Why Do More Experts Fail? A Theoretical Analysis of Model Merging

Anonymous ACL submission

Abstract

Model merging dramatically reduces storage and computational resources by combining multiple expert models into a single multi-task model. However, existing methods struggle to maintain performance gains as the number of merged models increases. In this paper, we investigate the key obstacles that limit the scalability of model merging. We prove that the limited effective parameter space imposes a strict constraint on the number of models that can be successfully merged. Through Gaussian Width analysis, we show that marginal benefits diminish according to a strictly concave function as more models are merged. Using Approximate Kinematics Theory, we further prove the existence of a unique optimal threshold beyond which additional models yield negligible improvements. To address this limitation, we propose a straightforward Reparameterized Heavy-Tailed method to extend the merged model's coverage and enhance performance. Empirical results on 19 benchmarks, including both knowledge-intensive and general-purpose tasks, validate our theoretical analysis. We believe that these results spark further research beyond the current scope of model merging.

1 Introduction

Artificial General Intelligence is the ultimate goal pursued by researchers. Model merging offers a promising solution by integrating multiple task-specific expert models into a unified multi-task model (Yang et al., 2024; Yang et al.; Lu et al., 2024). Most existing works utilize LoRA fine-tuning to efficiently obtain experts with varying capabilities (Shah et al., 2024; Jang et al., 2023; Stoica et al.; Liu et al., 2025c; Marczak et al.; Ding et al., 2025). By combining the strengths of diverse expert models, a merged system can handle a broader range of tasks and adapt more effectively to complex problems, significantly reducing deployment resource consumption compared to using multiple separate models.

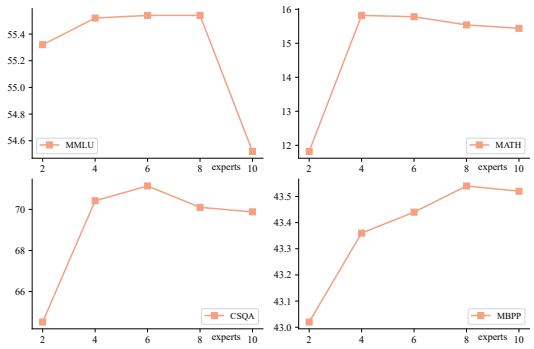

Figure 1: Exploration of the upper bound of effective expert merging in GENOME (Zhang et al., 2025), which verified scalability by enlarging the population size under a fixed expert pool.

The simplest approach directly adds or averages the weights of multiple models (Wortsman et al., 2022; Rame et al., 2022). However, this naive operation ignores parameter conflicts among expert models, often causing task interference and performance collapse. To address this, selection-based methods (Yadav et al., 2023; Yu et al., 2024) filter conflicting parameters before merging, which ignore deep-level overlaps and conflicts among parameter subspaces. Orthogonality-based strategies (Gao et al., 2024; Po et al., 2024; Zhang and Zhou, 2025) decompose parameters into orthogonal components and merge only non-interfering parts. This may discard some performance-critical non-orthogonal information. The latest evolutionary algorithms (Liu et al., 2025b; Feng et al.; Zhang et al., 2025) achieve new SOTA (state-of-the-art) by incorporating dynamic perception.

These merging methods have achieved landmark performance, but still have limitations in model merging. As shown in Figure 1, performance quickly plateaus as more experts are merged. A similar phenomenon can be observed with other merging methods. We find that the SOTA methods reach saturation after merging about six models

on most datasets (see Table 1), which greatly restricts the potential of model merging. Although some classical works (Yadav et al., 2024; Tao et al., 2024; Lee et al., 2025b) suggest the presence of a saturation effect in model merging, the reasons behind it are unexplored. Analyzing the underlying principles of this phenomenon is essential for overcoming the limitations of model merging capacity.

