Hierarchical Mixture-of-Experts with Two-Stage Optimization

Gleb Molodtsov [*1,2] **Alexander Miasnikov** [*1,2] **Aleksandr Beznosikov** [1,2,3]

Abstract

Sparse Mixture-of-Experts (MoE) models scale capacity by routing tokens to a small subset of experts. Expert grouping improves hardware utilization but introduces a subtle weight-space symmetry: experts within a group are exposed to similar token distributions, driving them toward permutation-equivalent (redundant) solutions. We propose **Hi-MoE**, a hierarchical framework that explicitly breaks this symmetry through two complementary objectives: (i) an *inter-group* balancing term that enforces fair traffic across device-aligned expert groups, and (ii) an *intra-group* diversity term that promotes complementary expert behaviors and prevents within-group collapse. We show that these objectives are interpretable as Lagrange multipliers of a principled balance–specialization constrained optimization problem. Experiments on Swin-MoE (Tiny ImageNet) and OLMoE-7B (58B tokens) show consistent improvements: Hi-MoE-7B achieves a 5.6% perplexity reduction and 40% better expert balance over OLMoE-7B. Our code is available at: `https://github.com/brain-lab-research/Hi-MoE`.

1. Introduction

Mixture-of-Experts (MoE) architectures achieve trillion-parameter scale while maintaining computational efficiency by routing each token to a small subset of expert networks (Shazeer et al., 2017; Fedus et al., 2022; Tang et al., 2025b). Yet uneven expert utilization – routing collapse – remains a critical challenge: a few experts attract most tokens while the rest remain idle, degrading effective capacity and creating device-level load skew (Soboleva, 2025).

A natural remedy is *expert grouping*: partitioning experts into device-aligned groups and enforcing balanced selection within each group (Tang et al., 2025a). Compared to global balancing over all experts, grouped balancing ensures that every group activates experts for each token, equalizing per-device FLOPs and communication. Yet this changes the symmetry structure of the model. Because experts from different groups are co-activated on the same tokens, training naturally encourages inter-group experts to become functionally aligned: each group learns to provide a similar useful computation for the same input. While such cross-group similarity can improve routing efficiency, it also makes diversity within each group more important.

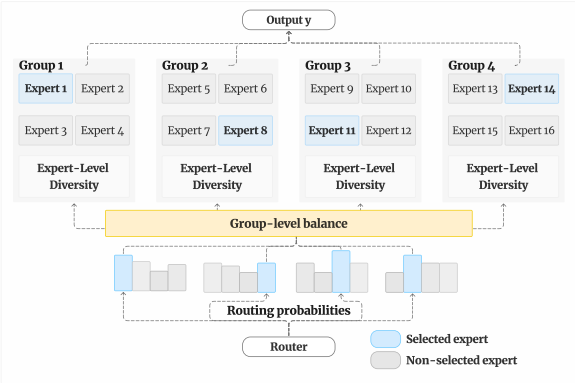


Figure 1. Hi-MoE architecture. Experts are organized into hierarchical groups. Inter-group regularization enforces balanced device-level utilization; intra-group regularization breaks symmetry within groups, promoting complementary specialization.

We propose `Hi-MoE` (Figure 1), a hierarchical framework that makes expert grouping an explicit optimization handle. First, experts selected from different groups for the same token should be sufficiently aligned, so that each group contributes a comparable computation and device utilization remains balanced. Second, experts within the same group should be maximally differentiated, because local balancing otherwise leaves them vulnerable to redundant, permutation-equivalent solutions. Thus, our analysis provides a theoretical foundation that connects weight-space symmetry breaking to both training dynamics (gradient decoupling) and information theory (routing mutual information).

[1]MIRAI, Moscow, Russia [2]BRAIn Lab, Moscow, Russia [3]Innopolis University, Innopolis, Russia. Correspondence to: Gleb Molodtsov <gleb.molodcov@gmail.com>.

Workshop on Weight-Space Symmetries, held in conjunction with the 43^{rd} International Conference on Machine Learning, Seoul, South Korea. 2026.

Contributions.

- We identify group-induced weight-space symmetry as a root cause of expert redundancy in grouped MoE, distinct from (and complementary to) standard routing collapse.
- We propose `Hi-MoE`, a drop-in hierarchical objective with inter-group balancing and intra-group diversity regularizers, compatible with any TopK MoE layer.
- We theoretically show that inter-group and intra-group objectives target complementary aspects of MoE routing. The former controls group-level load imbalance, while the latter reduces routing overlap between experts and supports more input-dependent expert assignments.
- We demonstrate consistent improvement on vision and language tasks, including Swin-MoE model performance on ImageNet, nanoGPT-1B pre-training on OpenWebText, and large-scale pre-training against OLMoE-7B.

2. Related Work

Load balancing in MoE. Standard load-balancing approaches augment the task loss with an auxiliary term $\mathcal{L}_{\text{load}}$ that encourages uniform token dispatch (Lepikhin et al., 2020; Fedus et al., 2022). Subsequent refinements (logit normalization, per-layer coefficients, temperature scaling) improve robustness (Wei et al., 2024; Du et al., 2022; Zoph et al., 2022; Dai et al., 2024). Loss-free methods instead modify the routing mechanism directly so that balance emerges without gradient interference (Wang et al., 2024; Zhou et al., 2022). All these approaches target expert-level collapse but do not address *within-group* symmetry.

Grouped and hardware-aware MoE. System-oriented work partitions experts into device-aligned groups to equalize FLOPs and communication (Han et al., 2025; Tang et al., 2025a; He et al., 2021; 2022). Grouping is effective for hardware efficiency, but as we argue, it induces intra-group symmetry and reduces routing diversity. Our work adds a principled diversity objective that operates precisely inside each group, recovering conditional capacity without sacrificing the systems benefits of grouping.

Hierarchical MoE. Hierarchical structures for mixtures of experts trace back to Jordan & Jacobs (1994) and have been revisited in modern contexts (Fritsch et al., 1996; Nguyen et al., 2024). Our work introduces two-stage routing to the grouped-MoE setting, with explicit symmetry-breaking objectives at each level.

