CARE-COM: GEOMETRY-GUIDED SEQUENTIAL FUNCTIONAL INTERVENTION FOR MIXTURE-OF-EXPERTS COMPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Sparse Mixture-of-Experts (MoE) models achieve high parameter capacity at low per-token cost, but their large expert pools create deployment memory pressure. Existing compression methods identify candidates for removal or merging using static signals computed from the initial model—parameter similarity, routing statistics, or output clustering. These signals do not directly measure the functional consequence of physically modifying an expert population, and they are not updated as compression proceeds.

We study expert compression through a functional intervention lens. We represent each expert by its responses to a fixed probe set—a *capability vector*—and show that distances between these vectors contain substantial predictive information about merge-induced functional damage: capability geometry achieves a mean held-out Spearman correlation of 0.7504 with observed merge damage on OLMoE-1B-7B-0924 (Muennighoff et al., 2024), versus 0.4797 for a local descriptor baseline. However, among 186 pairs satisfying a near-equal-distance criterion ($|\Delta d| < 10^{-4}$), observed KL damage spans 0.001522 to 0.007096, showing that geometry identifies a useful search neighborhood but does not uniquely determine intervention consequence.

This motivates CARE-COM: at each step, capability geometry generates a small candidate pool, each candidate is physically merged and evaluated for actual functional damage, the lowest-damage merge is committed, and the capability state is recomputed before the next step. In a controlled comparison on OLMoE, adaptive recomputation reduces cumulative KL divergence by 17.8%, 24.6%, and 15.9% at 60, 56, and 48 remaining experts relative to a frozen capability ranking, while also reducing perplexity at every target. On a common WikiText-2 benchmark, CARE-Adaptive achieves the lowest perplexity at 60 experts; Sub-MoE (Li et al., 2026) is slightly lower at 56 and 48 experts. These results support treating expert compression, in the evaluated OLMoE setting, as a sequential functional intervention problem in which the relevant functional state evolves after each physical modification.

1 INTRODUCTION

Sparse Mixture-of-Experts (MoE) architectures achieve high parameter capacity without proportionally increasing per-token computation (Shazeer et al., 2017; Fedus et al., 2022). A learned router selects a small subset of experts for each token, allowing models to contain many parameters while keeping active computation sparse. This design creates a deployment problem: the entire expert pool must remain stored even though only a fraction activates per token.

Expert compression—removing or merging experts—is a natural response. Recent methods approach this from several angles. Chen et al. (2025) perform retraining-free merging by hierarchically clustering experts based on their output behavior (HC-SMoE). Li et al. (2024) group experts by routing statistics, merge by weighted averaging, and compress further via low-rank decomposition (MC-SMoE). Li et al. (2026) cluster by output cosine similarity and merge via joint SVD in a shared subspace (Sub-MoE). Lasby et al. (2026) instead argue for pruning experts by a router-weighted activation saliency (REAP), providing a different perspective on one-shot compression.

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

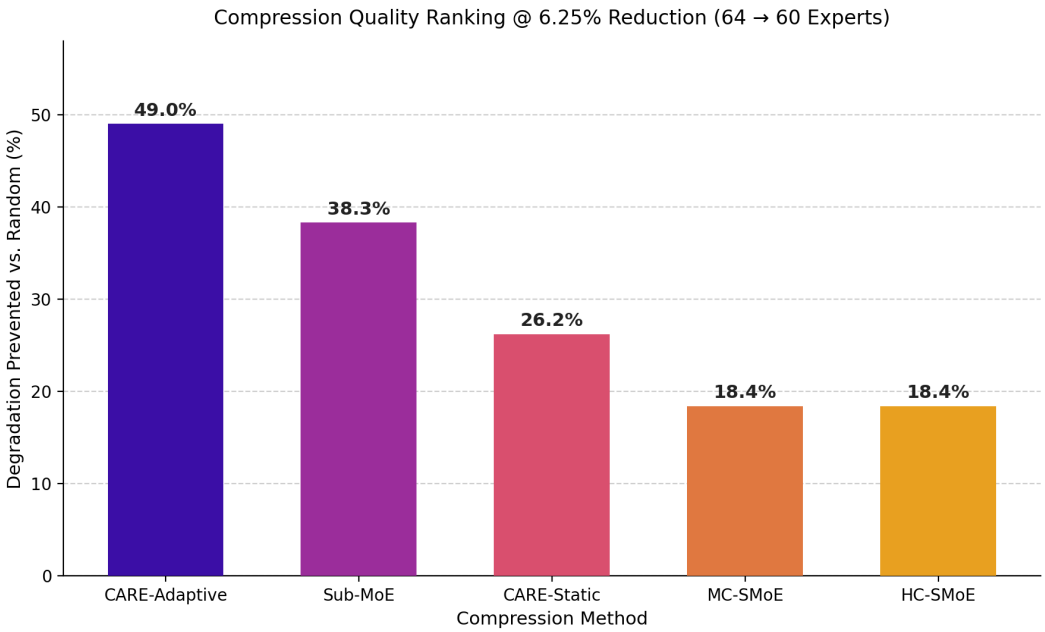

Figure 1: **Compression quality at 6.25% expert reduction** ($64 \rightarrow 60$)**.** The bar reports the percentage of WikiText-2 perplexity degradation—relative to the uncompressed model—prevented by each method, using the random baseline as the zero reference. CARE-Adaptive prevents 49.0% of degradation; Sub-MoE prevents 38.3%; CARE-Static 26.2%; HC-SMoE and MC-SMoE 18.4% each. At stronger compression levels (56, 48 experts), Sub-MoE achieves slightly lower perplexity than CARE-Adaptive (Table 1). The full three-level comparison is in Section 7.2.