Therefore, we leverage high-dimensional geometry (Vershynin, 2015) and the Approximate Kinematics Theory (Amelunxen et al., 2014) to investigate the causes of the saturation phenomenon in model merging. First, we provide a theoretical analysis of how the parameter space of the merged model evolves as the number of experts increases. The findings show that Gaussian Width (Vershynin, 2015) of the parameters ceases to grow with additional experts, indicating that the effective parameter space of the merged model gradually saturates, leading to a performance bottleneck. Furthermore, we analyze the merged model through Approximate Kinematics Theory and reveal that performance degradation arises from parameter redundancy. We also observe that the effective parameter space of the merged model is highly sparse, resulting in limited coverage. To improve the scalability of merging methods, we propose a simple Reparameterized Heavy-Tailed (RHT) method that enhances coverage of the parameter space by amplifying the heavy-tailed distribution, thereby improving overall performance. Experiments on both knowledge-intensive and general-purpose tasks verify the correctness of our method and theories. The main contributions of this work are:

- We prove that as the number of experts increases, the effective parameter space of the model rapidly saturates, leading to diminishing returns in performance;
- We prove the existence of an upper bound for model merging and provide its analytical expression, highlighting performance limitations caused by parameter redundancy and offering theoretical guidance for optimizing expert model merging;
- We introduce a simple Reparameterized Heavy-Tailed method to enhance the coverage of the merged model by extending its heavy-tailed distribution;
- Extensive experiments on both general-purpose and knowledge-intensive tasks validate the correctness and effectiveness of our theories and method.

2 Related Work

Expert fine-tuning Acquiring experts using efficient parameter fine-tuning (PEFT) is a popular paradigm. PEFT (Li and Liang, 2021; Liu et al., 2024a,b) has emerged as the dominant strategy for this purpose, valued for its ability to adapt models by modifying only a small subset of parameters. This substantially lowers the computational cost of creating multiple experts. In the context of model merging, PEFT techniques such as LoRA (Choi et al., 2024; Zhang et al., 2023; Luo et al., 2024) are frequently employed to efficiently integrate these experts into a single, parameter-efficient representation. However, this approach introduces a fundamental tension: while PEFT is primarily designed to optimize performance on individual downstream tasks, this objective is not always aligned with the goal of retaining and fusing the diverse knowledge of all experts. As a result, there exists an inherent trade-off between the efficiency of fine-tuning and the comprehensive integration of expertise.

Model Merging Model merging aims to optimize performance by leveraging complementary capabilities of different models. Static methods (Jang et al., 2024; Si et al., 2025b; Zeng et al., 2025) merge parameters without additional data, while dynamic methods (Yang et al.; Prabhakar et al., 2025; Wu et al.; Liu et al., 2025a) optimize merging weights for multi-skill composition. Recent research has modeled the merging of LLMs as an optimization problem, with approaches like (Akiba et al., 2025; Lee et al., 2025a; Feng et al.). However, the former tends to simplify evolutionary mechanisms or focus solely on merging coefficients, while the latter adjusts model weights using swarm intelligence, which may lead to local optima. GENOME, on the other hand, enhances the effectiveness of the evolutionary algorithm by incorporating genetic-level and population-level operations. Despite these efforts to merge multiple experts, the actual number of experts effectively merged for optimal performance is often much lower than anticipated. To investigate this phenomenon, we first examine the parameter space of experts and then expand it to enable the effective merging of more experts.

3 Theory

To formally describe the merging process, let $\theta_0 \in \mathbb{R}^d$ denote the weights of the pre-trained model,

$\{\theta_1, \theta_2, \cdots, \theta_n\}$ represent the expert parameters, where n represents the number of experts.

We prove the existence of an upper bound for model merging and provide a theoretical adaptive termination condition (Theorem 1). We reveal that the marginal contribution of additional experts to the overall subspace gradually diminishes, causing performance saturation (Theorem 2). Through Approximate Kinematics Theory, we reveal that parameter redundancy leads to performance degradation (Theorem 3). Based on these insights, we propose RHT to improve the parameter space coverage of the merged model (Theorem 5).