3. Hierarchical Mixture-of-Experts

Setup. We consider a Transformer MoE layer with N expert networks $\{f_i\}_{i=1}^N$, $f_i : \mathbb{R}^d \to \mathbb{R}^d$. Experts are partitioned into M disjoint groups $\mathcal{G}_1, \dots, \mathcal{G}_M$ (aligned with devices). The router produces logits $s(\mathbf{x}) \in \mathbb{R}^N$ and probabilities $p(\mathbf{x}) = \mathrm{softmax}(s(\mathbf{x}))$. For each group m, K_m experts are selected via a group-local TopK_m operator (typically $K_m = K/M$):

$$\mathrm{MoE}(\mathbf{x}) = \sum_{m=1}^{M} \sum_{i \in \mathcal{G}_m} \big(p_i(\mathbf{x}) \cdot \mathbb{I}_{\{i \in \mathrm{TopK}_m(\mathbf{p}_{\mathcal{G}_m}(\mathbf{x}))\}}\big) f_i(\mathbf{x}).$$

Routing factorization. The grouped structure induces a natural factorization $p(e \mid \mathbf{x}) = p(g \mid \mathbf{x})\, p(e \mid \mathbf{x}, g)$, where $p(g \mid \mathbf{x}) = \sum_{e \in \mathcal{G}_g} p(e \mid \mathbf{x})$ is the induced group mass. Expanding the ℓ_2 concentration of the global routing distribution yields

$$\|\mathbf{p}(\mathbf{x})\|_2^2 \;=\; \sum\nolimits_{g=1}^{M} p(g \mid \mathbf{x})^2 \, \|\mathbf{p}(\cdot \mid \mathbf{x}, g)\|_2^2, \qquad (1)$$

which separates inter-group concentration from intra-group concentration. This decomposition motivates addressing balance and diversity at separate levels.

3.1. Inter-group regularization

Device utilization depends on how routing mass is distributed across groups. We regularize the *post-selection* routing weights,

$$\widetilde{\boldsymbol{\pi}}(\mathbf{x}) \;:=\; \bigoplus_{m=1}^{M} \big(\mathbf{p}_{\mathcal{G}_m}(\mathbf{x}) \odot \mathbf{1}_{\mathrm{TopK}_m(\mathbf{p}_{\mathcal{G}_m}(\mathbf{x}))}\big), \qquad (2)$$

focusing the signal on experts that incur real compute and communication. The inter-group regularizer is

$$\mathcal{R}_{\text{inter}} \;=\; \lambda_{\text{inter}} \left\|\widetilde{\boldsymbol{\pi}}(\mathbf{x})\right\|_2^2. \qquad (3)$$

Minimizing $\|\widetilde{\boldsymbol{\pi}}\|_2^2$ discourages concentration among selected experts and, through (1), reduces group-level imbalance. This consistently lowers group load variance and improves uniform GPU work per step.

However, inter-group equalization has a side effect: it restricts routing freedom and can make experts in different groups functionally redundant. Independent groups trained under similar token streams learn similar decision boundaries – the very weight-space symmetry we seek to break.

3.2. Intra-group regularization

To recover conditional capacity, we introduce an anti-concentration regularizer on the pre-selection routing distribution:

$$\mathcal{R}_{\text{intra}} \;=\; -\,\lambda_{\text{intra}} \left\|\boldsymbol{\pi}(\mathbf{x})\right\|_2^2.$$

Maximizing $\|\boldsymbol{\pi}\|_2^2$ encourages the router to assign decisive, complementary weights – making expert assignments more

informative about the input – thereby breaking the permutation symmetry that grouping induces. We pair this with a bias-corrected routing computation to prevent degenerate within-group dominance:

$$\boldsymbol{\pi}(\mathbf{x}) \;=\; \mathrm{softmax}\!\left(\frac{\mathbf{g}(\mathbf{x}) - \tau\,\overline{\mathbf{g}}}{T}\right),\; \overline{\mathbf{g}} \leftarrow \beta\overline{\mathbf{g}} + (1-\beta)\mathbf{g}(\mathbf{x}),$$

where $\overline{\mathbf{g}}$ is an exponential moving average of router logits. This nudges routing away from entrenched modes without an additional balancing loss.

3.3. Complete objective

The `Hi-MoE` training objective combines task learning, standard load balancing (Lepikhin et al., 2020), and the proposed hierarchical regularizers:

$$\mathcal{L} \;=\; \mathcal{L}_{\text{task}} + \mathcal{L}_{\text{load}} + \mathcal{R}_{\text{intra}} + \mathcal{R}_{\text{inter}}. \tag{4}$$

We quantify load balance using the coefficient of variation $\mathrm{CV} = \sigma_{\text{load}}/\mu_{\text{load}}$ of token counts per expert; lower CV indicates more uniform utilization.

4. Analysis

We show that the hierarchical regularizers implement a principled *constrained* view of MoE training: maximize specialization subject to balance. Throughout, $\widetilde{\boldsymbol{\pi}}(\mathbf{x})$ denotes the post-Top-K weights (2), $r_g(\mathbf{x}) := (\sum_{e\in\mathcal{G}_g} \widetilde{\pi}_e)/(\sum_e \widetilde{\pi}_e)$ the normalized group assignment, and $S_{\max} = \max_g |\mathcal{G}_g|$ the maximum group size.

4.1. Inter-group regularizer bounds group imbalance

Theorem 4.1 (Inter-group regularizer controls group CV)**.** *Let $\bar{r} := \mathbb{E}[r(\mathbf{X})] \in \Delta^{M-1}$ be the mean group-load distribution. Then*

$$\|\bar{r}\|_2^2 \;\le\; \mathbb{E}[\|r(\mathbf{X})\|_2^2] \;\le\; S_{\max}\,\mathbb{E}[\|\widetilde{\boldsymbol{\pi}}(\mathbf{X})\|_2^2].$$

Consequently, the group-level coefficient of variation satisfies

$$\mathrm{CV}_{\text{group}}^2 + 1 \;=\; M\|\bar{r}\|_2^2 \;\le\; M S_{\max}\,\mathbb{E}[\|\widetilde{\boldsymbol{\pi}}(\mathbf{X})\|_2^2].$$

Discussion. Since $\mathcal{R}_{\text{inter}} \propto \|\widetilde{\boldsymbol{\pi}}\|_2^2$, Theorem 4.1 provides a *certified* link: minimizing $\mathcal{R}_{\text{inter}}$ directly upper-bounds the group-level CV. This formalizes why $\mathcal{R}_{\text{inter}}$ is not an ad-hoc trick.

