Whereas these designs encourage specialization, we relate how experts in pretrained models specialize to how they interact.

Several studies examine what experts do, individually and together. Activation-based domain experts can differ from experts with strong effects on model predictions (Hu et al., 2026). Beyond individual experts, hierarchical dictionary learning identifies semantically coherent groups from cross-layer coactivation (Tang et al., 2026). Expert-aware causal tracing asks whether restoring a corrupted factual prediction localizes to one expert or depends on the routed set (Lu et al., 2026). In dense language models, ablation studies find that when an attention layer or head is removed, later components change their behavior to compensate (McGrath et al., 2023; Rushing & Nanda, 2024). The effect of removing one component can thus depend on which others are present, and we study this dependence among MoE experts through the pairwise interaction of their effects on the loss.

Other work studies the geometry of expert outputs and router weights rather than the inputs experts receive. Routers tend to select experts with larger output norms, and expert outputs grow less similar in deeper layers, except the last (Lo et al., 2025). Router and expert gradients also share input directions (Ahrac et al., 2026), and orthogonality and routing-variance objectives have been introduced to promote specialization during post-training (Guo et al., 2025). Rather than imposing orthogonality, our controlled study asks when loss minimization alone can recover orthogonal teacher outputs.

MoE learning theory examines how experts and routers can learn to specialize from data. Existing results address latent clusters in classification (Chen et al., 2022) and in regression (Kawata et al., 2025), as well as joint router–expert specialization across heterogeneous distributions (Farhat et al., 2025). Parameter recovery has also been studied using distinct expert and router objectives (Makkuva et al., 2020) and likelihood-based EM (Fruytier et al., 2025). Our analysis concerns population gradient training of a softmax router and linear experts under squared loss on the model output. Following local-curvature approaches to neural-network and matrix recovery (Zhong et al., 2017; Wei et al., 2016), we establish local convergence near an orthogonal teacher solution and relate parameter recovery to vanishing output cosine and interaction.

3 Expert Interactions in Pretrained MoE Models

We present empirical findings on expert interactions in pretrained MoE language models. Section 3.1 identifies a pattern of near-additive loss effects under expert removal. We then examine how interaction relates to the domains experts serve (Section 3.2) and how it varies with depth (Section 3.3).

Models and data. We study six open-weight MoE language models from five model families (Table 1). In the following figures and tables, these models are referred to as OLMoE (Muennighoff et al., 2025), Qwen2 (Yang et al., 2024), Qwen3 (Yang et al., 2025), DeepSeek-V2-Lite (DeepSeek-AI et al., 2024), Gemma4 (Gemma Team, 2026), and Moonlight (Liu et al., 2025). Our analysis covers every pair of routed experts in every MoE layer, totaling over 800k expert pairs across the six models. Evaluation uses all 57 MMLU subjects (Hendrycks et al., 2021), each treated as one domain, with up to 200 questions per domain. Each question and its answer options form one prompt (Appendix A). We measure the language-modeling loss on these prompts as the next-token cross-entropy averaged over their tokens, in nats per token (prompt-token loss).

Table 1: Models used in this study, with active and total parameters, N routed experts per MoE layer, k of them selected per token, and L MoE layers.

Model	Active	Total	N	k	L
OLMoE-1B-7B	1.3B	6.9B	64	8	16
Qwen2-57B-A14B	14.0B	57.0B	64	8	28
Qwen3-30B-A3B	3.3B	30.5B	128	8	48
DeepSeek-V2-Lite	2.4B	16.0B	64	6	26
Gemma-4-26B-A4B	3.8B	25.2B	128	8	30
Moonlight-16B-A3B	3.0B	16.0B	64	6	26

3.1 Experts Are Nearly Additive

We evaluate the relationships among experts through their effects on end-to-end performance, examining how individual experts and expert pairs contribute to the loss. Specifically, let x denote the input to the l-th MoE layer and $s_i^l(x)$ the routing weight of the i-th expert, where $s_i^l(x) = 0$ if the expert is not selected. The output of the MoE layer is

$$y^l(x) = \sum_{i=1}^{N} g_i^l(x) = \sum_{i=1}^{N} s_i^l(x)\mathrm{FFN}_i^l(x),$$

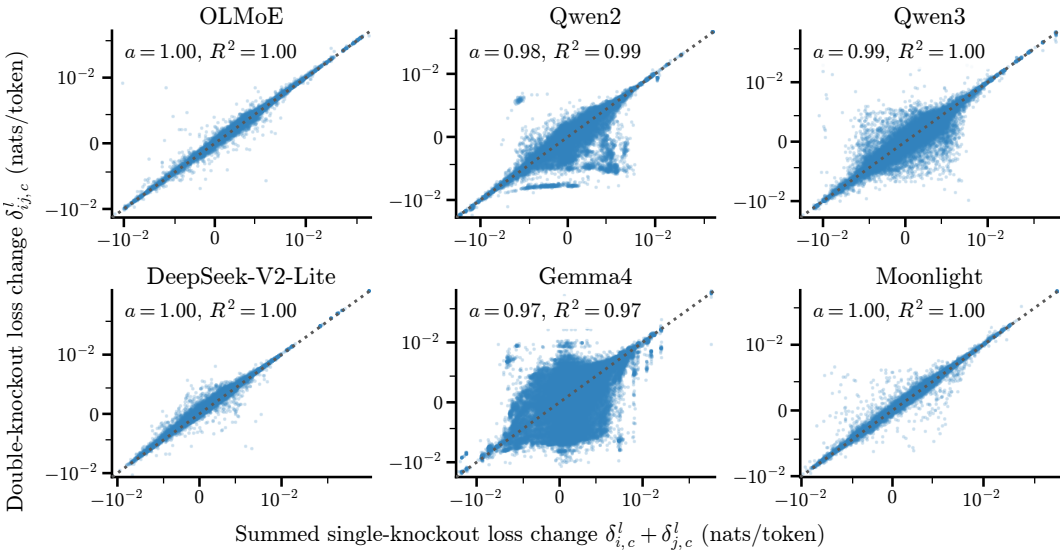


Figure 1: **Double-knockout loss changes track summed single-knockout loss changes.** Each point is one expert pair, with signed loss changes averaged over domains, so its vertical distance from the dotted line $y = x$ is its domain-averaged interaction. Negative values mean that removal lowers the loss, and a and R^2 are the slope and coefficient of determination of the least-squares fit $y = ax$ through the origin. Both axes use symmetric-log scaling, linear within $[-10^{-3}, 10^{-3}]$.

where $\mathrm{FFN}^l_i(\cdot)$ denotes the i-th expert in the l-th layer. To quantify the effects of individual experts and their interactions, we explicitly ablate selected expert outputs. Specifically, the MoE outputs after ablating expert i (*single knockout*) and jointly ablating experts i and j (*double knockout*) are

$$y^l_{-i}(x) = y^l(x) - g^l_i(x), \qquad y^l_{-i,-j}(x) = y^l(x) - g^l_i(x) - g^l_j(x).$$

The replacement is applied at every token of a prompt. The remaining active experts retain their original routing weights at the intervened layer, shared-expert contributions are retained, and all downstream layers are evaluated normally. Let $\delta^l_{i,c}$ denote the resulting change in the prompt-token loss of domain c when $y^l(x)$ is replaced with $y^l_{-i}(x)$, and let $\delta^l_{ij,c}$ denote the change when $y^l(x)$ is replaced with $y^l_{-i,-j}(x)$. We define the *interaction* as $r^l_{ij,c} = \delta^l_{ij,c} - \delta^l_{i,c} - \delta^l_{j,c}$, which measures the non-additive effect between experts i and j. In particular, $r^l_{ij,c} = 0$ indicates that their effects are additive, while a nonzero interaction indicates coupling between the two experts.

Figure 1 plots, for each expert pair, the domain average of $\delta^l_{i,c} + \delta^l_{j,c}$ against that of $\delta^l_{ij,c}$, so the vertical distance of each point from the line $y = x$ equals the pair's domain-averaged interaction. Across all six models, the points lie close to this line, with least-squares slopes of 0.97 to 1.00 through the origin and R^2 values of at least 0.97, although the dispersion is largest for Gemma4. Thus, jointly removing two experts changes the loss by approximately the sum of their single-knockout effects. The same pattern holds for the loss on the correct answer letter (Table 3). These results suggest that pairwise expert effects in MoEs are largely additive.

Despite the approximate additivity, small interactions $r^l_{ij,c}$ remain. We hypothesize that these interactions are related to two factors: (1) the domains the two experts serve and (2) how their outputs are subsequently processed by later layers. Since an expert contributes nothing to a prompt that never selects it, experts specializing in distinct domains should exhibit more additive effects, whereas experts that serve the same domains may interact more strongly. Moreover, since ablation changes the layer output by exactly the sum of the two contributions, the interaction arises only as their outputs propagate through later layers, so experts in earlier layers, whose outputs pass through more of them, may deviate more from additivity. Motivated by these intuitions, we next examine $r^l_{ij,c}$ as a function of domain (Section 3.2) and layer depth (Section 3.3).

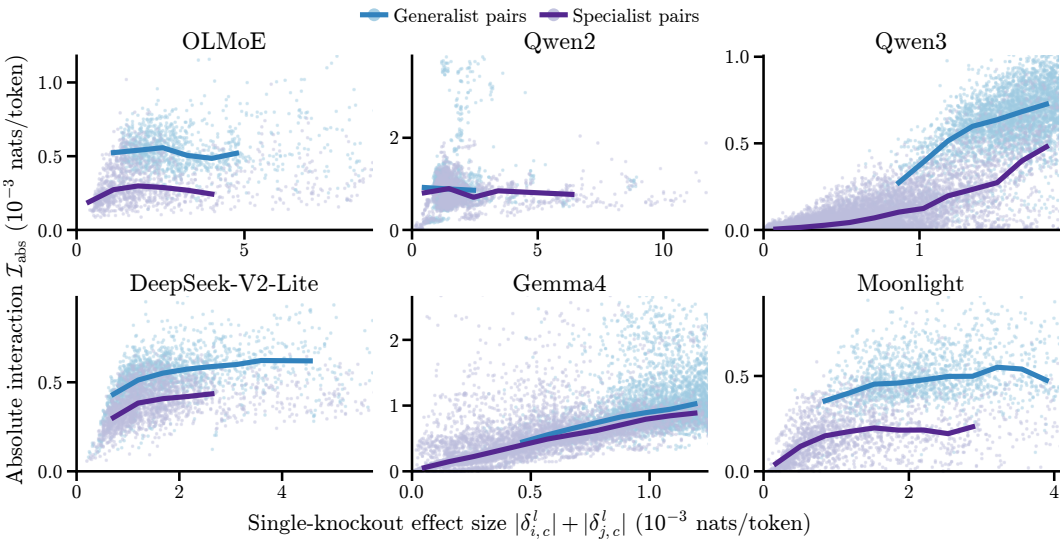


Figure 2: **Specialist pairs interact less than generalist pairs of the same effect size in most models.** Each point is one expert pair, placed by its single-knockout effect size and absolute interaction $\mathcal{I}_{\text{abs}}$, both averaged over domains. Curves are binned medians for specialist pairs (purple) and generalist pairs (blue), with layers combined. The axes cover the effect sizes of 98% of specialist pairs.