These methods share a common dependency: they derive expert-relationship scores from the model's initial state and apply the resulting decisions without re-evaluating the functional state of the remaining experts after each modification. This raises a question that sits at the center of this paper: *after earlier merges have physically changed the model, does a relationship computed from the original state still determine whether a subsequent merge is safe?*

We address this through three empirical findings, each building on the previous:

Claim 1: Capability geometry is predictive. We represent each expert by its responses to a fixed probe set and measure pairwise Euclidean distances. These distances predict merge-induced functional damage substantially better than a local descriptor baseline (mean held-out Spearman $\rho = 0.7504$ vs. 0.4797).

Claim 2: Geometry is informative but not sufficient. Among 186 pairs with near-equal capability distances ($|\Delta d| < 10^{-4}$), observed KL damage spans a nearly $5\times$ range (0.001522 to 0.007096). Geometry identifies where to look; it does not determine what to do.

Claim 3: Functional state should be recomputed after each intervention. We compare two procedures that differ only in whether capability is recomputed after each physical merge. Adaptive recomputation reduces cumulative KL divergence by 17.8–24.6% across evaluated compression targets in a controlled single-run study on OLMoE.

These findings lead to CARE-COM:[1] at each step, capability geometry generates a candidate pool of size $K = 5$, each candidate is physically merged and evaluated for actual functional damage, the lowest-damage merge is committed, and capability is recomputed from the modified model before the next step.

Our contributions are:

[1]CARE-COM is the method name; the acronym is not formally expanded.

- **Functional geometry as a compression signal.** We show that capability geometry is substantially more predictive of merge-induced damage than a local descriptor baseline in a controlled evaluation on OLMoE.
- **Geometry is informative but insufficient.** Near-equal capability distances correspond to substantially different intervention consequences, motivating explicit functional evaluation before committing a merge.
- **Sequential functional intervention.** We formulate expert compression as a state-evolving sequential problem and introduce CARE-COM, which physically evaluates and then commits each merge before recomputing the functional state.
- **Controlled validation.** Adaptive recomputation consistently reduces cumulative functional divergence relative to a frozen capability ranking in a controlled single-run study at all three evaluated compression targets.
- **Competitive external performance.** CARE-Adaptive achieves the lowest WikiText-2 perplexity at 60 experts and is competitive with Sub-MoE at 56 and 48 experts under a common benchmark protocol.

2 Problem Formulation

Consider a sparse MoE model M_t at compression step t, containing N_t experts. We evaluate each expert's response to a fixed probe set $\mathcal{P}$ and construct a capability representation:

$$C_i^{(t)} = F(e_i; \mathcal{P}), \tag{1}$$

where F is the capability extraction procedure. Pairwise Euclidean distances define the functional geometry:

$$d_t(i,j) = \left\| C_i^{(t)} - C_j^{(t)} \right\|_2. \tag{2}$$

A compression action $a_t = (i,j)$ physically merges experts i and j, producing $M_t^{(i,j)} = T(M_t, (i,j))$, where T is the implemented merge operator. We define intervention damage as the KL divergence between the original and temporarily modified model outputs on a calibration distribution $\mathcal{D}_{\text{cal}}$:

$$D_t(i,j) = \mathbb{E}_{x \sim \mathcal{D}_{\text{cal}}} \left[D_{\text{KL}} \left(p_{M_t}(\cdot|x) \,\|\, p_{M_t^{(i,j)}}(\cdot|x) \right) \right]. \tag{3}$$

The true compression objective is $\min_{i,j} D_t(i,j)$, not $\min_{i,j} d_t(i,j)$. Because exhaustive evaluation of all pairs is expensive, a key question is whether geometry can serve as a useful candidate-generation mechanism for restricting the set of pairs actually evaluated.

3 Functional Geometry Predicts Merge Consequences

3.1 Evaluation Design

We test whether capability geometry predicts merge-induced functional damage, using $D_t(i,j)$ as the target. We compare three representations on the same OLMoE-1B-7B-0924 evaluation:

- **Local.** A predictor from 11 local pre-merge descriptors (parameter norms, activation statistics, routing entropy, etc.).
- **Geometry-only.** $\hat{D}_{ij} = \|z_i - z_j\|_2$, using only capability distance.
- **Combined.** A learned predictor using both local descriptors and capability distance.

Evaluation uses expert-disjoint cross-validation folds. The candidate selection budget for this predictive evaluation is $q = 4$, fixed before the experiment; this is the number of candidate pairs considered when measuring Precision@K for the geometry predictor. This parameter is distinct from the candidate budget $K = 5$ used inside CARE-COM (Section 5). Results are reported as mean held-out Spearman ρ and Precision@K.