4.2. Intra-group regularizer decouples expert gradients

Theorem 4.2 (Overlap bounds cross-expert gradient coupling)**.** *Assume $\|g_i(\mathbf{x})\|_2 \le G$ for all $i, \mathbf{x}$. Define cross-expert coupling $C(\mathbf{x}) := \sum_{i\neq j} |\langle \pi_i g_i, \pi_j g_j\rangle|$. Then*

$$C(\mathbf{x}) \;\le\; G^2\big(1 - \|\boldsymbol{\pi}(\mathbf{x})\|_2^2\big).$$

Maximizing $\|\boldsymbol{\pi}\|_2^2$ (i.e., applying $\mathcal{R}_{intra}$) provably decreases an upper bound on how strongly different experts are updated on the same token.

Discussion. Anti-overlap reduces gradient alignment between experts, increasing the probability that experts follow different optimization trajectories and learn complementary functions – directly breaking the symmetry that grouping induces.

4.3. Intra-group regularizer maximizes routing mutual information

Theorem 4.3 (Balanced marginals $\Rightarrow$ $\|\pi\|_2^2$ maximizes collision MI)**.** *Let $p(i) := \mathbb{E}[\pi_i(\mathbf{X})]$ be marginal expert usage. If $p(i) = 1/N$ for all i, then the collision mutual information $I_2(\mathbf{X}; E) = H_2(E) - H_2(E \mid \mathbf{X})$ satisfies*

$$I_2(\mathbf{X}; E) \;=\; \log\big(N\,\mathbb{E}[\|\boldsymbol{\pi}(\mathbf{X})\|_2^2]\big),$$

which is strictly increasing in $\mathbb{E}[\|\boldsymbol{\pi}(\mathbf{X})\|_2^2]$.

Corollary 4.4. *Under balanced marginals, $\mathcal{R}_{intra}$ strictly increases $I_2(\mathbf{X}; E)$, i.e., increases routing specialization.*

Lagrangian interpretation. Define $\mathcal{C}_{\text{sys}} := \mathbb{E}[\|\widetilde{\boldsymbol{\pi}}\|_2^2]$ (group imbalance proxy) and $\mathcal{C}_{\text{ov}} := \mathbb{E}[1 - \|\boldsymbol{\pi}\|_2^2]$ (routing overlap). The constrained problem $\min\ \mathcal{L}_{\text{task}} + \mathcal{L}_{\text{load}}$ s.t. $\mathcal{C}_{\text{sys}} \le \varepsilon_{\text{sys}},\ \mathcal{C}_{\text{ov}} \le \varepsilon_{\text{ov}}$ has Lagrangian equivalent to (4). Thus $\lambda_{\text{inter}}, \lambda_{\text{intra}}$ are interpretable Lagrange multipliers: `Hi-MoE` maximizes specialization subject to hardware-aligned balance.

Proofs are provided in Appendix F. Full notation is summarized in Table 8 (Appendix E).

5. Experiments

We evaluate `Hi-MoE` in three pre-training settings: Swin-MoE-1B on Tiny ImageNet, nanoGPT (Karpathy, 2022) on OpenWebText and OLMoE-7B on the Dolma mixture (58B tokens). In all cases we organize experts into $M = 4$ hierarchical groups. We compare against Vanilla GShard-style MoE (Lepikhin et al., 2020), loss-free load balancing (Wang et al., 2024), and MoGE (Tang et al., 2025a).

5.1. Swin-MoE on Tiny ImageNet

We replace FFN layers of Swin Transformer (Liu et al., 2021) with Hi-MoE modules (≈1B total / 83M active parameters), training for 200 epochs with 4 groups of 16 experts, selecting 1 per group.

`Hi-MoE` outperforms all baselines in both accuracy and load balance, with Figure 2 showing the most uniform expert activation. Unlike MoGE, which suffers from intra-group

Table 1. Tiny ImageNet classification. Mean±std over 3 seeds.

Method	Acc@1 (%)	Acc@5 (%)	CV
Vanilla GShard	58.61 ± 0.24	80.39 ± 0.21	0.33 ± 0.02
Loss-free bal.	58.87 ± 0.29	79.65 ± 0.26	0.31 ± 0.02
MoGE	58.61 ± 0.27	80.92 ± 0.23	0.30 ± 0.02
Hi-MoE	$\mathbf{59.13 \pm 0.19}$	$\mathbf{81.18 \pm 0.17}$	$\mathbf{0.29 \pm 0.01}$

collapse despite balanced group usage, Hi-MoE avoids expert domination. Qualitative attention maps further show that its expert groups specialize in complementary visual features, indicating effective weight-space symmetry breaking through intra-group diversity regularization.

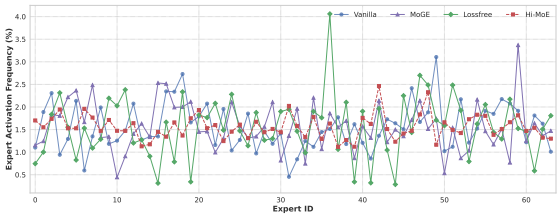


Figure 2. Expert activation frequency in Swin-MoE layer 7. Hi-MoE achieves the most uniform distribution; MoGE shows within-group collapse despite good inter-group balance.

5.2. nanoGPT on OpenWebText

We place Hi-MoE in 12 MoE layers of a 24-layer Transformer, yielding a 1.06B-parameter model with 507M active parameters per token, and train it for 49B tokens. We begin with the quality metrics presented in Table 2.

Table 2. Comparison across nanoGPT architectures. Metrics are reported as mean±std over 3 runs.