Theorem 1 (Upper Bound of Model Merging)**.** *As the number of merging experts increases, the variance of the combined model approaches a constant, and the performance of the model approaches saturation (Proof in Appendix A.1).*

Based on prior literature (Si et al., 2025a) and our preliminary experimental results (Wang et al., 2024; Huang et al., 2024a; Zhang et al., 2025)(Appendix Figure 8), the parameters of the expert model follow a Gaussian distribution. Specifically, the weights follow $\theta_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, with covariance between models i and j given by $\mathrm{Cov}(\theta_i, \theta_j) = \rho_{ij}\,\sigma_i\,\sigma_j$, where $|\rho_{ij}| \leq 1$. The variance of the merged model is given by:

$$\sigma^2_{\text{merge}} = \sigma^2\big(\rho + (1-\rho)\textstyle\sum_{i=1}^{n} \alpha_i^2\big). \quad (1)$$

In the uniform weight case $\alpha_i = 1/n$, as the number of experts $n \to \infty$, the merged variance approaches $\lim_{n\to\infty} \sigma^2_{\text{merge}} = \sigma^2\rho$, indicating a theoretical lower bound $\sigma^2\rho$. To ensure each additional expert reduces variance by at least $\Delta > 0$, we consider the marginal variance drop: $\sigma^2_{\text{merge}}(n-1) - \sigma^2_{\text{merge}}(n) \geq \Delta$, leading to an upper bound:

$$n(n-1) \leq \sigma^2(1-\rho)/\Delta. \quad (2)$$

This shows merging too many experts offers diminishing returns. Larger Δ requires fewer experts for strong performance. Regularizing ρ to enforce orthogonality can further improve merging. An adaptive stopping condition can be defined as $\Delta = \mathbb{E}[\sigma^2_{merge}(n-1) - \sigma^2_{merge}(n)]$, terminating merging when variance reduction falls below this threshold.

3.1 Marginal Effects in Parameter Subspace

Theorem 2 (Diminishing Marginal Effects in Model Merging)**.** *Let θ^* denote the parameters obtained by merging all n experts, and $L(\theta^*)$ denote the corresponding loss. The goal of model merging is to find the minimal merging size M such that the merged model achieves optimal merging performance. This objective is formulated as the following constrained minimization problem:*

$$\min_{M\in\mathbb{Z}^+} M \quad s.t. \quad \inf_{\theta\in\Theta_M} L(\theta) \leq L(\theta^*) + \epsilon, \quad (3)$$

where $\Theta_M \subset \mathbb{R}^D$ is the parameter space achievable by merging M experts, and the feasible set:

$$S(\epsilon) = \{\theta \in \mathbb{R}^D : L(\theta) \leq L(\theta^*) + \epsilon\}. \quad (4)$$

Here, $S(\epsilon)$ is the set of admissible parameter configurations for an M-expert model, ϵ is the performance tolerance threshold. Locally near θ^*, $S(\epsilon)$ can be approximated as an ellipsoid defined by the Hessian H of $L(\theta)$. The Gaussian Width (Vershynin, 2015) of the set $S(\epsilon)$ can be proven as (Appendix A.2):

$$w(S(\epsilon)) \approx \sqrt{2\epsilon \cdot Tr(H^{-1})}. \quad (5)$$

For M experts, the Gaussian Width becomes: $w(S_M) \approx \sqrt{2\epsilon \cdot \sum_{i=1}^{M} 1/\lambda_i}$, where λ_i is the i-th eigenvalue of H. The marginal contribution of adding the M-th expert is: $\Delta w_M = w(S_M) - w(S_{M-1})$. Since the square root function is concave, the marginal gain decreases as M increases: $\Delta w_M > \Delta w_{M+1}$.

This indicates that as the number of experts M increases, adding new experts expands the dimensionality of the parameter space, but the marginal contribution of each additional dimension to the Gaussian Width diminishes, ultimately leading to saturation of model merging performance.

3.2 Parameter Redundancy Effects

Theorem 3 (Parameter Redundancy and Expert Model Merging Performance)**.** *As the number of merged experts M increases, the number of non-zero parameters k grows. When parameter redundancy exceeds a threshold, maintaining the loss within the sublevel set becomes impossible, causing performance decline. Specifically, when k satisfies:*

$$k \leq d - \sum_{i=1}^{d-k} \frac{r_i^2}{\|\theta^* - \theta^k\|_2^2 + r_i^2}, \quad (6)$$

where d is the total number of parameters and $r_i = \sqrt{2\epsilon/\lambda_i}$ is the ellipsoid radius. Beyond this threshold, performance degradation becomes inevitable (Proof in Appendix A.3).