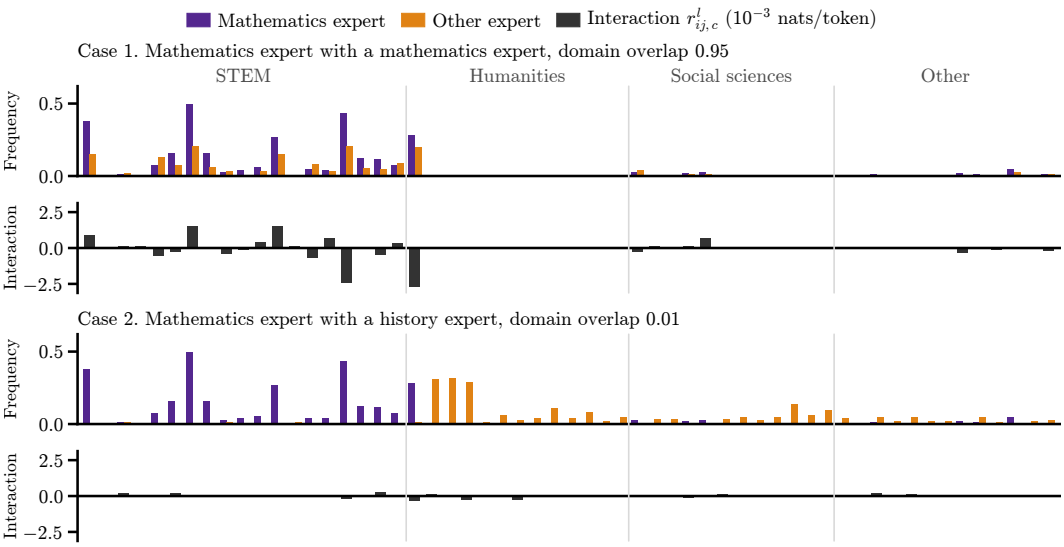


Figure 3: **Two experts interact on the domains they share.** In Qwen3 layer 24, a mathematics specialist is paired with another mathematics specialist (Case 1) and with a history specialist (Case 2). For each pair, the upper strip shows the two experts' activation frequencies and the lower strip their interaction in each of the 57 MMLU domains, grouped by category. Figure 10 labels every domain in the same order.

3.2 Experts That Share Domains Interact More

We first study how the interaction depends on the domains that the experts serve, with the specialization of an expert measured by the coefficient of variation (CV) of its activation frequency across domains, i.e., the fraction of a domain's tokens routed to it. In each layer, we regard the 20% of experts with the highest CV as *specialists* and the 20% with the lowest CV as *generalists*. This choice balances the contrast in specialization against the number of pairs (Appendix B.2). We call a pair of two specialists (generalists) a *specialist pair* (*generalist pair*).

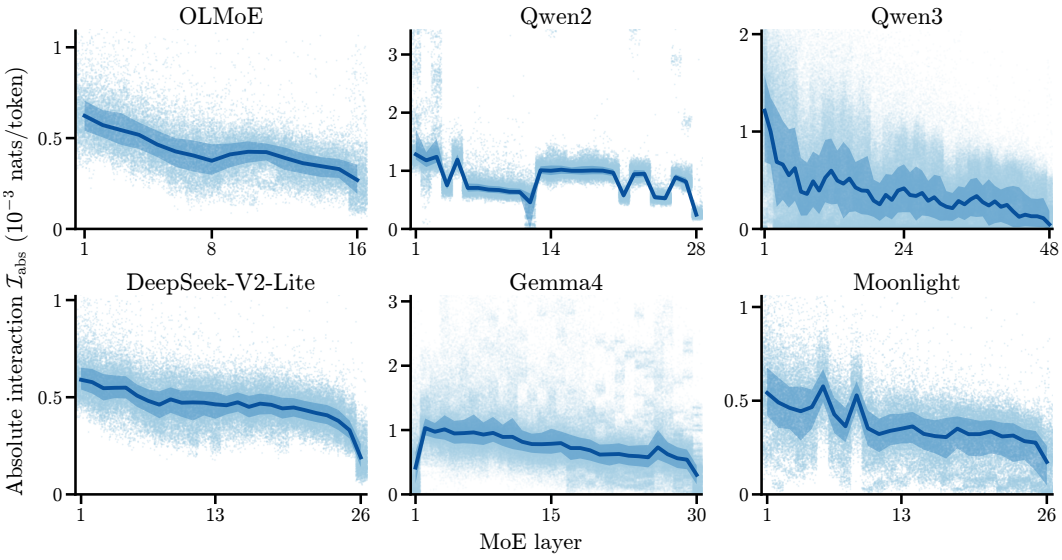


Figure 4: **Absolute interaction decreases with depth.** Each point is one expert pair, spread horizontally within its MoE layer and placed vertically by its absolute interaction $\mathcal{I}_{\text{abs}}$, averaged over domains. Curves are medians over the pairs of each layer, and bands span the 25th to 75th percentiles. The vertical axis covers 99% of each model's pairs.

Because the interaction residual $r^l_{ij,c}$ of a given expert pair often changes sign across domains, the aggregate view in Figure 1 obscures its domain dependence. To examine this variation more closely, we quantify the interaction strength between experts i and j using their *absolute interaction* $\mathcal{I}_{\text{abs}}$, defined as the domain average of $|r^l_{ij,c}|$. Similarly, we define their single-knockout effect size as the domain average of $|\delta^l_{i,c}| + |\delta^l_{j,c}|$.

Figure 2 plots the absolute interaction of generalist pairs and specialist pairs against their single-knockout effect size. At the same effect size (Appendix B.3), generalist pairs interact more than specialist pairs in most models.

This difference can be explained by how the two experts share domains. Two generalists share all domains, a generalist and a specialist share only the domains of the specialist, and two specialists share domains only if they serve the same ones. Figure 3 illustrates this with the interaction of two specialist pairs in Qwen3 on each domain. Each panel also reports the domain overlap of the pair, i.e., the cosine similarity of the two experts' activation frequencies across domains, which is close to one if they serve the same domains and close to zero otherwise. Two mathematics experts with an overlap of 0.95 (Case 1) interact on the mathematics domains they share, whereas the interaction between a mathematics expert and a history expert with an overlap of 0.01 (Case 2) is close to zero on all domains. In contrast, a typical generalist pair interacts on every domain, whereas a generalist paired with the mathematics specialist interacts mainly on the mathematics domains (Figure 10). Thus, the interaction mainly appears on the domains where both experts are active.

Across all expert pairs, including pairs of a generalist and a specialist, pairs with larger domain overlap interact more in all six models, and generalist pairs have larger overlap than specialist pairs, which accounts for their larger interaction (Appendix B.4). The association remains positive within nearly every layer after controlling for single-knockout effect size. In Qwen2, whose experts are routed almost uniformly across domains, even specialist pairs have an overlap close to one, which is why its two curves in Figure 2 nearly coincide. These results are consistent with our hypothesis. While prior work uses the domains an expert serves to describe its specialization, these domains also indicate which experts it interacts with.

3.3 Experts Interact Less in Later Layers

We then study how the interaction varies with layer depth. Figure 4 plots the absolute interaction $\mathcal{I}_{\text{abs}}$ of all expert pairs in each MoE layer, together with the median and interquartile range for each

layer. Across all six models, the absolute interaction generally decreases with depth and is smallest in the last MoE layer, where it drops sharply from the preceding layer in most models.

The outputs of the last MoE layer pass only through the output normalization, unembedding, and softmax before reaching the loss. Thus, any interaction between its experts must arise from these final computations. The weak interaction in this layer suggests that these computations induce little coupling between experts, while the stronger interactions in earlier layers emerge as their outputs propagate through subsequent layers. These findings are consistent with our hypothesis.

4 A Controlled Study of Single-Layer MoE Training

The previous sections show that expert effects in MoEs are largely additive, with the remaining interaction residuals related to the domains experts share and to layer depth. In this section, we study a representative task to explain how such additivity emerges during training.

We consider a regression problem with a clustered input distribution. Specifically, each sample is generated by first drawing a cluster index c uniformly from $\{1, \dots, N\}$ and then sampling $x \sim \mathcal{N}(\mu_c, \sigma^2 I_d)$, where $\sigma > 0$, $I_d \in \mathbb{R}^{d\times d}$ is the identity matrix, and $\mu_c \in \mathbb{R}^d$ is the unique cluster center, i.e., $\mu_c \neq \mu_{c'}$ for $c \neq c'$. For ease of analysis, we assume that the cluster centers have equal norm $\|\mu_c\| = M$. Our results can be extended to the general case. For the input generated from the c-th cluster, the output is $y = T_c x$, where $T_c \in \mathbb{R}^{m\times d}$ is the teacher mapping for the c-th cluster. In this model, joint router–expert training and the interactions between expert outputs remain analytically tractable. Related cluster-structured models have been studied in theoretical analyses of MoEs (Chen et al., 2022; Kawata et al., 2025).

We adopt a single-layer MoE model to learn these mappings. Let $W = (W_1, \dots, W_N)$ denote the collection of all N expert matrices, with $W_i \in \mathbb{R}^{m\times d}$. The router parameters are column vectors $r_i \in \mathbb{R}^d$, collected as $R = [r_1 \ \cdots \ r_N]^\top \in \mathbb{R}^{N\times d}$; its i-th row is therefore $r_i^\top$. We use full softmax routing to simplify the analysis by removing discrete top-k selection while retaining input-dependent routing. For the joint parameters $\theta = (W, R)$ and an input $x \in \mathbb{R}^d$, full softmax routing gives

$$s_i(x; R) = \frac{e^{r_i^\top x}}{\sum_{j=1}^N e^{r_j^\top x}}, \qquad y_\theta(x) = \sum_{i=1}^N s_i(x; R) W_i x. \tag{4.1}$$

Here $y_\theta(x) \in \mathbb{R}^m$ is the model prediction for input x, and the routing weights satisfy $s_i(x; R) > 0$ and $\sum_{i=1}^N s_i(x; R) = 1$. Both the expert parameters W and router parameters R are updated with the ℓ_2 loss $\mathcal{L}(\theta) = 1/2 \cdot \mathbb{E}_x \|y_\theta(x) - y\|^2$. The expectation is taken over the distribution of x, with $y = T_c x$ when x is generated from the c-th cluster.

The optimal predictor for this least-squares regression problem is $y^*(x) = \mathbb{E}[T_c x \mid x]$, where the expectation is taken with respect to the posterior distribution of the cluster index c given x. Under the equal-norm assumption on the cluster centers, this Gaussian-mixture posterior can be represented by the softmax router $R^* = [r_1^* \ \cdots \ r_N^*]^\top$, where $r_i^* = (\mu_i - \bar{\mu})/\sigma^2$ and $\bar{\mu} = N^{-1} \sum_{i=1}^N \mu_i$. Consequently, $y^*(x) = \sum_{i=1}^N s_i(x; R^*) T_i x$, which is realizable by the MoE model in Equation (4.1).