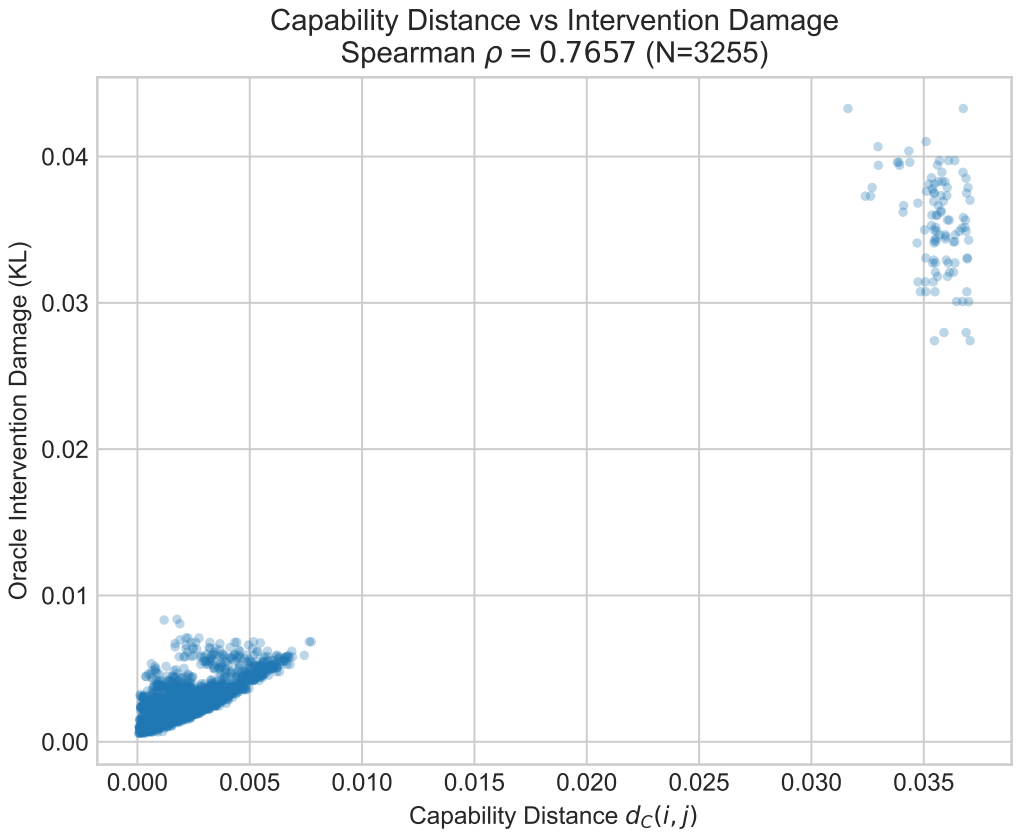

Figure 2: **Capability geometry predicts merge damage.** Mean held-out Spearman ρ: Local = 0.4797, Geometry = 0.7504, Combined = 0.8146. The pair-level ρ across all evaluated pairs is 0.7657. Note that geometry outperforms the combined predictor at $K = 10$ and $K = 25$; the combined predictor becomes stronger at $K = 50$. Neither uniformly dominates the other.

3.2 Results

Capability geometry substantially improves prediction of merge consequences over the local baseline. The mean held-out Spearman correlation increases from 0.4797 (local) to 0.7504 (geometry). The raw pair-level Spearman correlation is 0.7657. The combined representation reaches 0.8146.

The Precision@K results qualify this: geometry outperforms the combined predictor at $K = 10$ and $K = 25$; the combined predictor becomes stronger at $K = 50$. We do not claim that combining descriptors universally dominates geometry. The supportable conclusion is: **capability geometry contains substantial predictive information about merge-induced functional damage**, providing a principled mechanism for restricting the intervention search space.

4 Geometry Is Informative but Not Sufficient

4.1 Near-Equal-Distance Analysis

If capability distance completely determined merge consequence, pairs with near-equal distances would exhibit near-equal damage. We test this directly. Among 186 pairs satisfying $|\Delta d| < 10^{-4}$, observed KL damage ranges from 0.001522 to 0.007096—a nearly $5\times$ range. Two concrete examples (Figure 3): Pair A at distance ≈ 0.0156 produces KL ≈ 0.074, while Pair B at distance ≈ 0.0157 produces KL ≈ 0.046.