3.3 Reparameterized Heavy-Tailed Method

The expert weights follow a Gaussian distribution (Si et al., 2025a), and the parameters of the merged multi-expert model, $\mathbf{w} \in \mathbb{R}^d$, follow a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. For simplicity, we assume $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$, and define a two-step transformation: 1. Gaussian difference : $\mathbf{w}' = \mathbf{w} - \mathbf{g}$, $\mathbf{g} \sim \mathcal{N}(\boldsymbol{\mu}, \sigma_g^2\mathbf{I})$, where $\mathbf{w}$ and $\mathbf{g}$ are independent. As a result, $\mathbf{w}' \sim \mathcal{N}(\mathbf{0}, (\sigma^2 + \sigma_g^2)\mathbf{I})$ (Proof in Appendix A.4); 2. Component-wise nonlinear amplification: $\mathbf{w}'' = T(\mathbf{w}')$, $T : \mathbb{R}^d \to \mathbb{R}^d$.

Theorem 4 (Heavy-Tailed Emergence From Model Merging)**.** *Considering a single expert after Gaussian perturbation and nonlinear transformation:*

$$T(x) = \mathrm{sign}(x) \cdot |x|^{\gamma} \cdot \left(1 + \alpha \cdot e^{-\beta|x|}\right), \quad (7)$$

with $0 < \gamma < 1$, $\alpha > 0$, $\beta > 0$. *For a fixed scale* $\widehat{\sigma}$, *the tail probability of the transformed variable* $Y = T(x)$ *has the asymptotic form:*

$$P(|Y| > y) \sim C\, y^{-1/\gamma} \exp\left(-y^{2/\gamma}/2\widehat{\sigma}^2\right). \quad (8)$$

When merging multiple experts with heterogeneous scale $\widehat{\sigma}$, *the tail probability becomes a scale mixture. If the distribution of scales gives enough weight to very large variances, formally, if the distribution of* $\theta = 1/(2\widehat{\sigma}^2) \to 0^+$ *with exponent* $\delta > 0$, *then by Tauberian theory (Proof in Appendix A.5):*

$$P(|Y| > y) \propto |y|^{-\kappa}, \kappa = (1 + 2\delta)/\gamma. \quad (9)$$

Thus, the merged model exhibits power-law heavy tails.

Theorem 5 (Heavy-Tailed Distributions Enhance Model Coverage)**.** *Let* $\mathcal{P}_{\mathrm{Heavy}} = (0, \infty)$ *denote the parameter set of heavy-tailed Student's t-distributions, and* $\mathcal{P}_{\mathrm{Exp}} = \{\infty\}$ *denote the limiting parameter set corresponding to exponential-tailed Gaussian distributions (Proof in Appendix A.6). Then,* $\mathcal{P}_{\mathrm{Exp}} \subset \overline{\mathcal{P}_{\mathrm{Heavy}}}$ *with strict inclusion.*

Hence, the admissible parameter space of heavy-tailed models strictly dominates that of exponential-tailed models, implying that heavy-tailed families possess strictly greater coverage and expressive capacity in the parameter domain.

4 Experiments

4.1 Experiments Setup

Our experiments are strictly performed on high-performance computing hardware, NVIDIA-A800-SXM4-80GB, to ensure the efficiency and scalability of the model. Complete training hyperparameters and configurations are detailed in Appendix B.

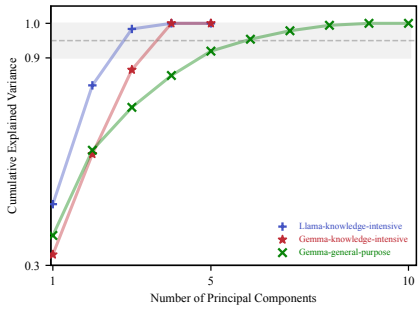

Figure 2: Cumulative variance across different experts.