Before the empirical and theoretical analyses, we first introduce an assumption on the teachers.

Assumption 4.1 (Orthogonal teachers)**.** *Distinct teachers have mutually orthogonal output spaces, expressed as $T_c^\top T_{c'} = 0$ for $c \neq c'$. Consequently, the teacher ranks satisfy $\sum_{c=1}^N \operatorname{rank}(T_c) \leq m$.*

This assumption requires $T_c x$ and $T_{c'} x$ to be orthogonal for every input whenever $c \neq c'$. Intuitively, experts specializing in different domains may produce outputs along distinct directions, making approximate orthogonality plausible. To empirically assess this assumption, we treat pretrained MoE models as teacher models and measure the absolute cosine similarity between the outputs of experts selected for the same token on MMLU. Figure 5 reports the median

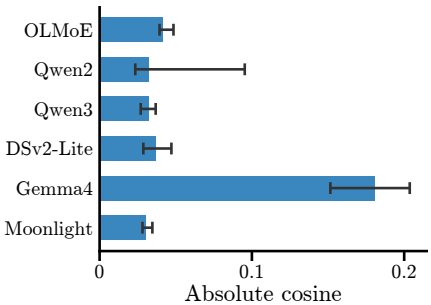


Figure 5: **Expert outputs of the six models are nearly orthogonal.**

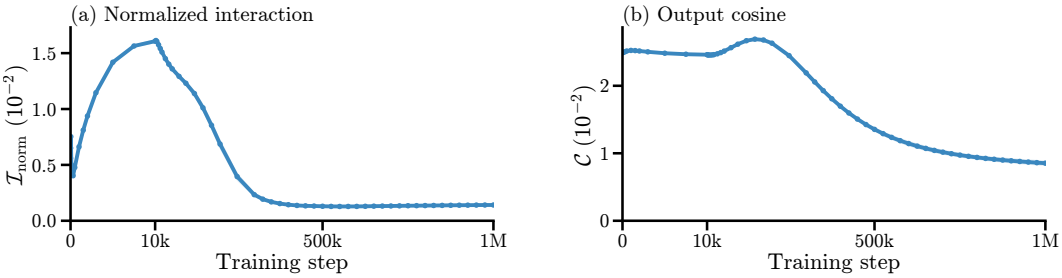


Figure 6: **Training from Gaussian initialization ends with lower normalized interaction and output cosine.** (a) Normalized interaction $\mathcal{I}_{\text{norm}}$ and (b) mean absolute output cosine $\mathcal{C}$ of the single-layer MoE, with clusters in place of domains. Curves and bands show the mean and population SD over three seeds; the first 10k steps take a fifth of each x-axis.

across layers of the average absolute cosine similarity, with error bars indicating the interquartile range. Across all six models, the median ranges from 0.03 to 0.18, indicating that expert outputs are approximately orthogonal and providing empirical support for our assumption.

4.1 Training Lowers Normalized Interaction and Output Cosine

In this subsection, we first show that the additivity in Section 3.1 also emerges in our simplified model, and then link this additivity to the output orthogonality of learned experts.

To test whether the near-additive loss effects observed in Section 3.1 also arise in the single-layer model, we initialize the model parameters from Gaussian distributions and train them using SGD. Here, c indexes Gaussian clusters rather than MMLU domains, and the loss changes $\delta_{i,c}$ and $\delta_{ij,c}$ are averaged over samples within each cluster. For a set $\mathcal{D}$ of evaluated expert-pair–cluster combinations, we define the normalized interaction $\mathcal{I}_{\text{norm}} = \sum_{(ij,c)\in\mathcal{D}} |\delta_{ij,c} - \delta_{i,c} - \delta_{j,c}| / \sum_{(ij,c)\in\mathcal{D}} (|\delta_{i,c}| + |\delta_{j,c}|)$. This quantity measures the aggregate magnitude of the interaction residual relative to the aggregate single-knockout effects. We also define the absolute interaction $\mathcal{I}_{\text{abs}}$ as the average of $|\delta_{ij,c} - \delta_{i,c} - \delta_{j,c}|$ over $\mathcal{D}$. Appendix C provides the experimental details.

In Figure 6(a), $\mathcal{I}_{\text{norm}}$ rises early in training and then declines substantially, followed by a small late rebound. It ends at approximately 1.4×10^{-3}, compared with 5.6×10^{-3} at initialization, and all three seeds finish below their initial values. Thus, the trained model exhibits substantially weaker interaction relative to the single-knockout effects, even though the absolute magnitude of the interaction residual is larger than at initialization.

To understand this additivity, we compute the interaction residual in the single-layer model explicitly as

$$\delta_{ij,c} - \delta_{i,c} - \delta_{j,c} = \mathbb{E}_{x|c}[s_i(x)s_j(x)\langle W_i x, W_j x\rangle]\,. \tag{4.2}$$

The right-hand side depends on two factors, the product of the routing weights $s_i(x)s_j(x)$ and the inner product $\langle W_i x, W_j x\rangle$, which measures the alignment between the outputs of experts i and j. To investigate whether output orthogonality is the main driver of the observed additivity, we next examine the average cosine similarity between expert outputs.

$$\mathcal{C}(W) = \frac{1}{\binom{N}{2}} \sum_{i<j} \mathbb{E}_x \left[\frac{|\langle W_i x, W_j x\rangle|}{\|W_i x\| \cdot \|W_j x\|} \right]. \tag{4.3}$$

Figure 6(b) shows that the mean absolute cosine similarity between expert outputs starts at 2.5×10^{-2}, approximately the value expected for random directions in 1024 dimensions. It rises slightly early in training and then declines to approximately 8.5×10^{-3} after one million steps, about one-third of its initial value. Thus, training drives the expert outputs toward greater orthogonality than would be expected from random directions.

Overall, our empirical study of this representative model confirms the emergence of expert additivity and links it to the increasing orthogonality of expert outputs during training.

4.2 JOINT TRAINING CONVERGES LOCALLY TO ORTHOGONAL OUTPUTS

The experiment in Section 4.1 shows that training can produce near-orthogonal expert outputs. We now analyze local convergence around the orthogonal solution $W_i = T_i$, $R = R^*$. We show that these solutions are locally stable and attracting, that is, sufficiently small perturbations of the router and expert parameters decay under population gradient flow, restoring orthogonal outputs.

Let $\theta(t) = (W(t), R(t))$ denote the parameter trajectory under population gradient flow for the loss $\mathcal{L}(\theta)$, with $\dot{\theta} = (\dot{W}, \dot{R})$, $\dot{W} = -\nabla_W \mathcal{L}$, and $\dot{R} = -\nabla_R \mathcal{L}$. For two parameter values $\theta = (W, R)$ and $\theta' = (W', R')$, we use the Euclidean parameter norm $\|\theta - \theta'\|^2 = \sum_{i=1}^N \|W_i - W_i'\|_F^2 + \|R - R'\|_F^2$ to measure their distance, where $\|\cdot\|_F$ is the Frobenius norm.

To simplify the analysis, we account for two parameter symmetries that leave model predictions unchanged. Adding a common vector to all router rows leaves the routing weights unchanged. We can therefore subtract their mean and work without loss of generality in the centered space $\Theta = \{(W, R) : \sum_{i=1}^N r_i = 0\}$. This shift invariance also implies $\sum_{i=1}^N \nabla_{r_i} \mathcal{L} = 0$, so the router remains centered throughout gradient flow. Simultaneously permuting the experts and their corresponding router rows also leaves predictions unchanged. We therefore let $\mathcal{M} = \{(W, R) \in \Theta : \exists\, \pi \in S_N,\ W_i = T_{\pi(i)},\ r_i = r^*_{\pi(i)} \text{ for all } i\}$ denote the set of all simultaneous permutations of the centered teacher solution, where S_N is the set of permutations of $\{1, \ldots, N\}$. We measure recovery by $\operatorname{dist}(\theta, \mathcal{M}) = \min_{\theta' \in \mathcal{M}} \|\theta - \theta'\|$, the distance to the nearest teacher-equivalent solution. Appendix D.2 gives the full notation.

Theorem 4.1 (Local recovery of orthogonal outputs)**.** *Under Assumption 4.1, there exist model-dependent constants $r, \kappa, K_0 > 0$ such that, if $e_0 = \operatorname{dist}(\theta(0), \mathcal{M}) < r$, the joint gradient flow $\theta(t)$ approaches $\mathcal{M}$ and the mean absolute output cosine decays to zero, with*

$$\operatorname{dist}(\theta(t), \mathcal{M}) \le e_0 e^{-\kappa t}, \qquad \mathcal{C}(W(t)) \le \min\{1, K_0 \sqrt{e_0}\, e^{-\kappa t/2}\}. \tag{4.4}$$

Here e_0 is the initial parameter distance, r is the admissible initialization radius, κ is the local contraction rate, and K_0 controls the scale of the cosine bound.

Theorem 4.1 establishes exponential local convergence of the router and expert parameters to a teacher-equivalent solution, with mean absolute output cosine tending to zero. As the expert contributions converge to orthogonal limits, their inner-product terms vanish, driving absolute interaction $\mathcal{I}_{\text{abs}}$ to zero. For the conditional-mean target y^*, the normalization denominator approaches a positive limit, so normalized interaction $\mathcal{I}_{\text{norm}}$ also tends to zero; Appendix D.1 proves these implications. Together with the experiment in Section 4.1, this identifies a mechanism by which training produces additive expert effects in the single-layer model, providing a controlled counterpart to the near-additive behavior of pretrained language models in Figure 1. Appendix D gives the theorem's proof and its gradient-descent counterpart, with supporting lemmas proved in Appendix E.

5 CONCLUSION

Across six open-weight MoE language models, we find that removing two experts together changes the loss by approximately the sum of the changes caused by removing each separately, so experts' effects on the loss add nearly as their outputs do. Absolute interaction is larger for expert pairs with greater domain overlap at comparable single-knockout effect sizes and generally decreases with layer depth. The domains an expert serves thus indicate not only its specialization but also which experts it interacts with. In a single-layer MoE with squared loss, training from Gaussian initialization produces near-orthogonal expert outputs and reduces normalized interaction. We further prove exponential local convergence to an orthogonal teacher solution, establishing the recovery of orthogonal outputs as a sufficient mechanism for additive expert effects in this setting.

By connecting the near-additivity of expert effects to output orthogonality in a tractable model, our results provide a starting point for understanding how experts interact in pretrained MoE models and how near-additive behavior can emerge during training. Characterizing how these interactions form in the intermediate layers, and extending the theoretical analysis to nonlinear experts and top-k routing, remain important directions for future work.

AI USE STATEMENT

In this work, we used generative AI tools for assistance with proof writing, feedback on research methodology or experiments, and method implementation. We have not used generative AI tools for developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, or proposing or refining hypotheses, and generating synthetic datasets, assisting with translation, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, and interpreting results are not applicable to this work.