Method	OWT PPL	CV
Vanilla GShard-style	2.985 ± 0.024	0.31 ± 0.016
Loss-free balancing	2.979 ± 0.027	0.37 ± 0.017
ST-MoE	2.981 ± 0.022	0.33 ± 0.016
MoGE	2.977 ± 0.025	0.25 ± 0.018
Hi-MoE	$\mathbf{2.947 \pm 0.019}$	$\mathbf{0.23 \pm 0.013}$

To analyze load balancing in depth, we examine the expert activation patterns across network layers. Activation maps in Figure 3 show that MoGE can appear balanced across groups while remaining imbalanced within them, with one expert dominating its pair. GShard-style routing reduces some local imbalance but remains uneven across devices, while loss-free routing shows severe underuse and sharp spikes. Hi-MoE achieves near-uniform expert utilization.

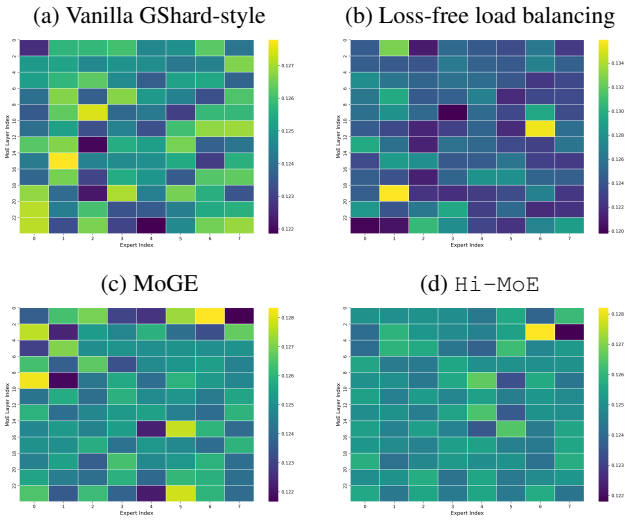


Figure 3. Expert activation heatmaps across all 12 MoE layers during nanoGPT-1B training. Subplots show different layers, with experts on the x-axis and training steps on the y-axis.

5.3. Large-Scale Validation: OLMoE-7B

We pre-train both a standard OLMoE-7B and Hi-MoE-7B for 2 epochs on 58B tokens from the Dolma mixture. Both use 64 experts and TopK=8 routing; Hi-MoE-7B structures experts into 4 groups with 2 selected per group.

Table 3. Perplexity comparison across 11 validation domains.

Domain	OLMoE-7B	Hi-MoE-7B	Δ(%)
wikitext_103	32.997	**31.381**	**-4.9**
c4_en	39.852	**37.345**	**-6.3**
dolma_books	39.704	**36.674**	**-7.6**
dolma_common-crawl	42.722	**39.872**	**-6.7**
dolma_pes2o	11.685	**11.301**	**-3.3**
dolma_reddit	47.023	**43.802**	**-6.8**
dolma_stack	3.811	**3.765**	**-1.2**
dolma_wiki	31.831	**30.008**	**-5.7**
ice	**23.819**	23.888	+0.3
m2d2_s2orc	23.286	**21.803**	**-6.4**
pile	13.364	**12.978**	**-2.9**
Average			**-5.6**

Hi-MoE-7B yields consistent perplexity improvements across nearly all domains, with gains increasing with scale.

Discussion

Hi-MoE shows that expert grouping induces a weight-space symmetry that, left unaddressed, limits the effective capacity of MoE models. Explicit symmetry breaking through hierarchical regularization turns this structural constraint into a reliable quality and efficiency gain.

Acknowledgments

The work was supported by the Ministry of Economic Development of the Russian Federation (agreement No. 139-15-2025-013, dated June 20, 2025, IGK 000000C313925P4B0002).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Dai, D., Deng, C., Zhao, C., Xu, R., Gao, H., Chen, D., Li, J., Zeng, W., Yu, X., Wu, Y., et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. *arXiv preprint arXiv:2401.06066*, 2024.

de Mathelin, A., Deheeger, F., Mougeot, M., and Vayatis, N. Deep anti-regularized ensembles provide reliable out-of-distribution uncertainty quantification. *arXiv preprint arXiv:2304.04042*, 2023.

Du, N., Huang, Y., Dai, A. M., Tong, S., Lepikhin, D., Xu, Y., Krikun, M., Zhou, Y., Yu, A. W., Firat, O., et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International conference on machine learning*, pp. 5547–5569. PMLR, 2022.

Fedus, W., Zoph, B., and Shazeer, N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. volume 23, pp. 1–39, 2022.

Fritsch, J., Finke, M., and Waibel, A. Adaptively growing hierarchical mixtures of experts. *Advances in Neural Information Processing Systems*, 9, 1996.

Han, Y., Pan, L., Peng, J., Tao, Z., Zhang, W., and Zhang, Y. Grace-moe: Grouping and replication with locality-aware routing for efficient distributed moe inference. *arXiv preprint arXiv:2509.25041*, 2025.

He, J., Qiu, J., Zeng, A., Yang, Z., Zhai, J., and Tang, J. Fastmoe: A fast mixture-of-expert training system. *arXiv preprint arXiv:2103.13262*, 2021.

He, J., Zhai, J., Antunes, T., Wang, H., Luo, F., Shi, S., and Li, Q. Fastermoe: modeling and optimizing training of large-scale dynamic pre-trained models. In *Proceedings of the 27th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, pp. 120–134, 2022.

Jordan, M. I. and Jacobs, R. A. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2): 181–214, 1994.

Karpathy, A. NanoGPT. https://github.com/karpathy/nanoGPT, 2022.

Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N., and Chen, Z. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*, 2020.

Liu, Y. and Yao, X. Ensemble learning via negative correlation. *Neural networks*, 12(10):1399–1404, 1999.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Nguyen, H., Han, X., Harris, C., Saria, S., and Ho, N. On expert estimation in hierarchical mixture of experts: Beyond softmax gating functions. *arXiv preprint arXiv:2410.02935*, 2024.

Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., and Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

Soboleva, D. Router wars: Which moe routing strategy actually works, 2025. URL https://www.cerebras.ai/blog/moe-guide-router.