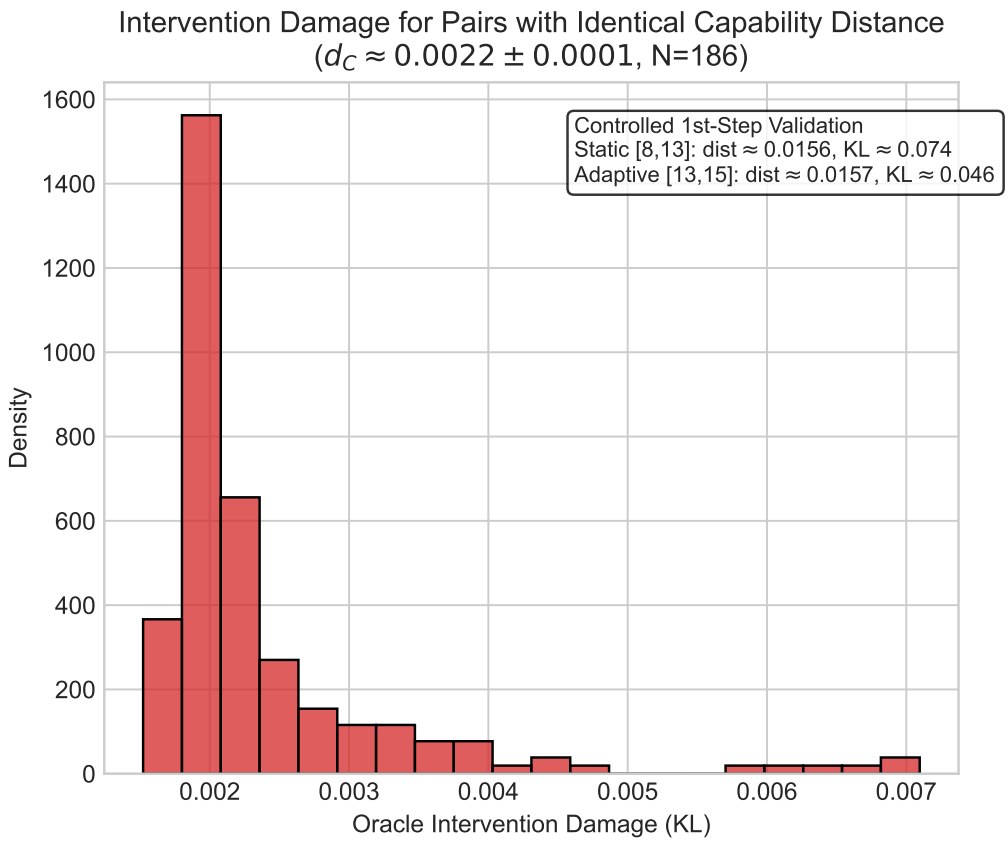

Figure 3: **Intervention damage for near-equal capability distances ($|\Delta d| < 10^{-4}$).** Among 186 such pairs, observed KL damage spans 0.001522 to 0.007096—a nearly $5\times$ range. Example: Pair A at distance ≈ 0.0156 has KL ≈ 0.074; Pair B at distance ≈ 0.0157 has KL ≈ 0.046. Near-equal capability distance does not uniquely determine intervention consequence.

This is not evidence that capability geometry is useless—Section 3.2 shows it is strongly predictive. Rather, it demonstrates that geometric proximity identifies a useful search neighborhood but does not uniquely specify the consequence of a physical intervention. The gap motivates two design choices in CARE-COM: (1) use geometry to generate candidates, not to make the final decision, and (2) evaluate actual physical damage before committing a merge.

5 CARE-COM

5.1 Conceptual Distinction from Similarity-Based Merging

Unlike similarity-based hierarchical merging (e.g., HC-SMoE (Chen et al., 2025)), which selects and commits merges using output similarity scores, CARE-COM explicitly evaluates physical intervention damage. For each candidate pair identified by geometry, a temporary merge is performed and evaluated on calibration data. The merge minimizing measured functional damage (KL divergence) is committed. Following this, the capability representation is recomputed from the modified model. In short: geometry identifies where to look; functional intervention evaluation determines what to do; state recomputation determines where to look next.

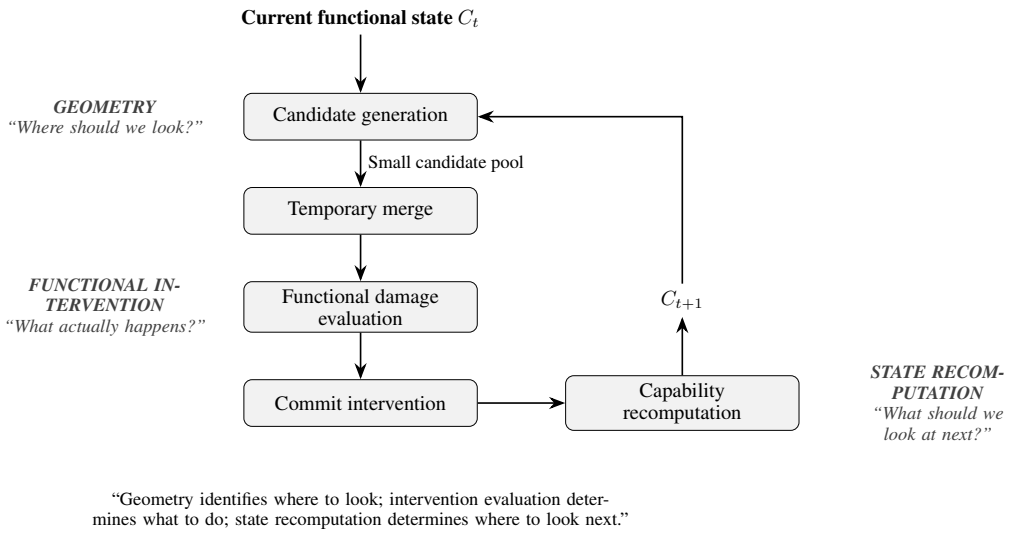

Figure 4: **The CARE-COM sequential intervention loop.** At each step, the current capability geometry C_t generates a candidate pool (Eq. 4). Each candidate is temporarily merged and evaluated for functional damage (Eq. 5). The lowest-damage merge is committed and capability is recomputed from the modified model (Eq. 6). This distinguishes CARE-COM from methods that select merges by similarity score alone.