REFERENCES

Sagi Ahrac, Noya Hochwald, and Mor Geva. Routers learn the geometry of their experts: Geometric coupling in sparse mixture-of-experts. *arXiv preprint arXiv:2605.12476*, 2026. URL https://arxiv.org/abs/2605.12476.

Lucas Bandarkar, Chenyuan Yang, Mohsen Fayyaz, Junlin Hu, and Nanyun Violet Peng. Multilingual routing in mixture-of-experts. In *International Conference on Learning Representations*, 2026.

Zixiang Chen, Yihe Deng, Yue Wu, Quanquan Gu, and Yuanzhi Li. Towards understanding the mixture-of-experts layer in deep learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=MaYzugDmQV.

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, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.70. URL https://aclanthology.org/2024.acl-long.70/.

DeepSeek-AI, Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, et al. DeepSeek-V2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024. URL https://arxiv.org/abs/2405.04434.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, et al. DeepSeek-V4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026. URL https://arxiv.org/abs/2606.19348.

Yehya Farhat, Hamza ElMokhtar Shili, Fangshuo Liao, Chen Dun, Mirian Del Carmen Hipolito Garcia, Guoqing Zheng, Ahmed Hassan Awadallah, Robert Sim, Dimitrios Dimitriadis, and Anastasios Kyrillidis. Learning to specialize: Joint gating-expert training for adaptive MoEs in decentralized settings. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=3FBByWp6GL.

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.

Quentin Fruytier, Aryan Mokhtari, and Sujay Sanghavi. Learning mixtures of experts with EM: A mirror descent perspective. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=wjZcCbTvrU.

Gemma Team. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026. URL https://arxiv.org/abs/2607.02770.

Hongcan Guo, Haolang Lu, Guoshun Nan, Bolun Chu, Jialin Zhuang, Yuan Yang, Wenhao Che, Xinye Cao, Sicong Leng, Qimei Cui, and Xudong Jiang. Advancing expert specialization for better MoE. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=iydmH9boLb.

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. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Peixuan Hou, Yunbo Hou, Bin Chen, Li He, Jian Xu, Weiping Li, Bo Zheng, and Guojie Song. From pseudo-balancing to true specialization: Memory-aware routing for mixture-of-experts. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 17320–17337, 2026.

Guimin Hu, Meng Li, Qiwei Peng, Lijie Hu, Boyan Xu, and Ruichu Cai. What gets activated: Uncovering domain and driver experts in MoE language models. *arXiv preprint arXiv:2601.10159*, 2026. URL https://arxiv.org/abs/2601.10159.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024. URL https://arxiv.org/abs/2401.04088.

Ryotaro Kawata, Kohsei Matsutani, Yuri Kinoshita, Naoki Nishikawa, and Taiji Suzuki. Mixture of experts provably detect and learn the latent cluster structure in gradient-based learning. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=2xmOEbpYv1.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, Yanru Chen, Huabin Zheng, Yibo Liu, Shaowei Liu, Bohong Yin, Weiran He, Han Zhu, Yuzhi Wang, Jianzhou Wang, Mengnan Dong, Zheng Zhang, Yongsheng Kang, Hao Zhang, Xinran Xu, Yutao Zhang, Yuxin Wu, Xinyu Zhou, and Zhilin Yang. Muon is scalable for LLM training. *arXiv preprint arXiv:2502.16982*, 2025. URL https://arxiv.org/abs/2502.16982.

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*. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.findings-naacl.251/.

Yuetian Lu, Ali Modarressi, Yihong Liu, and Hinrich Schütze. Expert-aware causal tracing of factual recall in sparse MoE language models. *arXiv preprint arXiv:2606.03780*, 2026. URL https://arxiv.org/abs/2606.03780.

Ashok Makkuva, Sewoong Oh, Sreeram Kannan, and Pramod Viswanath. Learning in gated neural networks. In Silvia Chiappa and Roberto Calandra (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 3338–3348. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/makkuva20a.html.

Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. The hydra effect: Emergent self-repair in language model computations. *arXiv preprint arXiv:2307.15771*, 2023. URL https://arxiv.org/abs/2307.15771.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Evan 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 *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=xXTkbTBmqq.

Zihan Qiu, Zeyu Huang, Bo Zheng, Kaiyue Wen, Zekun Wang, Rui Men, Ivan Titov, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Demons in the detail: On implementing load balancing loss for training specialized mixture-of-expert models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.acl-long.249/.

Qwen Team. Qwen3.6-35B-A3B: Agentic coding power, now open to all, April 2026. URL https://qwen.ai/blog?id=qwen3.6-35b-a3b.

Cody Rushing and Neel Nanda. Explorations of self-repair in language models. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=5ZwEifshyo.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=B1ckMDqlg.

Yuanbo Tang, Naifan Zhang, Yan Tang, Meixuan Chen, Shuhan Huang, Tingyu Cao, and Yang Li. Specialization through collaboration: Understanding expert interaction in mixture-of-expert large language models. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2326–2339, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-380-7. doi: 10.18653/v1/2026.eacl-long.104. URL https://aclanthology.org/2026.eacl-long.104/.

Ryan Wang, Akshita Bhagia, and Sewon Min. EMO: Pretraining mixture of experts for emergent modularity. *arXiv preprint arXiv:2605.06663*, 2026a. URL https://arxiv.org/abs/2605.06663.

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

Ke Wei, Jian-Feng Cai, Tony F. Chan, and Shingyu Leung. Guarantees of Riemannian optimization for low rank matrix recovery. *SIAM Journal on Matrix Analysis and Applications*, 37(3):1198–1222, 2016. doi: 10.1137/15M1050525. URL https://doi.org/10.1137/15M1050525.

Fuzhao Xue, Zian Zheng, Yao Fu, Jinjie Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. OpenMoE: An early effort on open mixture-of-experts language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 55625–55655. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/xue24c.html.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Kai Zhong, Zhao Song, Prateek Jain, Peter L. Bartlett, and Inderjit S. Dhillon. Recovery guarantees for one-hidden-layer neural networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 4140–4149. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/zhong17a.html.

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 Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=jdJo1HIVinI.

Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. ST-MoE: Designing stable and transferable sparse expert models. *arXiv preprint arXiv:2202.08906*, 2022.

APPENDIX

A Details of the Language-Model Experiments **13**

B Additional Language-Model Results **14**
B.1 Additivity at the Answer Position 14
B.2 Specialist and Generalist Groups 14
B.3 Activation Frequency and Effect Size 14
B.4 Domain Overlap and Interaction 15

C Details of the Single-Layer MoE Experiments **17**
C.1 Gaussian-Initialization Experiment 17
C.2 Interaction and Cosine Measurements 18

D Theory of the Single-Layer MoE **18**
D.1 Interaction and Expert-Output Inner Products 18
D.2 Target Solutions and Notation 19
D.3 Supporting Lemmas 20
D.4 Proof of Theorem 4.1 21
D.5 Population Gradient Descent 21

E Proofs of the Supporting Lemmas **22**
E.1 Proof of Lemma D.1 22
E.2 Proof of Lemma D.2 23

A DETAILS OF THE LANGUAGE-MODEL EXPERIMENTS

Models. We list the checkpoints in Table 2, all taken from Hugging Face at their main revision. Qwen3-30B-A3B is the post-trained release of Qwen3-30B-A3B-Base, and the other five are base models. Four models also apply shared experts to every token. Specifically, Qwen2 has one large shared expert, DeepSeek-V2-Lite and Moonlight have two each, and Gemma4 has a dense feed-forward block alongside its routed experts. We keep the shared experts and do not knock them out, so our analysis only covers routed experts. In DeepSeek-V2-Lite and Moonlight, the first layer is dense, and their 26 MoE layers start from the second layer. Qwen3, Gemma4, and Moonlight renormalize the routing weights of the selected experts, and knockouts keep these weights unchanged.

Table 2: Hugging Face checkpoints and numbers of shared experts per MoE layer.

Model	Checkpoint	Shared experts
OLMoE	`allenai/OLMoE-1B-7B-0924`	0
Qwen2	`Qwen/Qwen2-57B-A14B`	1
Qwen3	`Qwen/Qwen3-30B-A3B`	0
DeepSeek-V2-Lite	`deepseek-ai/DeepSeek-V2-Lite`	2
Gemma4	`google/gemma-4-26B-A4B`	1
Moonlight	`moonshotai/Moonlight-16B-A3B`	2

Evaluation data. MMLU consists of four-option multiple-choice questions from 57 subjects, which are grouped into STEM, humanities, social sciences, and other. We treat each subject as one domain in our analysis. For each domain, we evaluate 200 questions, or all of them if fewer are available, taken first from the test split and then from the validation and development splits. Each prompt consists of a question, its answer options, and an `Answer:` suffix, without a chat template or in-context examples. In total, we evaluate 9,722 questions across the 57 domains for every model.

B Additional Language-Model Results

B.1 Additivity at the Answer Position

The prompt-token loss in Figure 1 scores every token of a question and its options, whereas MMLU evaluates the answer predicted by the model. Therefore, we also examine the additivity at the answer position using Answer-NLL, the negative log probability of the correct answer letter under the full-vocabulary distribution, averaged over the questions of each domain. Unlike accuracy, Answer-NLL is continuous. Specifically, we replace the prompt-token loss in the definitions of $\delta^l_{i,c}$ and $\delta^l_{ij,c}$ in Section 3.1 with Answer-NLL, and Table 3 compares the domain averages of $\delta^l_{ij,c}$ and $\delta^l_{i,c} + \delta^l_{j,c}$ under both losses. As in Figure 1, the slope a measures the agreement with the additive line $y = x$, and R^2 measures how much of the variation across pairs is explained by the fit.

Table 3: Agreement between double-knockout and summed single-knockout loss changes under prompt-token loss and Answer-NLL, measured by the slope a and R^2 of the least-squares fit $y = ax$ in Figure 1.

	Prompt-token		Answer-NLL	
Model	a	R^2	a	R^2
OLMoE	1.000	0.999	1.001	0.997
Qwen2	0.983	0.991	1.003	0.997
Qwen3	0.991	0.996	0.998	0.998
DeepSeek-V2-Lite	1.001	1.000	1.002	0.999
Gemma4	0.967	0.973	0.966	0.937
Moonlight	1.001	0.999	1.001	0.998

For Answer-NLL, the fitted slopes stay close to one in every model, and the fits explain nearly all of the variation across pairs, again least for Gemma4. Thus, the near-additivity carries over to the answer position, and removing a pair of experts still changes the answer loss by approximately the sum of their single-knockout effects.

B.2 Specialist and Generalist Groups

Figure 7 plots the CV of the experts in each model. Within a layer, the CV increases smoothly from the most uniform expert to the most specialized one, without a gap that marks a natural boundary. Moreover, the scale of the CV differs by more than an order of magnitude across models. In particular, the experts of Qwen2 are routed almost uniformly across domains, so even its specialist tail lies below the generalist tail of every other model, whereas Qwen3 and Gemma4 have strongly specialized experts. Therefore, Section 3.2 defines specialists and generalists by the CV rank within each layer, so a specialist is an expert more specialized than the rest of its layer, rather than one whose CV exceeds a fixed threshold.