Tang, Y., Li, X., Liu, F., Guo, W., Zhou, H., Wang, Y., Han, K., Yu, X., Li, J., Zang, H., et al. Pangu pro moe: Mixture of grouped experts for efficient sparsity. *arXiv preprint arXiv:2505.21411*, 2025a.

Tang, Y., Yin, Y., Wang, Y., Zhou, H., Pan, Y., Guo, W., Zhang, Z., Rang, M., Liu, F., Zhang, N., et al. Pangu ultra moe: How to train your big moe on ascend npus. *arXiv preprint arXiv:2505.04519*, 2025b.

Wang, L., Gao, H., Zhao, C., Sun, X., and Dai, D. Auxiliary-loss-free load balancing strategy for mixture-of-experts. *arXiv preprint arXiv:2408.15664*, 2024.

Wei, T., Zhu, B., Zhao, L., Cheng, C., Li, B., Lü, W., Cheng, P., Zhang, J., Zhang, X., Zeng, L., et al. Skywork-moe: A deep dive into training techniques for mixture-of-experts language models. *arXiv preprint arXiv:2406.06563*, 2024.

Zhou, Y., Lei, T., Liu, H., Du, N., Huang, Y., Zhao, V., Dai, A., Chen, Z., Le, Q., and Dean, J. Mixture-of-experts with expert choice routing. In *Advances in Neural Information Processing Systems*, pp. 7103–7114, 2022.

Zoph, B., Bello, I., Kumar, S., Du, N., Huang, Y., Dean, J., Shazeer, N., and Fedus, W. St-moe: Designing stable and transferable sparse expert models. *arXiv preprint arXiv:2202.08906*, 2022.

A. Expert Specialization Maps

Figure 4 shows group-level attention maps from Swin-Hi-MoE trained on Tiny ImageNet. Despite fixed group partitions, different groups attend to distinct visual features, which is consistent with increased functional specialization.

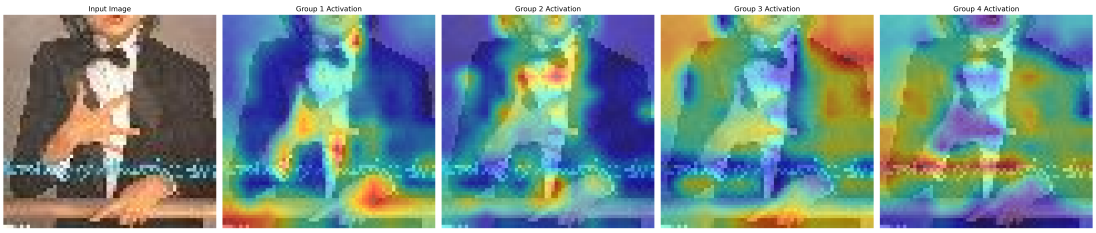


Figure 4. Group-level attention patterns of Swin-Hi-MoE on Tiny ImageNet. Group 1 (hands/rails), Group 2 (light-colored objects, e.g., white shirt), Group 3 (dark-colored objects, e.g., suit), Group 4 (body parts).

B. Experimental Reproducibility Checklist

B.1. Hardware

All experiments were run on a node with $4\times$ NVIDIA H200 GPUs. As an end-to-end reference point, in our setup pre-training Swin Transformer-1B takes 24 GPU-hours, nanoGPT-1B takes 372 GPU-hours, and `Hi-MoE`-7B takes approximately 632 GPU-hours.

B.2. `Hi-MoE` fitting hyperparameters

To ensure reproducibility and simplify adoption, we summarize the main architectural and optimization hyperparameters used to fit `Hi-MoE` across our experimental setups. Table 4 reports the configuration alongside model-specific settings for nanoGPT, Swin-MoE, and `Hi-MoE`-7B.

Table 4. `Hi-MoE` hyperparameters for different setups.

Hyperparameter	**nanoGPT**	**Swin-MoE**	**`Hi-MoE`-7B**
Hidden dimension (d)	1,024	96	2,048
Attention heads	16	[3, 6, 12, 24]	16
Layers	24	[2, 2, 18, 2]	16
Context / image size	1,024 tokens	192×192	4,096 tokens
Total experts (N)	8	64	64
Expert groups (M)	4	4	4
Top-k per group	1	1	2
Active experts per token	4	4	8
Expert FFN dimension	$4 \times 1{,}024 = 4{,}096$	$4 \times d_{\text{stage}}$	1,024
α ($\mathcal{L}_{\text{load}}$ coefficient)	$0.01^{\dagger}$	$0.01^{\dagger}$	$0.0001^{\dagger}$
T	1.0^{*}	1.0^{*}	1.0
λ_{intra}	0.1^{*}	0.1^{*}	0.1
λ_{inter}	0.05^{*}	0.05^{*}	0.05
τ	0.01^{*}	0.01^{*}	0.01
β	0.9^{*}	0.9^{*}	0.9
β_1 (AdamW)	$0.9^{\dagger}$	$0.9^{\dagger}$	$0.9^{\dagger}$
β_2 (AdamW)	$0.95^{\dagger}$	$0.999^{\dagger}$	$0.95^{\dagger}$

*Tuned over: $T \in \{0.5, 0.75, 1.0, 1.25, 1.5\}$, $\lambda \in \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 0.05, 0.1\}$, $\tau \in \{1, 0.1, 0.01, 0.001, 0.0001\}$, $\beta \in \{0.1, 0.3, 0.5, 0.7, 0.8, 0.9, 0.95, 0.99, 0.999\}$. For `Hi-MoE`-7B, these parameters were not tuned due to model scale and computational constraints.
†Taken from original codebase configurations.

Note that the hyperparameters for `Hi-MoE`-7B were either taken directly from the original OLMoE-7B configuration or transferred from the nanoGPT setup.

GPU utilization. For system-level efficiency, one can adopt the same distributed training pipeline as Pangu MoE (Tang et al., 2025a), combining tensor, expert, pipeline (including virtual pipeline), and context parallelism. While (Tang et al., 2025a) also highlights engineering-driven throughput gains under such setups, our work intentionally focuses on the methodological side – improving routing behavior via hierarchical optimization. Importantly, `Hi-MoE` is fully compatible with these parallelism pipelines and, by enforcing more uniform expert utilization, can translate into higher effective GPU utilization when deployed in the same system regime.