5.2 Sequential Functional Intervention

CARE-COM formalizes this loop, evolving the state ($C_0 \to a_0 \to M_1 \to C_1 \to a_1 \dots$) rather than relying on a frozen initial ranking ($C_0 \to \mathrm{Rank}(C_0) \to a_1, a_2 \dots$). At step t with capability state C_t:

$$\mathcal{N}_K(C_t) = \mathrm{TopK}_{(i,j)}\left[-d_t(i,j)\right], \tag{4}$$

$$(i_t, j_t) = \arg\min_{(i,j)\in\mathcal{N}_K(C_t)} D_t(i,j), \tag{5}$$

$$M_{t+1} = T(M_t, (i_t, j_t)), \quad C_{t+1} = F(M_{t+1}; \mathcal{P}). \tag{6}$$

Equation 4 restricts the search space. Equation 5 evaluates physical damage. Equation 6 updates the model and functional state.

5.3 Candidate Budget

The budget $K = 5$ controls temporary physical evaluations per step, balancing search quality against evaluation cost. Sensitivity to K is not experimentally evaluated; this is a fixed design choice, not an optimal bound (see Section 8).

5.4 Physical Merge Operator

For OLMoE (Muennighoff et al., 2024), the implemented merge operator averages selected expert parameters:

$$\theta_{i,j}^{\mathrm{merged}} = \frac{\theta_i + \theta_j}{2}. \tag{7}$$

The merged expert replaces both originals, the expert list is reindexed, and router parameters are updated consistently to preserve expected routing mass. CARE-COM does not simulate a merge; it modifies actual model parameters. Full architecture and merge details are in Appendix B.

6 EXPERIMENTAL SETUP

6.1 MODEL AND EVALUATION

All controlled CARE-COM experiments use OLMoE-1B-7B-0924 (Muennighoff et al., 2024), an approximately 7B-parameter sparse MoE model with 64 experts per MoE layer and top-k routing. Intervention damage $D_t(i, j)$ is evaluated using KL divergence on a held-out calibration distribution. Perplexity is reported on WikiText-2. Expert capability representations are extracted once per expert using a fixed probe instrument applied to the current model state.

6.2 CONTROLLED STATIC VS. ADAPTIVE PROTOCOL

To isolate the effect of adaptive recomputation, we compare:

- **CARE-Static:** Capability geometry is computed from the initial 64-expert model and the ranking is frozen for all subsequent merges.
- **CARE-Adaptive:** Capability geometry is recomputed from the modified model after every physical merge.

Both conditions share: the same OLMoE-1B-7B-0924 checkpoint, physical merge operator, calibration data, target expert counts ($64 \rightarrow 60$, 56, 48), functional damage metric, candidate budget $K = 5$, and random seeds. The only difference is whether Equation 6 is applied. The comparison is a single experimental run under fixed seeds and batches; robustness to stochastic variation is not established by this study.

6.3 EXTERNAL BENCHMARK

We evaluate CARE-Adaptive and CARE-Static against HC-SMoE (Chen et al., 2025), MC-SMoE (Li et al., 2024), and Sub-MoE (Li et al., 2026) under a common WikiText-2 perplexity protocol using the same OLMoE checkpoint. REAP (Lasby et al., 2026) is a pruning method that removes experts rather than merging them; because our benchmark protocol evaluates physical expert merging, we do not report REAP perplexity numbers under this protocol. REAP is discussed as a complementary pruning approach in Related Work.

7 RESULTS

7.1 ADAPTIVE RECOMPUTATION REDUCES FUNCTIONAL DIVERGENCE

Under the controlled protocol (Section 6.2), adaptive recomputation reduces cumulative KL divergence at every evaluated compression target: 17.8% at 60 experts ($2.477 \rightarrow 2.036$), 24.6% at 56 experts ($6.820 \rightarrow 5.143$), and 15.9% at 48 experts ($17.694 \rightarrow 14.881$). WikiText-2 perplexity is also lower at every target: 11.548 vs. 12.112 at 60, 13.812 vs. 14.820 at 56, 20.136 vs. 21.032 at 48.

This result directly isolates the effect of adaptive capability recomputation: after physical intervention, recomputing the functional state provides information that a frozen initial ranking does not have. We do not interpret this as proof that recomputation is optimal for every MoE architecture or compression regime. The result is specific to the controlled OLMoE evaluation described here and to a single experimental run.

7.2 EXTERNAL COMPRESSION BENCHMARK

CARE-Adaptive achieves the lowest perplexity at 60 experts and consistently improves over CARE-Static at every level. Sub-MoE (Li et al., 2026) achieves slightly lower perplexity at 56 and 48 experts. We do not characterize CARE-COM as universally superior to Sub-MoE. The result demonstrates that adaptive functional intervention is competitive with recent subspace-based merging, while consistently and substantially improving over its static counterpart.

The comparison with HC-SMoE and MC-SMoE shows larger gaps at 48 experts (30.069 and 27.655 vs. 20.136), suggesting that CARE-Adaptive degrades more gracefully at stronger compression.