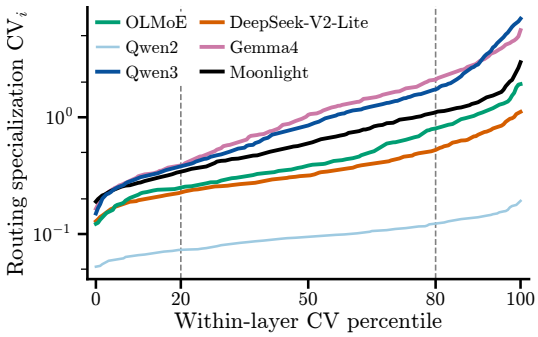


Figure 7: **CV rises smoothly within layers, and its scale differs across models.** Each curve shows a model's CV against the within-layer percentile of experts, from least to most specialized, as a median over layers (logarithmic vertical axis). Dashed lines at the 20th and 80th percentiles bound the generalist and specialist groups.

We choose the 20% tails as a compromise between the separation of the two groups and the number of pairs within each group. Each group then contains about $0.2N$ experts, i.e., 12 for $N = 64$ and 25 for $N = 128$, which gives 66 or 300 pairs per group in each layer. In comparison, the 10% tails leave only 15 or 66 pairs, whereas the 30% tails reach the middle of the distribution.

We exclude experts with zero activation frequency, whose CV is undefined, before ranking. The group size follows the number N of experts in the layer, and ties are broken by expert index.

B.3 Activation Frequency and Effect Size

As shown in Figure 8, experts with lower CV, i.e., those serving the domains more evenly, are activated more often in every model and within every layer. Therefore, generalists are activated more often than specialists in nearly every layer, by a small margin in Qwen2 and by more than a factor of ten in Gemma4.

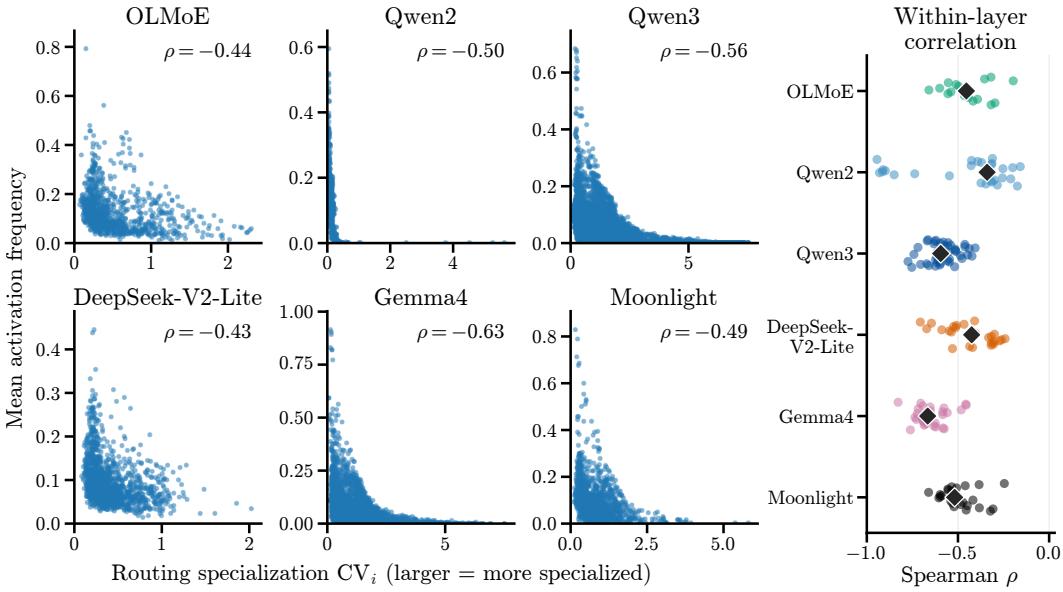


Figure 8: **Generalists are activated more often than specialists.** Left, mean activation frequency against routing CV (higher CV means more specialized routing), one point per expert and layer, with Spearman correlations across all of them. Right, the same correlation within each MoE layer (small points) and its median per model (diamonds); every within-layer correlation is negative.

The mean activation frequency of the two experts in a pair also correlates with their absolute interaction in every layer, but this correlation largely disappears once the single-knockout effect size is controlled for. This suggests that frequently activated experts interact more mainly because removing them changes the loss more. Thus, Figure 2 compares pairs at the same effect size.

B.4 Domain Overlap and Interaction

In this subsection, we give more details on how the domain overlap defined in Section 3.2 relates to the interaction. The two pairs in Figure 3 pair a mathematics specialist in Qwen3 layer 24 (expert 48) with two other specialists. Specifically, among the specialists in this layer with mean activation frequency above 2%, we choose the one with the most similar activation profile (expert 83, also mathematics) and the one with the least similar profile (expert 42, history), and the two pairs have similar single-knockout effect sizes. Figure 9 applies the same rule to a computer science specialist in OLMoE layer 13 (expert 6). Its interaction with another computer science specialist (expert 11, Case 3) is largest on the computer science domains, whereas its interaction with a politics specialist (expert 52, Case 4) is close to zero on all domains.

Figure 10 shows the same view for generalists in Qwen3 layer 24. Specifically, we choose a typical generalist pair (experts 26 and 50), whose effect size and ratio of absolute interaction to effect size are closest to the medians of the layer's generalist pairs. The two experts are active and interact in every domain. Moreover, when the mathematics specialist is paired with a generalist (expert 123, whose domain overlap with expert 48 is the median among the generalists), their interaction mainly appears on the mathematics domains, where both experts are active.

Figure 11 extends the comparison in Figure 2 to all expert pairs, grouping them by their domain overlap. We observe that pairs with larger overlap exhibit larger interaction in the five models other than Qwen2. In Qwen2, 99% of the pairs have an overlap of at least 0.9, which leaves one group.

Since Figure 11 combines all layers, whose overlap and interaction differ, we also examine the relation within each layer. Specifically, we rank the pairs in a layer by their overlap, absolute interaction, and single-knockout effect size, remove from the first two rankings the part explained by the effect size, and compute the correlation between the remaining parts. This partial correlation is positive in every layer of OLMoE, Qwen3, DeepSeek-V2-Lite, Gemma4, and Moonlight, and in 26 of the 28 layers of Qwen2, where it is weak because its experts are routed almost uniformly (Figure 7).

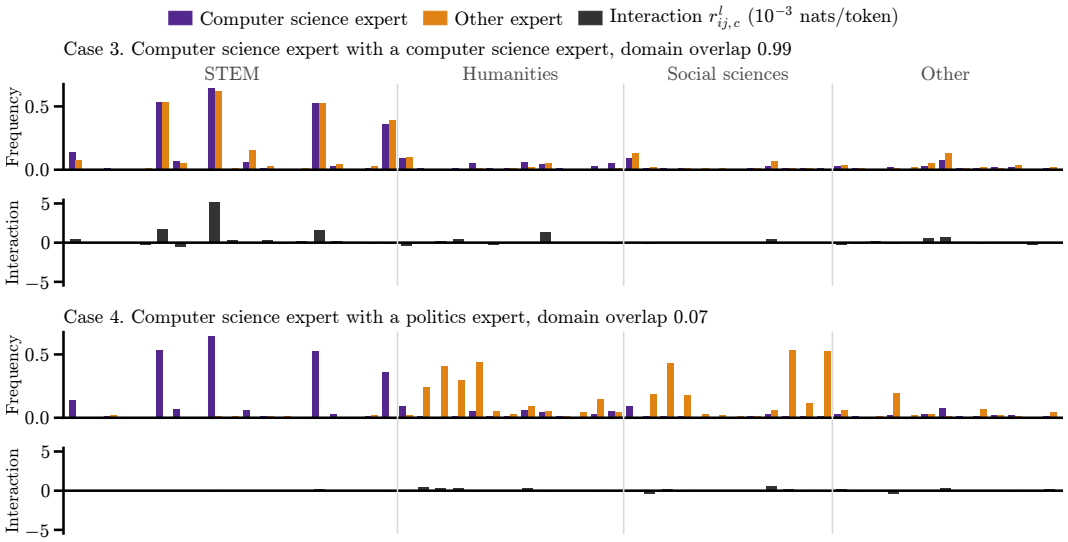


Figure 9: **Interaction follows the domains that two experts share.** In OLMoE layer 13, a computer science specialist (expert 6) is paired with the specialist of most similar activation profile (expert 11, Case 3) and with that of least similar profile (expert 52, Case 4). For each pair, the upper strip shows both experts' activation frequencies and the lower strip their interaction in each of the 57 MMLU domains, grouped by category. Figure 10 labels every domain in the same order.

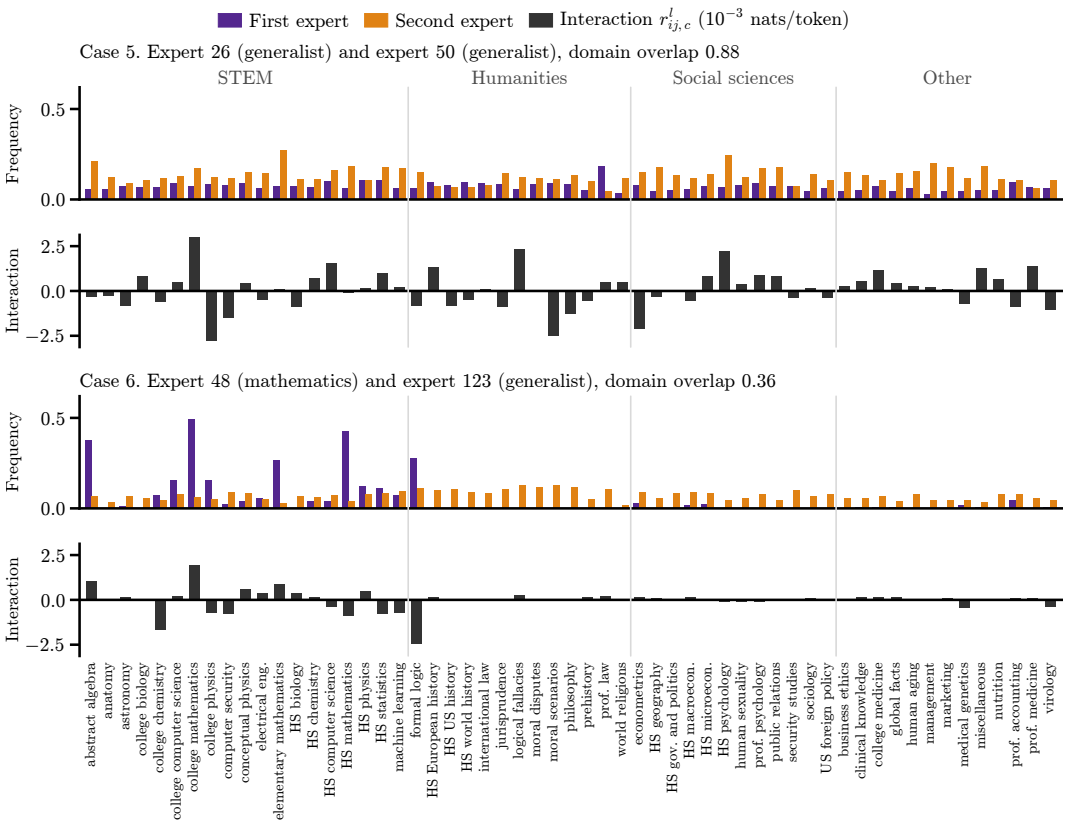


Figure 10: **Generalist pairs interact in every domain.** In Qwen3 layer 24, a typical generalist pair (experts 26 and 50, Case 5) is shown with the mathematics specialist expert 48 paired with a generalist (expert 123, Case 6). Strips show activation frequencies and interaction in every domain, with the interaction on the scale of Figure 3.