C. nanoGPT-1B on OpenWebText

Component-wise ablation. To isolate the contribution of each hierarchical component, Table 5 reports an ablation on nanoGPT-1B. The pattern is consistent: $\mathcal{R}_{\text{inter}}$ primarily improves balance, $\mathcal{R}_{\text{intra}}$ primarily improves quality, and the bias correction acts as a stabilizer rather than the source of the gains.

Table 5. Component-wise ablation of `Hi-MoE` on nanoGPT-1B.

Config	PPL	CV
MoGE	2.977	0.25
+ $\mathcal{R}_{\text{intra}}$	2.960	0.27
+ $\mathcal{R}_{\text{inter}}$	2.972	0.21
+ bias correction	2.976	0.25
+ $\mathcal{R}_{\text{intra}} + \mathcal{R}_{\text{inter}}$	2.950	0.23
`Hi-MoE`	**2.947**	**0.23**

Load balancing. `Hi-MoE` produces an almost uniform expert-usage pattern achieving better results than baselines. Figure 5 confirms these observations by showing the final expert activation frequencies in the 5th MoE layer.

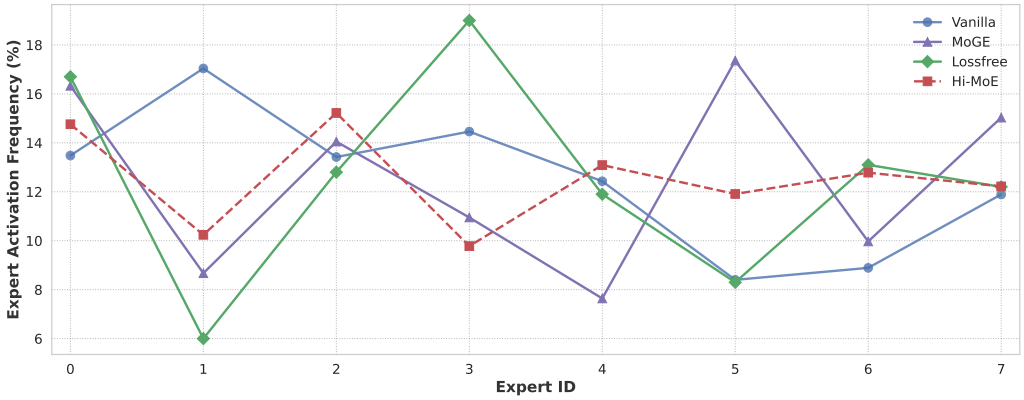
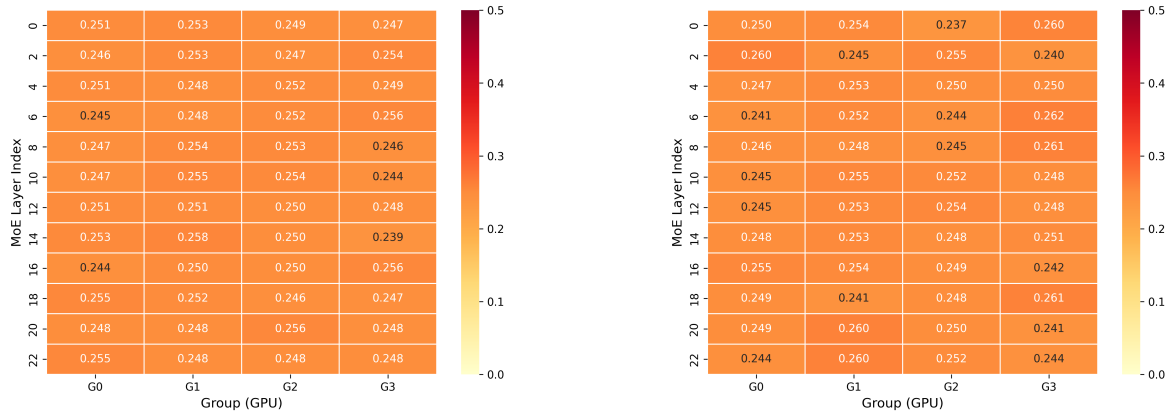


Figure 5. Expert activation frequency distribution in the 5th MoE layer of nanoGPT-1B after training.

Next, we examine the per-layer group utilization. MoGE and `Hi-MoE` achieve uniform group load by selecting an equal number of experts from each group. In Figure 6, we compare group utilization for Vanilla MoE and the loss-free routing strategy.

Although the aggregate group workload appears nearly uniform, this is misleading at the per-token level. Because Vanilla and loss-free routing do not constrain the number of groups activated per token, individual tokens tend to concentrate their selected experts on a subset of groups. On average each token activates experts from *only* 3.142 *out of* 4 *groups* (GPUs) under Vanilla routing and 3.151 under loss-free routing. Thus, despite balanced total workloads, roughly one GPU per token *receives no computation*, directly reducing device utilization.



(a) Vanilla GShard style

(b) `Loss-free load balancing`

Figure 6. Group workload across layers.

D. OLMoE-7B Downstream Tasks

Beyond perplexity metrics, we evaluate both models on downstream tasks. Table 6 reports performance on common sense reasoning, question answering, and knowledge benchmarks. All tasks are evaluated with `lm-evaluation-harness`; unless stated otherwise, results are 0-shot, while MMLU is 5-shot. We report either accuracy (`acc`) or length-normalized accuracy (`len_norm`), following the benchmark default.

Table 6. Downstream task performance comparison between OLMoE-7B and `Hi-MoE-7B`.