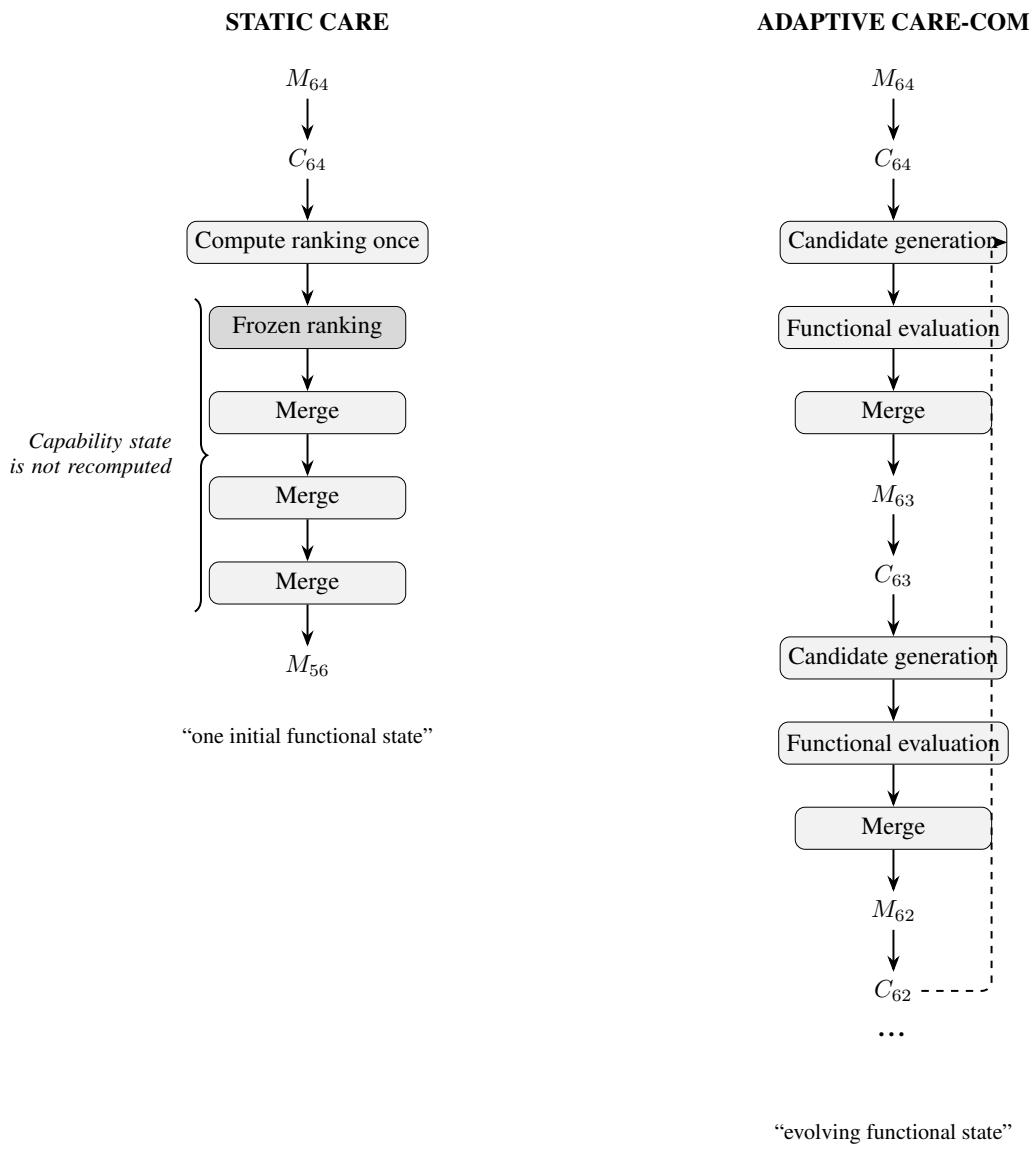

Figure 5: **Cumulative KL divergence (left) and WikiText-2 perplexity (right) for CARE-Static vs. CARE-Adaptive.** Adaptive recomputation reduces cumulative KL by 17.8% at 60 experts (2.477→2.036), 24.6% at 56 experts (6.820→5.143), and 15.9% at 48 experts (17.694→14.881). This is a controlled single-run study; multi-seed robustness is not established.

We note that this comparison is made under a single evaluation protocol on one model; different protocols or architectures may yield different orderings.

8 Limitations and Discussion

The three empirical findings—geometry predicts damage, geometry is insufficient alone, and adaptive recomputation improves over static selection—are established in a bounded context. **Single model run:** All experiments use OLMoE-1B-7B-0924 (64 experts/layer) in a single run under fixed seeds and batches. Generalization to other MoE architectures, router designs, scales, or multi-seed robustness is not established. **Probe calibration dependence:** Capability geometry and intervention damage rely on specific probe inputs and calibration data; different sets may alter results. **Operator budget:** We only study parameter averaging as the merge operator. The candidate budget

Table 1: **WikiText-2 perplexity (lower is better) under common benchmark protocol.** CARE-Adaptive consistently improves over CARE-Static at all compression levels. CARE-Adaptive achieves the lowest perplexity at 60 experts. Sub-MoE (Li et al., 2026) achieves lower perplexity at 56 and 48 experts. REAP (Lasby et al., 2026) operates by expert pruning rather than merging and is not evaluated under this merging protocol (see Section 6.3). Bold indicates best result for each column among merging methods.

Method	Type	60 Experts	56 Experts	48 Experts
CARE-Adaptive (Ours)	Merge	**11.548**	13.812	20.136
Sub-MoE (Li et al., 2026)	Merge	11.811	**13.735**	**19.477**
CARE-Static	Merge	12.112	14.820	21.032
HC-SMoE (Chen et al., 2025)	Merge	12.303	14.795	30.069
MC-SMoE (Li et al., 2024)	Merge	12.303	15.283	27.655
Random	Merge	12.758	17.298	38.104
REAP (Lasby et al., 2026)	Prune	not evaluated under this merging protocol		

$K = 5$ controls the cost of functional evaluation; sensitivity to K is not experimentally established. **Undissected advantage:** The controlled experiment isolates recomputation but does not fully separate the effect of functional damage selection from geometry-based selection.