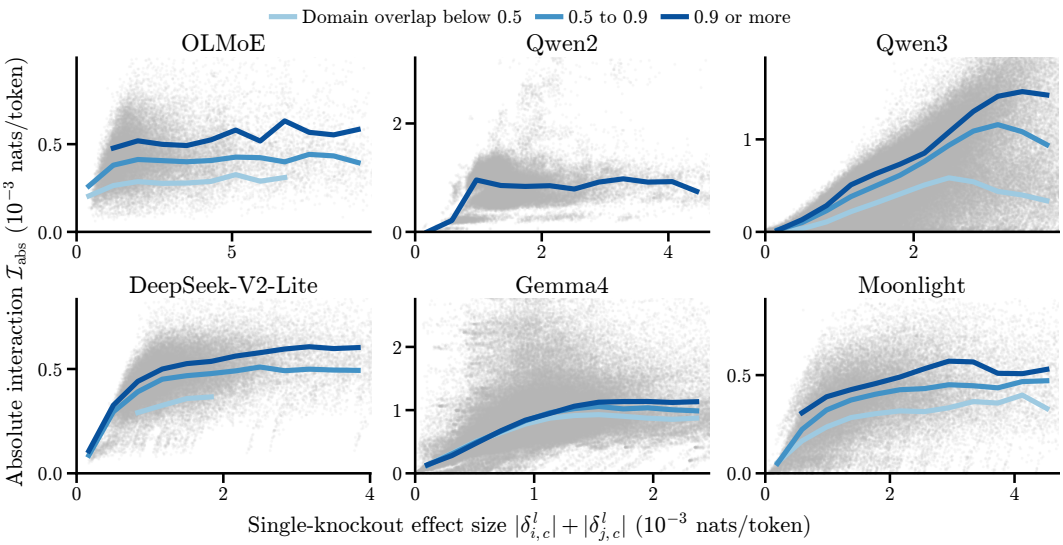


Figure 11: **At the same effect size, pairs with more domain overlap interact more.** Absolute interaction against single-knockout effect size, as in Figure 2, with layers combined. Each gray point is one expert pair, and lines are binned medians for pairs grouped by domain overlap; a group with under 1% of a model's pairs is not drawn, which leaves one group in Qwen2.

Moreover, generalist pairs have the highest overlap, and specialist pairs have the lowest. Using the same within-layer test, we observe that generalist pairs interact more than other pairs with the same effect size, but no more once the overlap is also controlled for, whereas specialist pairs still interact slightly less. These results suggest that the domain overlap accounts for the larger interaction of generalist pairs and for part of the smaller interaction of specialist pairs.

C Details of the Single-Layer MoE Experiments

C.1 Gaussian-Initialization Experiment

Inputs are drawn from 64 uniformly sampled Gaussian clusters with center norm $M = 120$ and standard deviation $\sigma = 3$, and are then rescaled by the standard deviation of the full data matrix rather than normalized one by one. Each sample in cluster c has target $T_c x$, with fixed mutually orthogonal teacher output spaces and rank 16 per teacher. Each student expert is parameterized by two trainable low-rank factors whose product defines W_i. The full softmax router and both expert factors are jointly updated from Gaussian initialization; no teacher parameters are used to initialize the student.

Table 4: Configuration of the single-layer MoE.

Setting	Value
Experts and clusters	64 each
Input and output dimensions	$d = m = 1024$
Teacher / student expert rank	16 / 16
Student experts	Trainable low-rank factors
Router	Trainable full softmax
Samples / minibatch	32,768 / 8,192
Learning rate, router / experts	1.875×10^{-4} / 1.5×10^{-3}
Router initialization SD	$0.8/\sqrt{1024} = 0.025$
Factor initialization SDs	$1/\sqrt{1024}$ and $1/\sqrt{16}$
Training steps / seeds	10^6 / 0, 1, 2
Evaluated expert pairs	All 2,016 unordered pairs

Evaluation uses all fixed samples and all unordered pairs at each checkpoint. Training uses minibatch SGD with distinct learning rates for the expert factors and router, and no weight decay, router freezing, or learning-rate decay. These finite-sample, factorized, randomly initialized dynamics differ from the population dynamics analyzed in Theorem 4.1.

C.2 Interaction and Cosine Measurements

The interaction measures follow Section 4.1, with the index c identifying a Gaussian cluster. For every saved checkpoint, we evaluate the original fixed samples and their cluster-specific targets. Writing $g_i(x) = s_i(x)W_i x$ for the expert contribution, the cluster-level statistics are

$$\begin{aligned} \delta_{i,c} &= \mathbb{E}_{x|c}\left[\tfrac{1}{2}\|g_i(x)\|^2 - \langle y_\theta(x) - T_c x, g_i(x)\rangle\right], \\ \delta_{ij,c} &= \delta_{i,c} + \delta_{j,c} + \mathbb{E}_{x|c}\langle g_i(x), g_j(x)\rangle. \end{aligned} \tag{C.1}$$

Each expectation is estimated over all samples in that cluster. We then take the absolute values of the cluster-level interaction and single-knockout effects, and sum over all clusters and unordered expert pairs. Clusters receive equal weight, as do domains in Section 3.2. We compute $\mathcal{I}_{\mathrm{norm}}$ separately for each seed as the ratio of the summed interaction magnitudes to the summed single-knockout magnitudes. All evaluated denominators are positive.

These statistics are evaluated from saved parameters and the original reference samples, without further training. Direct single- and double-knockout evaluations on a small test case agree with the inner-product calculation to numerical precision. Mean absolute output cosine $\mathcal{C}$ is averaged uniformly over all fixed input–pair combinations.

Figure 6 includes every recorded checkpoint, with no smoothing or seed selection. Curves and bands show the mean and population standard deviation of the three seed-level values. All three seeds end with lower $\mathcal{I}_{\mathrm{norm}}$ and mean absolute output cosine than at initialization, but higher $\mathcal{I}_{\mathrm{abs}}$. The seed-mean $\mathcal{I}_{\mathrm{norm}}$ changes from 5.60×10^{-3} at initialization to 1.41×10^{-3} at one million steps, a reduction of approximately 75%. It reaches an early peak of 1.61×10^{-2} at 12.5k steps and a later minimum of 1.28×10^{-3} at 550k steps before increasing slightly. Over the full run, the seed-mean $\mathcal{I}_{\mathrm{abs}}$ changes from 3.66×10^{-3} to 2.12×10^{-2}. Thus, the lower final normalized interaction does not imply a monotone trajectory or a reduction in absolute interaction.

D Theory of the Single-Layer MoE

This appendix first connects the interaction measures of Section 4.1 to expert-output geometry in the single-layer model (Appendix D.1). It then proves Theorem 4.1 and gives its population gradient-descent counterpart. The argument separates two questions, whether the parameters return to a teacher-equivalent solution and whether this recovery makes the expert outputs orthogonal. Each question has its own obstacle. First, router and expert updates are coupled. At a teacher solution, the prediction residual vanishes, so the loss Hessian is the expected Gram matrix of the output Jacobian. The Gaussian-mixture and orthogonal-teacher structure make this joint curvature positive after removing the router-shift symmetry, which yields local parameter contraction. Second, small teacher-output norms can amplify cosine normalization. A Gaussian small-ball estimate controls these inputs in expectation and converts parameter contraction into the cosine bound without requiring full-rank teachers. We first specify the parameter space and state the two supporting lemmas. We then prove contraction within a fixed solution neighborhood and apply the cosine bound. Appendix E contains the proofs of the lemmas.

D.1 Interaction and Expert-Output Inner Products

For a sample in cluster c, write $g_i(x) = s_i(x)W_i x$ for expert i's contribution and $r_{\mathrm{fit}} = y_\theta(x) - T_c x$ for the prediction residual. The conditional-mean objective replaces $T_c x$ by $y^*(x)$; the knockout identities below hold for either target. During knockout the target and routing weights are fixed, so removing expert i changes the prediction to $y_\theta - g_i$. The per-input loss changes $\delta_i(x)$ and $\delta_{ij}(x)$ are defined as the perturbed loss minus the baseline loss.

Proposition D.1 (Interaction under squared loss)**.** *For squared loss $\frac{1}{2}\|\cdot\|^2$ and any prediction residual,*

$$\delta_i(x) = \tfrac{1}{2}\|g_i\|^2 - \langle r_{\mathrm{fit}}, g_i\rangle, \qquad \delta_{ij}(x) - \delta_i(x) - \delta_j(x) = \langle g_i, g_j\rangle. \tag{D.1}$$

Consequently, the cluster-level loss changes satisfy

$$\delta_{ij,c} - \delta_{i,c} - \delta_{j,c} = \mathbb{E}_{x|c}\langle g_i(x), g_j(x)\rangle. \tag{D.2}$$

Orthogonal expert outputs give zero interaction for every input and cluster, whatever the target.

Proof. Expanding the single-knockout squared loss gives the first identity. In the double-knockout expansion, subtracting the two single-knockout changes cancels the residual-linear terms and individual squared norms, leaving $\langle g_i, g_j \rangle$. Taking the conditional expectation within cluster c gives the second identity. Since $g_i = s_i W_i x$, orthogonal expert outputs make this inner product zero. □

Normalized interaction and output cosine. The normalized interaction $\mathcal{I}_{\mathrm{norm}}$ of Section 4.1 applies directly to these cluster-level effects. Its numerator is the sum of $|\mathbb{E}_{x|c}\langle g_i, g_j \rangle|$, whereas its denominator uses $|\delta_{i,c}| + |\delta_{j,c}|$, with $\delta_{i,c}$ given by Proposition D.1. Multiplying the squared loss by a positive constant scales both sums equally and leaves $\mathcal{I}_{\mathrm{norm}}$ unchanged. The geometric connection follows from

$$\left|\mathbb{E}_{x|c}\langle g_i, g_j \rangle\right| \le \mathbb{E}_{x|c}[s_i s_j \|W_i x\| \, \|W_j x\| \, |\cos(W_i x, W_j x)|] \,, \tag{D.3}$$

where the product is zero if either output vanishes. Orthogonality is sufficient for zero interaction, but a small output cosine alone does not determine $\mathcal{I}_{\mathrm{norm}}$, because contribution magnitudes, residual projections in single-knockout effects, and the averaging order also matter. The numerator takes an absolute value after the signed cluster mean, whereas $\mathcal{C}$ averages absolute cosines over inputs.

