REAP the Experts: Why Pruning Prevails for One-Shot MoE Compression

Mike Lasby*†, Ivan Lazarevich, Nish Sinnadurai, Sean Lie, Yani Ioannou†, Vithursan Thangarasa

ICLR

Motivation

- **Can we reduce the memory overhead of Sparsely-Activated Mixture-of-Experts (SMoE) models without sacrificing high-quality generative outputs or fine-tuning the compressed model?**
- Imbalanced expert usage motivated prior frequency-based pruning.
- Expert merging has been shown to outperform frequency-based pruning on multiple choice (MC) benchmarks.
- Does **strong accuracy on MC tasks correlate with generative tasks?**
- How do expert pruning and merging **alter the functional output space?**

Key insights

- Existing one-shot merging techniques use **tied router-gates which introduce irreducible error** due to the loss of the router's independent modulation of expert outputs.
- Merging distorts the functional manifold topology, resulting in **functional subspace collapse** in high-granularity MoEs.
- We introduce **REAP** (Router-weighted Expert Activation Pruning), a novel methodology for compressing MoEs that explicitly **minimizes the upper reconstruction error bound** by considering both router gate-values and expert activation norms.
- REAP consistently outperforms other expert compression methods across diverse architectures. Achieving **near-lossless compression on code generation tasks** after pruning 50% of experts from Qwen3-Coder-480B and Kimi-K2.

Analysis

- In the elementary case of reducing two experts to one we show that merging introduces an irreducible error **proportional to the variance of the router's input-dependent mixing ratio.**
- PCA decomposition of expert activations reveals that expert merging contracts the functional output manifold, inducing **functional subspace collapse**. Pruning as a **coordinate subspace operation** better preserves the topology.
- Measured via 1-Wasserstein distance, **merging introduces novel expert functions** that fundamentally distort the original manifold topology.

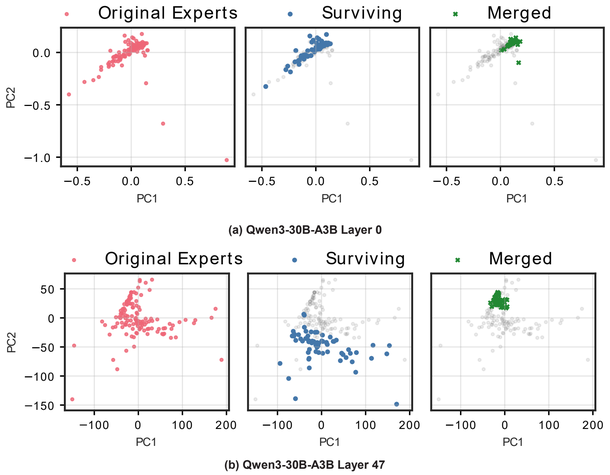

Figure 1: **Functional subspace (PCA) for Qwen3-30B-A3B.** Pruning (blue) preserves the expert output manifold geometry; merging (green) collapses it toward the centre. The contraction under merging is dramatically more pronounced in high-granularity MoEs with many experts per layer.

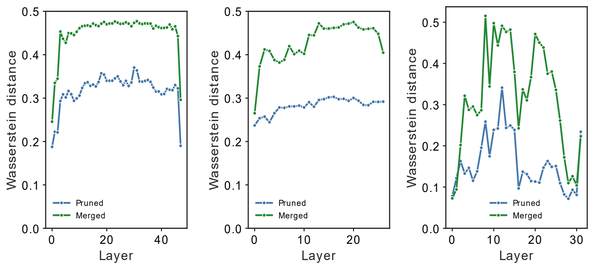

Figure 2: **1-Wasserstein distance between the compressed and original expert output manifolds measured in normalized angular distance.** Merging introduces novel functions which distort the manifold.

Method

- The expected error of pruning expert j is:

$$\mathcal{E}_{prune} = \mathbb{E}_{x|j\in\mathcal{T}(x)}\Big[\Big\| \underbrace{g_j(x)f_j(x) - g'_i(x)f_i(x)}_{\text{substitution error}} - \underbrace{\frac{g_j(x)-g'_i(x)}{1-g_j(x)+g'_i(x)} \sum_{k\neq i,j} g_k(x)f_k(x)}_{\text{renormalization error}} \Big\|_2^2\Big]$$

- We establish that the **upper bound on the substitution error magnitude** due to pruning expert j is $g_j(x)\big(||f_j(x)|| + ||f_i(x)||\big)$ where expert i is the *promoted* expert, which varies per token.
- Directly minimizing the magnitude of the known components is an **effective heuristic to minimize the expected error**.
- The **REAP saliency score** is defined as the average of an expert's weighted magnitude over tokens for which it is active (i.e., $\mathcal{X}_j = \{x \mid j \in \mathcal{T}(x)\}$):

$$S_j = \frac{1}{|\mathcal{X}_j|} \sum_{x\in\mathcal{X}_j} g_j(x) \cdot \big\|f_j(x)\big\|_2$$

- Crucially, calculating this average conditionally over $\mathcal{X}_j$ rather than globally **decouples the expert's functional impact from its frequency of activation**.