9 Conclusion

We introduced CARE-COM, a sequential functional intervention framework for MoE compression, motivated by three empirical observations on OLMoE-1B-7B-0924: capability geometry predicts merge-induced functional damage substantially better than a local baseline; near-equal capability distances can correspond to substantially different intervention consequences; and recomputing the capability state after each physical merge reduces cumulative functional divergence relative to a frozen ranking in a controlled single-run study.

These findings support the following design: geometry identifies where to look, functional intervention evaluation determines what to do, and state recomputation determines where to look next. CARE-Adaptive achieves the lowest WikiText-2 perplexity at 60 experts among evaluated merging methods; Sub-MoE achieves slightly lower perplexity at 56 and 48 experts. We do not claim universal superiority. The results support treating expert compression, in the evaluated OLMoE setting, as a sequential functional intervention problem in which the relevant functional state evolves after each physical modification. Whether this framing generalizes across architectures, probe instruments, merge operators, and compression regimes is a question for future work.

10 Related Work

Expert merging. HC-SMoE (Chen et al., 2025) merges experts by hierarchical clustering of outputs, updating the similarity criterion sequentially. MC-SMoE (Li et al., 2024) groups by routing statistics, merging via activation-weighted averaging with low-rank compression. Sub-MoE (Li et al., 2026) clusters by output cosine similarity and merges via joint SVD.

Expert pruning. REAP (Lasby et al., 2026) removes experts using a router-weighted activation saliency score, arguing that pruning can outperform merging for generative tasks.

Positioning. CARE-COM differs by explicitly measuring functional intervention damage rather than relying strictly on similarity or saliency scores, and by recomputing capability representations after physical modification. While HC-SMoE also updates its clustering criterion sequentially, it selects based on output similarity. The static-versus-adaptive experiment isolates the effect of recomputation, though it does not fully disentangle functional damage evaluation from similarity-based selection (Section 8).

Reproducibility Statement

All experiments use OLMoE-1B-7B-0924 (Muennighoff et al., 2024) (checkpoint publicly available). The candidate budget is $K = 5$ throughout. The physical merge operator averages expert parameters and reindexes the model as described in Appendix B. Intervention damage is evaluated using KL divergence on a fixed calibration set. Perplexity is reported on WikiText-2 test split. Computational cost and evaluation counts are reported in Appendix C. The controlled static-versus-adaptive comparison uses identical random seeds, calibration batches, evaluation batches, and compression targets. The code and probe construction details are available in the anonymous repository: `https://anonymous.4open.science/r/CARE-EXP-D921/`.

AI Use Statement

In this work, we used generative AI tools (large language models) to assist with: literature organization and related-work drafting; LaTeX formatting and equation typesetting; code documentation; and writing and polishing the paper for clarity and consistency. We did not use generative AI tools to generate experimental results, produce figures from data, or design the core methodology. All AI-assisted text was reviewed, revised, and verified by the authors. We take full responsibility for the final content of this paper, including all claims, experimental results, and figures.

References

I-Chun Chen, Chelun Hsiao, Bo-Sheng Liu, Yen-Chi Huang, and Min-Yuh Hu. Retraining-free merging of sparse mixture-of-experts via hierarchical clustering. In *International Conference on Machine Learning*, 2025. URL `https://arxiv.org/abs/2410.08589`.

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. URL `http://jmlr.org/papers/v23/21-0998.html`.

Mike Lasby, Ivan Lazarevich, Nish Sinnadurai, Sean Lie, Yani Ioannou, and Vithursan Thangarasa. REAP the experts: Why pruning prevails for one-shot MoE compression. In *International Conference on Learning Representations*, 2026. URL `https://arxiv.org/abs/2510.13999`.

Lujun Li, Qiyuan Zhu, Jiacheng Wang, Wei Li, Hao Gu, Sirui Han, and Yike Guo. Sub-MoE: Efficient mixture-of-expert LLMs compression via subspace expert merging. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026. URL `https://arxiv.org/abs/2506.23266`.

Pingzhi Li, Zhenyu Yang, and Tianlong Yu. Merge, then compress: Demystify efficient SMoE with hints from its routing policy. In *International Conference on Learning Representations*, 2024. URL `https://arxiv.org/abs/2310.01334`.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, et al. OLMoE: Open mixture-of-experts language models. *arXiv preprint arXiv:2409.02060*, 2024. URL `https://arxiv.org/abs/2409.02060`.

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`.

A Additional Capability-Geometry Analysis

The pair-level Spearman correlation across all evaluated expert pairs is 0.7657; the mean held-out Spearman across expert-disjoint folds is 0.7504. The 186 near-equal-distance pairs used in Section 4 satisfy $|\Delta d| < 10^{-4}$; their KL damage spans 0.001522 to 0.007096. Figure 6 provides a concrete example.