Connection to local convergence. Theorem 4.1 concerns the conditional-mean objective and an orthogonal teacher-equivalent solution. In this setting, the same cluster-aggregated interaction measures also vanish under parameter recovery.

Corollary D.1 (Vanishing interaction near an orthogonal solution)**.** *For the conditional-mean target, suppose $r_{\mathrm{fit},t} \to 0$ and $g_{i,t} \to g_{i,*}$ in L^2 within each cluster, for every expert i. Assume that the limiting contributions are pairwise orthogonal almost surely and*

$$\sum_{c, i<j} \mathbb{E}_{x|c}\big(\|g_{i,*}\|^2 + \|g_{j,*}\|^2\big) > 0.$$

Then the cluster-aggregated $\mathcal{I}_{\mathrm{abs}}$ and $\mathcal{I}_{\mathrm{norm}}$ of Section 4.1 tend to zero.

Proof. Cauchy–Schwarz and L^2 convergence imply that each cluster mean $\mathbb{E}_{x|c}\langle g_{i,t}, g_{j,t} \rangle$ tends to zero. The residual projections in the single-knockout effects also vanish, so

$$\sum_{c, i<j} (|\delta_{i,c,t}| + |\delta_{j,c,t}|) \longrightarrow \tfrac{1}{2} \sum_{c, i<j} \mathbb{E}_{x|c}\big(\|g_{i,*}\|^2 + \|g_{j,*}\|^2\big) > 0.$$

There are finitely many clusters and pairs, so both conclusions follow. Parameter convergence in the linear-expert model implies the required L^2 convergence because softmax weights are bounded and Gaussian inputs have finite second moments. The nonzero teachers and positive softmax weights give the positive limiting denominator. □

Scope of the geometric connection. These identities concern squared loss at the model output. For language models, the final cross-entropy loss also depends on nonlinear downstream computation, so Euclidean orthogonality at an intervened layer alone does not imply loss additivity. The exact connection established here applies to the single-layer study in Section 4.

D.2 Target Solutions and Notation

We use the model and assumptions of Section 4. The index c denotes a Gaussian cluster, i, j denote experts, and d, m are the input and output dimensions. All expectations in the convergence analysis are population expectations over the Gaussian-mixture input distribution.

Posterior target. Equal cluster probabilities and equal-norm centers give

$$\mathbb{P}(c = i \,|\, x) = \frac{\exp[-\|x - \mu_i\|^2/(2\sigma^2)]}{\sum_{j=1}^{N} \exp[-\|x - \mu_j\|^2/(2\sigma^2)]} = \frac{\exp(\mu_i^\top x/\sigma^2)}{\sum_{j=1}^{N} \exp(\mu_j^\top x/\sigma^2)}. \tag{D.4}$$

The terms involving $\|x\|^2$ and the common center norm cancel. Thus the centered router $r_i^* = (\mu_i - \bar{\mu})/\sigma^2$, with $\bar{\mu} = N^{-1} \sum_i \mu_i$, realizes these probabilities. We write $s_i^*(x) = s_i(x; R^*)$ and $\theta_* = ((T_i)_{i=1}^N, R^*)$. At this solution, $y_{\theta_*}(x) = y^*(x)$, so θ_* minimizes $\mathcal{L}$, and $\mathcal{C}((T_i)_{i=1}^N) = 0$.

Parameter symmetries and distance. Adding a common vector to the router rows leaves the prediction unchanged and gives $\sum_i \nabla_{r_i}\mathcal{L} = 0$. Their mean therefore remains constant under gradient flow and gradient descent. We remove this common shift by working on $\Theta = \{(W, R) : \sum_i r_i = 0\}$. Simultaneously permuting the experts and router rows gives the finite set of teacher-equivalent solutions

$$\mathcal{M} = \left\{(W, R) \in \Theta : W_i = T_{\pi(i)},\ r_i = r^*_{\pi(i)}\ (i = 1, \dots, N),\ \pi \in S_N\right\}, \tag{D.5}$$

where S_N is the set of permutations of $\{1, \dots, N\}$. Every element of $\mathcal{M}$ minimizes $\mathcal{L}$ and has orthogonal expert outputs. We do not require $\mathcal{M}$ to contain all global minimizers.

The Euclidean parameter norm and distance to this set are

$$\|\theta - \theta'\|^2 = \sum_{i=1}^{N} \|W_i - W'_i\|_F^2 + \|R - R'\|_F^2, \qquad \operatorname{dist}(\theta, \mathcal{M}) = \min_{\theta' \in \mathcal{M}} \|\theta - \theta'\|. \tag{D.6}$$

Vector norms are Euclidean; $\|\cdot\|_F$ and $\|\cdot\|_{\mathrm{op}}$ denote Frobenius and spectral norms. Derivatives use vectorized expert matrices and orthonormal coordinates on the centered-router subspace.

Derivatives. Let $J_\theta(x) = \nabla_\theta y_\theta(x)$ be the output Jacobian in these coordinates and $J_*(x) = J_{\theta_*}(x)$. A parameter direction is $\xi = (U, V)$, where $U = (U_i)_{i=1}^N$, $U_i \in \mathbb{R}^{m\times d}$, and the rows of $V \in \mathbb{R}^{N\times d}$ are $v_i^\top$ with $\sum_i v_i = 0$. Thus $J_\theta(x)\xi$ is the first-order output variation in direction ξ, and $\|\xi\|^2 = \sum_i \|U_i\|_F^2 + \sum_i \|v_i\|^2$. The target $y^*(x)$ is fixed when taking derivatives.

Population objective and sampled targets. The conditional-mean property gives

$$\mathcal{L}(\theta) = \frac{1}{2}\mathbb{E}_x\|y_\theta(x) - y^*(x)\|^2 + \frac{1}{2}\mathbb{E}_{c,x}\|T_c x - y^*(x)\|^2. \tag{D.7}$$

Indeed, the cross term vanishes after conditioning on x. The second term is independent of θ, so $\mathcal{L}$ and the conditional-mean objective, the first term, have identical gradients and Hessians. Their sample gradients differ. With cluster-specific targets, the sample gradient at θ_* is $J_*(x)^\top[y^*(x) - T_c x]$, which need not be zero. The results below concern population gradient flow and gradient descent, not the finite-sample minibatch updates used in Section 4.1.

D.3 Supporting Lemmas

The first lemma supplies positive curvature for the coupled router and expert parameters. The second converts parameter error into output cosine, including when teacher matrices are rank deficient.

Lemma D.1 (Positive curvature at a teacher solution)**.** *Under Assumption 4.1, $\mathcal{L}$ is twice continuously differentiable on Θ, and*

$$H_* := \nabla^2\mathcal{L}(\theta_*) = \mathbb{E}_x[J_*(x)^\top J_*(x)] \succ 0. \tag{D.8}$$

In particular, the eigenvalues $\lambda_* := \lambda_{\min}(H_*)$ and $\Lambda_* := \lambda_{\max}(H_*)$ satisfy $0 < \lambda_* \le \Lambda_* < \infty$.

Centering removes the common-shift subspace of the router. Positive curvature on the remaining parameter space follows from the model assumptions. The proof in Appendix E.1 first eliminates expert perturbations using inputs for which one posterior weight dominates, then eliminates router perturbations using teacher orthogonality.

Lemma D.2 (Parameter error controls output cosine)**.** *Under the same assumptions, every $\theta = (W, R) \in \Theta$ satisfies*

$$\mathcal{C}(W) \le \min\left\{1, K_0\sqrt{\operatorname{dist}(\theta, \mathcal{M})}\right\}, \tag{D.9}$$

where the finite constant

$$K_0 = \frac{8}{N}\left(\frac{2(M^2 + \sigma^2)}{\pi\sigma^2}\right)^{1/4}\left(\sum_{i=1}^{N}\|T_i\|_{\mathrm{op}}^{-2/3}\right)^{3/4} \tag{D.10}$$

depends only on the input distribution and teachers.

The difficulty is normalization, since even a nonzero teacher can have arbitrarily small output norm on Gaussian inputs. Appendix E.2 controls the probability of this event and proves the bound without a positive lower bound on output norms.

D.4 Proof of Theorem 4.1

We combine the lemmas in three steps. We choose a neighborhood of a single solution, prove that the trajectory remains there and contracts, and transfer the parameter bound to output cosine.

Proof. **Step 1. A strongly convex solution neighborhood.** Set $\kappa = \lambda_*/2$. By Lemma D.1 and continuity of the Hessian, there is an $r > 0$ such that

$$\kappa I \preceq \nabla^2 \mathcal{L}(\theta) \preceq \tfrac{3}{2}\Lambda_* I \qquad \text{whenever } \|\theta - \theta_*\| \le r. \tag{D.11}$$

Here and below, I is the identity on the centered parameter space. To obtain these bounds, choose the Hessian variation to be at most $\lambda_*/2$ in spectral norm. Permutations act isometrically on the parameter space and preserve the loss, so the same bounds hold around every element of $\mathcal{M}$.

Because $\mathcal{M}$ is finite and its elements are distinct, we may also choose r smaller than half the minimum distance between distinct solutions. Every point within distance r of $\mathcal{M}$ then has a unique nearest solution. Relabel this solution at initialization as θ_* and keep the labeling fixed. Integrating the lower Hessian bound along the segment from θ_* gives

$$\mathcal{L}(\theta) - \mathcal{L}(\theta_*) \ge \frac{\kappa}{2}\|\theta - \theta_*\|^2, \qquad \langle \theta - \theta_*, \nabla\mathcal{L}(\theta)\rangle \ge \kappa\|\theta - \theta_*\|^2 \tag{D.12}$$

inside this ball. In particular, its center is the only minimizer there.

Step 2. Invariance and parameter contraction. Let $e_0 = \operatorname{dist}(\theta(0), \mathcal{M}) < r$. Along gradient flow, as long as the trajectory lies in the ball,

$$\frac{d}{dt}\frac{1}{2}\|\theta(t) - \theta_*\|^2 = -\langle \theta(t) - \theta_*, \nabla\mathcal{L}(\theta(t))\rangle \le -\kappa\|\theta(t) - \theta_*\|^2. \tag{D.13}$$

Therefore $\|\theta(t) - \theta_*\| \le e_0 e^{-\kappa t}$. A first boundary-hitting time would contradict this bound, since $e_0 < r$. The trajectory remains in the compact ball of radius e_0, and the locally Lipschitz vector field $-\nabla\mathcal{L}$ consequently has a solution for all $t \ge 0$. The nearest solution stays equal to θ_*, yielding

$$\operatorname{dist}(\theta(t), \mathcal{M}) = \|\theta(t) - \theta_*\| \le e_0 e^{-\kappa t}, \qquad \theta(t) \longrightarrow \theta_*. \tag{D.14}$$

Step 3. Recovery of orthogonal outputs. Lemma D.2 now gives

$$\mathcal{C}(W(t)) \le \min\{1, K_0\sqrt{e_0}\, e^{-\kappa t/2}\} \longrightarrow 0. \tag{D.15}$$

This proves both bounds in Theorem 4.1. For an uncentered router, subtract its initial row mean before applying the theorem and restore it afterward; predictions and output cosine are unchanged. □

The result establishes local recovery of the nearest teacher-equivalent solution. It neither assumes global identifiability nor proves convergence from Gaussian initialization. The cosine bound decreases with time, but does not require the measured cosine to decrease at every instant. Corollary D.1 connects the recovered solution to vanishing interaction.