Task	**OLMoE-7B**	**`Hi-MoE-7B`**	**Δ (%)**
Common Sense Reasoning			
COPA (acc)	53.0	**57.0**	**+7.5**
Winogrande (acc)	49.1	**50.8**	**+3.5**
CommonsenseQA (len_norm)	**27.4**	26.6	-2.9
Social IQA (len_norm)	39.2	**39.4**	**+0.5**
Question Answering			
ARC-Easy (acc)	41.4	**41.9**	**+1.2**
ARC-Challenge (len_norm)	**22.7**	22.1	-2.6
OpenBookQA (len_norm)	25.2	**26.8**	**+6.3**
General Knowledge			
PIQA (len_norm)	**58.1**	57.5	-1.0
HellaSwag (len_norm)	30.5	**31.3**	**+2.6**
MMLU (5-shot MC, avg)			
STEM (acc)	21.2	**25.9**	**+22.2**
Humanities (acc)	23.6	**26.4**	**+11.9**
Social Sciences (acc)	24.6	**25.8**	**+4.9**

Moreover, it achieves 40% better balance. Thus, hierarchical symmetry breaking improves both utilization and quality.

Table 7. Expert balance: `Hi-MoE-7B` improves CV by 40%.

Metric	**OLMoE-7B**	**Hi-MoE-7B**
Expert Balance (CV)	0.257	**0.153**

E. Notation

Table 8. Notation used throughout the paper.

Symbol	Meaning
$C(\mathbf{x})$	total (soft) cross-expert gradient coupling on token $\mathbf{x}$
d	token/hidden dimension (so $\mathbf{x} \in \mathbb{R}^d$)
$E \sim \pi(\mathbf{X})$	stochastic expert identity under router π
$f_i : \mathbb{R}^d \to \mathbb{R}^d$	expert network i
$g : \mathbb{R}^d \to \mathbb{R}^N$	router / gating network
$g_i(\mathbf{x})$	expert-local gradient on token $\mathbf{x}$ (w.r.t. θ_i)
G	uniform bound with $\|g_i(\mathbf{x})\|_2 \le G$
$H_2(E)$	collision entropy of E
$I_2(\mathbf{X}; E)$	collision mutual information between token and expert
K	Top-K sparsity (number of selected experts)
K_m	Top-K within group m
L	(soft) group load vector for a batch
L_g	load assigned to group g
M	number of expert groups
N	number of experts
$p(e \mid \mathbf{x})$	router distribution over experts (generic notation)
$p(g \mid \mathbf{x})$	induced group mass for token $\mathbf{x}$
$p(i)$	marginal usage probability of expert i
$r(\mathbf{x}) \in \Delta^{M-1}$	normalized group assignment induced by $\widetilde{\boldsymbol{\pi}}(\mathbf{x})$
$\bar{r}$	mean group-load distribution $\mathbb{E}[r(\mathbf{X})]$
$S_{\max}$	maximum group size $\max_{g \in [M]} \lvert\mathcal{G}_g\rvert$
T	softmax temperature in bias-corrected router
$\mathcal{B} = \{\mathbf{x}_b\}_{b=1}^B$	batch of B token representations
$\mathcal{C}_{\text{ov}}$	proxy cost for routing overlap
$\mathcal{C}_{\text{sys}}$	proxy cost for group/GPU imbalance
Δ^{n-1}	probability simplex in $\mathbb{R}^n$
$\mathbb{I}\{\cdot\}$	indicator function
$\mathcal{G}_g$	index set of experts in group g
$\mathcal{L}$	total training objective
$\mathcal{L}_{\text{load}}$	standard auxiliary load-balancing loss
$\mathcal{L}_{\text{task}}$	task loss
$\mathcal{R}_{\text{inter}}$	inter-group regularizer ($\propto \|\widetilde{\boldsymbol{\pi}}\|_2^2$)
$\mathcal{R}_{\text{intra}}$	intra-group regularizer ($\propto -\|\boldsymbol{\pi}\|_2^2$)
$\boldsymbol{\pi}(\mathbf{x}) \in \Delta^{N-1}$	router probabilities over experts for token $\mathbf{x}$
$\pi_i(\mathbf{x})$	i-th component of $\boldsymbol{\pi}(\mathbf{x})$
$\widetilde{\boldsymbol{\pi}}(\mathbf{x})$	post-Top-K routing weights
$\mathbf{x}, \mathbf{X}$	token representation (deterministic / random variable)
$\overline{\mathbf{g}}$	EMA of router logits
$\theta_i \in \mathbb{R}^P$	parameters of expert i
P	number of parameters per expert
λ_{inter}	weight of $\mathcal{R}_{\text{inter}}$
λ_{intra}	weight of $\mathcal{R}_{\text{intra}}$
μ	mean of $\{L_g\}_{g=1}^M$
μ_{load}	mean of token counts per expert
σ^2	variance of $\{L_g\}_{g=1}^M$
σ_{load}	std of token counts per expert
τ	bias-correction strength in router logits
$\widehat{L} \in \Delta^{M-1}$	normalized load distribution $\widehat{L}_g = L_g / \sum_h L_h$
$\mathrm{CV}(L)$	coefficient of variation of L: σ/μ
ε_{ov}	overlap constraint level
ε_{sys}	system-imbalance constraint level

F. Supplementary Analysis

In this appendix, we provide proofs for all theoretical statements from Section 4.

Recall that M denotes the number of expert groups (e.g., GPU-aligned shards). Consider any nonnegative group load vector $L = (L_1, \dots, L_M) \in \mathbb{R}_+^M$ computed on a batch (hard counts or soft mass, both are admissible here). Let $\mu := \frac{1}{M}\sum_{g=1}^M L_g$ and $\sigma^2 := \frac{1}{M}\sum_{g=1}^M (L_g - \mu)^2$, and define $\mathrm{CV}(L) := \sigma/\mu$ as in (4). Define the *normalized* load distribution $\widehat{L} \in \Delta^{M-1}$ by $\widehat{L}_g := L_g / \sum_{h=1}^M L_h$.

Proposition F.1 (CV–ℓ_2 equivalence)**.** *For any $L \in \mathbb{R}_+^M$ with $\sum_g L_g > 0$,*

$$\mathrm{CV}(L)^2 \;=\; M\big\|\widehat{L}\big\|_2^2 \;-\; 1.$$

In particular, minimizing $\mathrm{CV}(L)$ is equivalent to minimizing $\|\widehat{L}\|_2^2$.