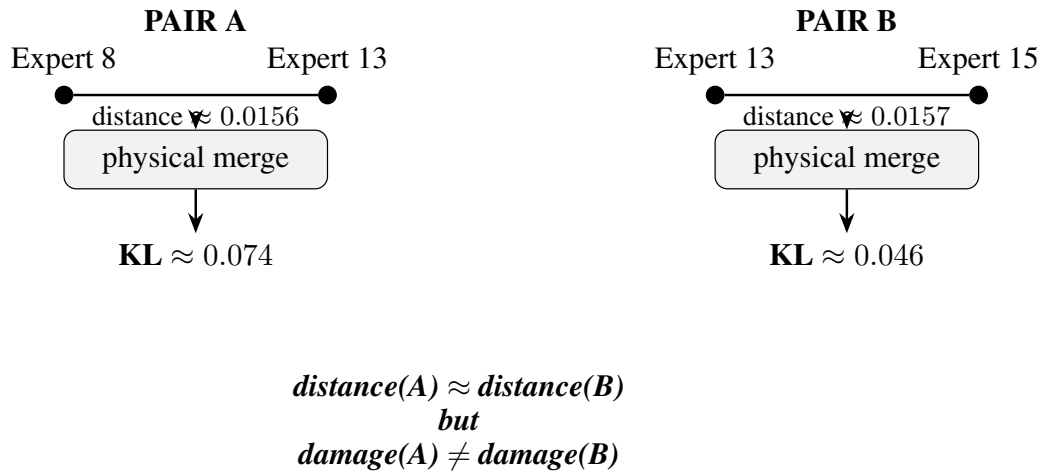

Figure 6: A controlled example illustrating the near-equal-distance result of Section 4. Pair A: capability distance ≈ 0.0156, observed KL ≈ 0.074. Pair B: capability distance ≈ 0.0157, observed KL ≈ 0.046. Despite a distance difference smaller than 10^{-4}, the intervention consequences differ substantially.

B OLMoE Architecture and Physical Merge Details

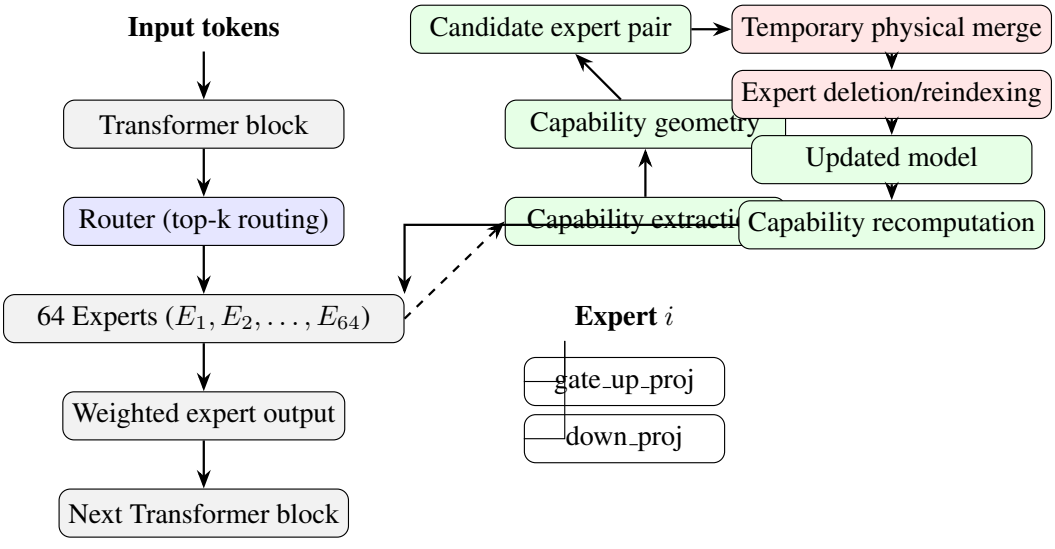

Figure 7: OLMoE architecture. Each MoE layer contains 64 experts. A top-k router selects a sparse subset per token. CARE-COM operates on the expert weight tensors of each MoE layer.

OLMoE-1B-7B-0924 contains multiple MoE layers, each with 64 experts. Each expert is a feed-forward network with gate, up, and down projection tensors. The router produces per-expert gate weights via a learned linear projection followed by top-k selection and softmax normalization.

Physical merge. For selected pair (i, j):

$$\theta_{i,j}^{\text{merged}} = \frac{\theta_i + \theta_j}{2}. \tag{8}$$

Expert i is replaced by the merged expert, expert j is deleted, and the expert list is reindexed. The router weight column for expert j is added to the column for expert i to preserve expected routing mass. The resulting model retains a valid native forward pass and is directly evaluable.

C Computational Cost and Efficiency

At the 48-expert target ($64 \to 48$, 16 merges), the audited implementation reports:

- CARE-Static: 115 exact functional evaluations; approximately 3387 s total runtime.
- CARE-Adaptive: 80 exact functional evaluations; approximately 666 s total runtime.

In this implementation and hardware environment, CARE-Adaptive uses approximately 30.4% fewer exact evaluations and approximately 80.3% less measured wall-clock time. These measurements are specific to the reported implementation and hardware. They should not be interpreted as hardware-independent universal runtime comparisons.

D Reproducibility Details

All experiments use the publicly available OLMoE-1B-7B-0924 checkpoint (Muennighoff et al., 2024). WikiText-2 perplexity is evaluated using the standard test split. The calibration distribution for KL divergence evaluation is a held-out subset of the pretraining data distribution. The probe instrument consists of a fixed set of synthetic inputs designed to probe diverse expert output behaviors; full probe construction details will be released with the code. The controlled static-versus-adaptive comparison uses identical random seeds, calibration batches, evaluation batches, and compression target sequence.