D.5 Population Gradient Descent

The same curvature bounds also control simultaneous discrete updates of the router and experts.

Corollary D.2 (Local recovery under gradient descent)**.** *Use the radius r and rate κ from the preceding proof. For $\theta_{k+1} = \theta_k - \eta\nabla\mathcal{L}(\theta_k)$, suppose $e_0 = \operatorname{dist}(\theta_0, \mathcal{M}) < r$ and $0 < \eta \le 2/(3\Lambda_*)$. Then all iterates remain in the same solution neighborhood and*

$$\begin{aligned} \operatorname{dist}(\theta_k, \mathcal{M}) &\le (1 - \eta\kappa)^k e_0, \\ \mathcal{C}(W_k) &\le \min\{1, K_0\sqrt{e_0}\,(1 - \eta\kappa)^{k/2}\} \longrightarrow 0. \end{aligned} \tag{D.16}$$

Proof. Relabel the solution nearest to initialization as θ_*. Whenever $\|\theta_k - \theta_*\| < r$, the Hessian averaged along the segment to the current iterate,

$$\overline{H}_k = \int_0^1 \nabla^2\mathcal{L}\big(\theta_* + u(\theta_k - \theta_*)\big)\, du,$$

satisfies $\kappa I \preceq \overline{H}_k \preceq \frac{3}{2}\Lambda_* I$. Since $\nabla\mathcal{L}(\theta_*) = 0$, the update obeys

$$\|\theta_{k+1} - \theta_*\| = \|(I - \eta\overline{H}_k)(\theta_k - \theta_*)\| \le (1 - \eta\kappa)\|\theta_k - \theta_*\|. \tag{D.17}$$

The step-size condition makes $I - \eta\overline{H}_k$ positive semidefinite and $0 < 1 - \eta\kappa < 1$. Induction proves invariance of the initial solution neighborhood and the distance bound. Lemma D.2 then gives the cosine bound. The parameter convergence also satisfies the hypotheses of Corollary D.1. □

E Proofs of the Supporting Lemmas

We prove the two lemmas used in Appendix D.

E.1 Proof of Lemma D.1

Proof. **Step 1. Smoothness and the Hessian at the solution.** On every compact parameter set, the model output and its first two parameter derivatives are bounded by polynomials in $\|x\|$. This follows by differentiating the linear experts and the softmax router, whose logit derivatives are bounded. The Gaussian mixture has finite moments of every order, so dominated differentiation gives the continuous population Hessian

$$\nabla^2\mathcal{L}(\theta) = \mathbb{E}_x\left[J_\theta(x)^\top J_\theta(x) + \sum_{a=1}^{m}\big(y_\theta(x) - y^*(x)\big)_a \nabla_\theta^2 (y_\theta(x))_a\right]. \tag{E.1}$$

At θ_* the prediction residual vanishes pointwise. Hence $H_* = \mathbb{E}[J_*^\top J_*]$, and for every centered direction $\xi = (U, V)$,

$$\xi^\top H_*\xi = \mathbb{E}_x\|J_*(x)\xi\|^2, \qquad J_*(x)\xi = \sum_i s_i^*(x)U_i x + \sum_i s_i^*(x)(v_i^\top x)[T_i x - y^*(x)]. \tag{E.2}$$

Suppose $\xi^\top H_*\xi = 0$. Then $J_*(x)\xi = 0$ almost surely. This function is continuous, and the input density is strictly positive everywhere, so it vanishes for every $x \in \mathbb{R}^d$. We show that this forces both U and V to vanish.

Step 2. Eliminate expert perturbations. For expert i, consider the open cone

$$\mathcal{K}_i = \{z \in \mathbb{R}^d : (\mu_i - \mu_j)^\top z > 0 \text{ for all } j \ne i\}.$$

It is nonempty because the equal-norm and distinct-center assumptions give $(\mu_i - \mu_j)^\top \mu_i = \frac{1}{2}\|\mu_i - \mu_j\|^2 > 0$. Fix $z \in \mathcal{K}_i$ and let $\gamma(z) = \min_{j\ne i}(\mu_i - \mu_j)^\top z/\sigma^2 > 0$. Along the input ray $x = \alpha z$ as $\alpha \to \infty$,

$$\frac{s_j^*(\alpha z)}{s_i^*(\alpha z)} = \exp\left[-\frac{\alpha(\mu_i - \mu_j)^\top z}{\sigma^2}\right] \le e^{-\alpha\gamma(z)} \quad (j \ne i). \tag{E.3}$$

Consequently, $1 - s_i^*(\alpha z) \le (N-1)e^{-\alpha\gamma(z)}$. The expert term in Equation (E.2), divided by α, equals $U_i z + O(e^{-\alpha\gamma(z)})$. For the router term, the dominant expert satisfies

$$T_i\alpha z - y^*(\alpha z) = \sum_{j\ne i} s_j^*(\alpha z)(T_i - T_j)\alpha z = O(\alpha e^{-\alpha\gamma(z)}).$$

The other expert residuals are $O(\alpha)$ but their routing weights are exponentially small. Since $v_j^\top(\alpha z) = O(\alpha)$, the entire router term divided by α is $O(\alpha e^{-\alpha\gamma(z)})$. Therefore

$$0 = \frac{J_*(\alpha z)\xi}{\alpha} = U_i z + O(\alpha e^{-\alpha\gamma(z)}) \longrightarrow U_i z. \tag{E.4}$$

The constants in these bounds may depend on the fixed ξ and z, but not on α. Thus U_i vanishes on a nonempty open set and, being linear, is identically zero. This holds for every expert.

Step 3. Eliminate router perturbations. Let P_i be the orthogonal projection onto $\mathrm{range}(T_i)$. Teacher orthogonality gives $P_i T_j = 0$ for $j \ne i$ and $P_i T_i = T_i$. With $U = 0$, projecting Equation (E.2) gives

$$0 = P_i J_*(x)\xi = s_i^*(x)\left[\left(v_i - \sum_j s_j^*(x)v_j\right)^\top x\right] T_i x. \tag{E.5}$$

Every teacher is nonzero, so outside a finite union of measure-zero kernels, $T_i x \neq 0$ for all i. Since the softmax weights are positive, all $v_i^\top x$ must equal the same weighted average. It follows that $(v_i - v_j)^\top x = 0$ almost everywhere, and hence $v_i = v_j$ for all i, j. The centering constraint $\sum_i v_i = 0$ then gives $V = 0$.

We have proved that $\xi^\top H_* \xi = 0$ implies $\xi = 0$. On the finite-dimensional centered parameter space, this establishes $H_* \succ 0$ and the claimed positive, finite extreme eigenvalues. □

E.2 Proof of Lemma D.2

Proof. Fix a target in $\mathcal{M}$ and relabel it as θ_*. For a vector z, write $\widehat{z} = z/\|z\|$ when $z \neq 0$ and $\widehat{z} = 0$ otherwise. This agrees with the zero-denominator convention for $\mathcal{C}$ in Equation (4.3).

Step 1. Reduce cosine to normalized-output error. For $v \neq 0$, the triangle and reverse triangle inequalities give

$$\|\widehat{u} - \widehat{v}\| \leq \min\left\{2, \frac{2\|u - v\|}{\|v\|}\right\}. \tag{E.6}$$

The bound also holds for $u = 0$. Teacher outputs are mutually orthogonal and nonzero almost surely, so

$$|\langle \widehat{W_i x}, \widehat{W_j x}\rangle| \leq \|\widehat{W_i x} - \widehat{T_i x}\| + \|\widehat{W_j x} - \widehat{T_j x}\|.$$

Averaging over expert pairs, each expert appears in $N - 1$ pairs, yielding

$$\mathcal{C}(W) \leq \frac{2}{N} \sum_{i=1}^{N} \mathbb{E}_x \|\widehat{W_i x} - \widehat{T_i x}\|. \tag{E.7}$$

Step 2. Control small teacher outputs. Let b_i be a unit right singular vector for the largest singular value of T_i. Then $\|T_i x\| \geq \|T_i\|_{\mathrm{op}} |b_i^\top x|$. Conditional on any cluster, $b_i^\top x$ is Gaussian with variance σ^2, whose density is at most $1/(\sigma\sqrt{2\pi})$. Therefore, for every $h > 0$,

$$\mathbb{P}(\|T_i x\| < h) \leq \sqrt{\frac{2}{\pi}} \frac{h}{\sigma \|T_i\|_{\mathrm{op}}}. \tag{E.8}$$

The input second moment also satisfies $\mathbb{E}[xx^\top] \preceq (M^2 + \sigma^2) I_d$. Splitting the expectation at $\|T_i x\| = h$, and using Equation (E.6) on the two events, gives

$$\mathbb{E}_x \|\widehat{W_i x} - \widehat{T_i x}\| \leq \frac{2\sqrt{M^2 + \sigma^2}\|W_i - T_i\|_F}{h} + 2\sqrt{\frac{2}{\pi}} \frac{h}{\sigma \|T_i\|_{\mathrm{op}}}. \tag{E.9}$$

Minimizing the right-hand side over $h > 0$ gives

$$\mathbb{E}_x \|\widehat{W_i x} - \widehat{T_i x}\| \leq 4 \left(\frac{2(M^2 + \sigma^2)}{\pi \sigma^2}\right)^{1/4} \|T_i\|_{\mathrm{op}}^{-1/2} \|W_i - T_i\|_F^{1/2}. \tag{E.10}$$

If $W_i = T_i$, the left-hand side is zero and the same bound holds.

Step 3. Aggregate parameter errors. Hölder's inequality with exponents $4/3$ and 4 yields

$$\sum_i \|T_i\|_{\mathrm{op}}^{-1/2} \|W_i - T_i\|_F^{1/2} \leq \left(\sum_i \|T_i\|_{\mathrm{op}}^{-2/3}\right)^{3/4} \left(\sum_i \|W_i - T_i\|_F^2\right)^{1/4}.$$

Combining this with Equation (E.7) gives

$$\mathcal{C}(W) \leq K_0 \left(\sum_i \|W_i - T_i\|_F^2\right)^{1/4} \leq K_0 \sqrt{\|\theta - \theta_*\|}. \tag{E.11}$$

The constant K_0 is invariant under teacher permutations. Choosing the nearest target, and using $\mathcal{C}(W) \leq 1$, proves Equation (D.9). No step requires a teacher rank greater than one. □