Proof. Let $S := \sum_{g=1}^M L_g$, so $\mu = S/M$ and $L_g - \mu = S(\widehat{L}_g - 1/M)$. Then

$$\sigma^2 = \frac{1}{M}\sum_{g=1}^M S^2\Big(\widehat{L}_g - \frac{1}{M}\Big)^2 = \frac{S^2}{M}\sum_{g=1}^M\Big(\widehat{L}_g^2 - \frac{2}{M}\widehat{L}_g + \frac{1}{M^2}\Big).$$

Since $\sum_g \widehat{L}_g = 1$, we obtain $\sum_g(\widehat{L}_g - \frac{1}{M})^2 = \|\widehat{L}\|_2^2 - \frac{1}{M}$, hence $\sigma^2 = \frac{S^2}{M}(\|\widehat{L}\|_2^2 - \frac{1}{M})$ and $\mathrm{CV}(L)^2 = M\|\widehat{L}\|_2^2 - 1$. □

Proposition F.1 shows that *balance is exactly an ℓ_2 concentration objective*. Hence any surrogate that provably controls an ℓ_2-concentration of group loads is directly tied to CV.

F.1. Inter-group regularizer: additional proofs

We now connect the inter-group regularizer $\mathcal{R}_{\text{inter}}$ in (3) to group imbalance. For a token representation $\mathbf{x}$, let $\widetilde{\boldsymbol{\pi}}(\mathbf{x}) \in \mathbb{R}_+^N$ be the post-selection (sparsified) weights defined in (2). Define the induced *group-mass* vector $r(\mathbf{x}) \in \mathbb{R}_+^M$ by

$$r_g(\mathbf{x}) \;:=\; \sum_{e \in \mathcal{G}_g} \widetilde{\pi}_e(\mathbf{x}), \qquad g = 1, \dots, M.$$

When $\widetilde{\boldsymbol{\pi}}(\mathbf{x})$ is normalized to sum to one on the selected support, $r(\mathbf{x})$ is a distribution over groups and $\sum_g r_g(\mathbf{x}) = 1$. $S_{\max} := \max_g |\mathcal{G}_g|$ is the largest group size.

Lemma F.2 (Group-sum ℓ_2 bound)**.** *For any nonnegative $w \in \mathbb{R}_+^N$ and group partition $\{\mathcal{G}_g\}_{g=1}^M$, let $m_g := \sum_{e\in\mathcal{G}_g} w_e$. Then $\|m\|_2^2 \;\le\; S_{\max}\,\|w\|_2^2$.*

Proof. Fix a group g. By Cauchy–Schwarz, $m_g^2 = (\sum_{e\in\mathcal{G}_g} w_e)^2 \le |\mathcal{G}_g| \sum_{e\in\mathcal{G}_g} w_e^2 \le S_{\max}\sum_{e\in\mathcal{G}_g} w_e^2$. Summing over g: $\sum_g m_g^2 \le S_{\max}\|w\|_2^2$. □

Proof of Theorem 4.1. The first inequality is Jensen's inequality for the convex function $v \mapsto \|v\|_2^2$. The second inequality is Lemma F.2 applied pointwise with $w = \widetilde{\boldsymbol{\pi}}(\mathbf{X})$ and $m = r(\mathbf{X})$. The CV bound follows from Proposition F.1. □

F.2. Intra-group anti-regularizer: additional proofs

If balancing pressure drives routing toward token-independent assignments, experts tend to be trained on essentially the same signal, yielding redundant solutions. In contrast, `Hi-MoE` adds $\mathcal{R}_{\text{intra}} = -\lambda_{\text{intra}}\|\boldsymbol{\pi}(\mathbf{x})\|_2^2$. Note that on the probability simplex, increasing $\|\boldsymbol{\pi}\|_2^2$ makes routing *more decisive* (peaked), which breaks symmetry and prevents experts from receiving identical mixtures. For any probability vector $\pi \in \Delta^{N-1}$,

$$\sum_{i\neq j}\pi_i\pi_j \;=\; 1 - \|\pi\|_2^2. \tag{5}$$

Equivalently, if $E_1, E_2 \overset{\text{i.i.d.}}{\sim} \pi$, then $\mathbb{P}(E_1 \neq E_2) = 1 - \|\pi\|_2^2$. Thus, $\mathcal{R}_{\text{intra}}$ is a principled anti-overlap regularizer: it explicitly reduces how much two different experts share the same token in expectation, aligned with negative correlation learning and modern anti-regularization ideas (Liu & Yao, 1999; de Mathelin et al., 2023).

Proof of Theorem 4.2. By Cauchy–Schwarz and the gradient bound, $|\langle \pi_i g_i, \pi_j g_j \rangle| \le \pi_i \pi_j \|g_i\|_2 \|g_j\|_2 \le G^2 \pi_i \pi_j$. Summing over $i \neq j$ and applying Equation (5) gives the result. □

To formalize coverage of experts, we use a standard specialization proxy: how informative the expert identity is about the input token. Consider stochastic routing where $E \sim \pi(\mathbf{X})$ given a random token $\mathbf{X}$. A quantity tightly connected to $\|\pi\|_2^2$ is the collision conditional entropy which decreases when routing becomes more decisive:

$$H_2(E \mid \mathbf{X}) \;:=\; -\log \mathbb{E}_{\mathbf{X}}\Big[\sum_{i=1}^{N} \pi_i(\mathbf{X})^2\Big] \;=\; -\log \mathbb{E}_{\mathbf{X}}\big[\|\pi(\mathbf{X})\|_2^2\big].$$

Proof of Theorem 4.3. By definition,

$$I_2(\mathbf{X}; E) = -\log \sum_i p(i)^2 + \log \mathbb{E}_{\mathbf{X}}\Big[\sum_i \pi_i(\mathbf{X})^2\Big] = \log\Big(\frac{\mathbb{E}[\|\pi(\mathbf{X})\|_2^2]}{\sum_i p(i)^2}\Big).$$

If $p(i) = 1/N$, then $\sum_i p(i)^2 = 1/N$, hence $I_2(\mathbf{X}; E) = \log(N\, \mathbb{E}[\|\pi(\mathbf{X})\|_2^2])$. □