Results

- We evaluate REAP on MoEs ranging from **20B to 1T parameters** across code generation, mathematical reasoning, and creative writing.
- **REAP demonstrates consistently higher accuracy** on generative tasks.
- **Combining REAP and quantization** enables scaling to 1T parameters.
- Merged models have consistently **lower N-gram diversity and alignment** with the uncompressed model.

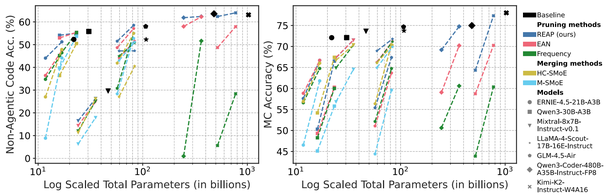

Figure 3: **Coding and MC accuracy across all models vs. parameters.**

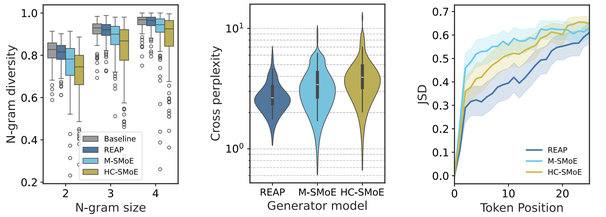

Figure 4: **N-Gram diversity, cross-perplexity, and JSD** of 50% compressed Qwen3-30B-A3B. JSD is measured between the compressed and baseline model logits vs. completion token position.

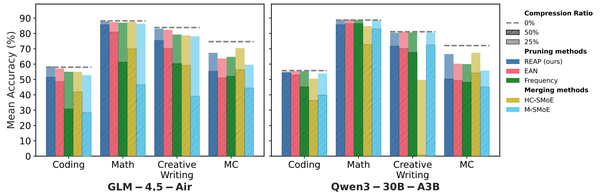

Figure 5:**GLM-4.5-Air and Qwen3-30B accuracy vs. task type**.

Table 1: **Large-scale pruned SMoEs on coding, tool-use, and MC benchmarks.**

Model	Compression	Method	Non-Agentic Coding Eval+	Non-Agentic Coding LiveCode	Non-Agentic Coding Code Avg	Agentic Coding SWE-Bench-Verified	Tool-Use (BFCLv3) Non-Live	Tool-Use (BFCLv3) Live	Tool-Use (BFCLv3) Multi-Turn	Tool-Use (BFCLv3) Overall	MC MC Avg
Qwen3-Coder-480B-A35B-Instruct-FP8	Baseline		0.841	0.431	0.636	0.540	0.866	0.825	0.380	0.690	0.750
	25%	Frequency	0.737	0.296	0.516	0.378	0.844	0.763	0.355	0.654	0.606
		EAN	0.827	0.419	0.623	0.534	0.831	0.813	0.384	0.676	0.702
		REAP	0.831	0.416	**0.624**	**0.540**	0.878	0.823	0.392	**0.698**	**0.748**
	50%	Frequency	0.007	0.012	0.010	0.000	0.200	0.392	0.000	0.197	0.506
		EAN	0.777	0.382	0.580	**0.536**	0.822	0.774	0.383	0.659	0.591
		REAP	0.822	0.415	**0.619**	0.522	0.849	0.801	0.371	**0.674**	**0.692**
Kimi-K2-Instruct-W4A16	Baseline		0.828	0.434	0.631	0.554	0.840	0.802	0.355	0.666	0.780
	25%	Frequency	0.486	0.082	0.284	0.000	0.644	0.603	0.045	0.431	0.604
		EAN	0.779	0.379	0.579	0.562	0.819	0.802	0.335	**0.652**	0.703
		REAP	0.840	0.440	**0.640**	**0.580**	0.842	0.801	0.263	0.635	**0.773**
	50%	Frequency	0.112	0.000	0.056	0.000	0.255	0.397	0.003	0.218	0.439
		EAN	0.722	0.253	0.487	**0.576**	0.778	0.767	0.173	**0.573**	0.587
		REAP	0.819	0.429	**0.624**	**0.576**	0.785	0.743	0.164	0.564	**0.643**

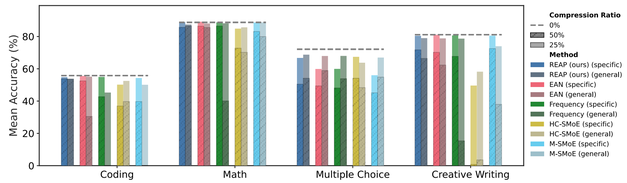

Figure 6: **Accuracy vs. task with domain specific vs. general calibration data.**

Project Page

* Work completed while on internship
† University of Calgary

UNIVERSITY OF CALGARY

cerebras

OpenReview