Datasets. Our experiments rely on two categories of datasets: general-purpose and synthetically constructed. The general-purpose datasets (D_{gend}) include two widely adopted benchmarks. The first is the **GLUE** (Wang et al., 2018) benchmark, covering **CoLA** (Warstadt et al., 2019), **MNLI** (Williams et al., 2018), **MRPC** (Dolan and Brockett, 2005), **QNLI** (Rajpurkar et al., 2016), **QQP** (Iyer et al., 2017), **RTE** (Giampiccolo et al., 2007), and **SST-2** (Socher et al., 2013). The second consists of six standard LLM evaluation benchmarks spanning key capabilities: commonsense reasoning (**MMLU** (Hendrycks et al.)), mathematics (**MATH** (Hendrycks et al., 2021)), code generation (**MBPP** (Austin et al., 2021)), multilingual processing (**MGSM** (Shi et al.)), affective computing (**EmoryNLP** (Zahiri and Choi, 2018)), and question answering (**CSQA** (Talmor et al., 2019)). In addition, we construct synthetic datasets to evaluate knowledge-intensive (D_{knowd}) scenarios, designing two task types: title generation (**Phy-title**, **Chem-title**, **Bio-title**) and translation (**Phy-trans**, **Chem-trans**, **Bio-trans**). To build these datasets, we randomly sample 500 seed instances from the original expert training corpus, retrieve semantically aligned references from domain-specific knowledge bases using k-nearest neighbor search, and generate task-specific QA pairs with GPT-4o-mini. Low-quality samples are iteratively refined through expert consensus until meeting quality standards. Each dataset is then split into 150 validation and 350 test samples. For the biology title generation task, we further ensure data reliability by directly selecting 200 validation and 1,077 test samples from the original knowledge base, accompanied by expert consensus evaluation.

Baselines. We compare various representative model **merging methods**, including Weight Aver-

Model	MMLU	MATH	MGSM	CSQA	MBPP	EmoryNLP
GENOME-2LoRA	55.32(0.5)	11.82(0.5)	34.22(0.6)	64.52(1.7)	43.02(0.4)	34.78(0.2)
GENOME-4LoRA	55.52(0.6)	**15.82(0.9)**	**36.48(0.9)**	70.42(0.7)	43.36(0.4)	34.40(0.7)
GENOME-6LoRA	**55.54(0.3)**	15.78(0.7)	36.06(1.0)	**71.14(1.0)**	43.44(0.2)	**35.12(0.6)**
GENOME-8LoRA	55.54(0.8)	15.54(0.3)	35.46(0.7)	70.10(0.6)	**43.54(0.2)**	35.04(0.5)
GENOME-10LoRA	54.52(0.9)	15.44(0.8)	36.14(1.3)	69.88(0.7)	43.52(0.5)	35.04(0.6)
Swarms-2LoRA	54.96(0.3)	10.10(0.4)	34.00(0.3)	64.94(0.5)	43.50(0.3)	34.54(0.4)
Swarms-4LoRA	**55.70(1.0)**	**16.22(1.4)**	**36.66(1.1)**	**70.44(0.9)**	43.48(0.4)	34.56(0.9)
Swarms-6LoRA	55.46(0.3)	15.60(0.3)	36.16(1.4)	69.76(0.8)	43.30(0.6)	34.78(0.5)
Swarms-8LoRA	55.12(1.1)	15.30(0.7)	35.30(2.1)	68.76(1.4)	**43.86(0.3)**	35.30(0.5)
Swarms-10LoRA	55.46(0.5)	15.04(0.9)	36.12(1.1)	69.58(1.5)	43.52(0.5)	**35.86(1.3)**

Table 1: Performance of GENOME and Model Swarms with 2–10 LoRA fusions on the D_{gend} corpus under the DARE_TIES merging strategy. Results averaged over 5 random seeds with standard deviations.

	Phy-title		Phy-trans	Chem-title		Chem-trans	Bio-title		Bio-trans
Model	F1	ROUGE	BLEU	F1	ROUGE	BLEU	F1	ROUGE	BLEU
Base	45.95(1.1)	39.04(0.7)	50.61(0.4)	41.54(1.1)	33.83(1.0)	19.09(0.2)	28.27(0.3)	23.65(0.1)	24.49(0.8)
Single	50.64(0.7)	43.15(0.4)	**53.42(0.3)**	49.20(0.7)	40.48(0.7)	22.56(0.8)	35.69(0.3)	30.15(0.1)	28.65(1.5)
3-LoRA	**56.70(0.5)**	**48.47(0.4)**	51.88(0.2)	**59.10(0.1)**	**52.08(0.1)**	38.22(0.3)	**39.79(0.1)**	**34.87(0.1)**	**47.68(0.1)**
4-LoRA	56.19(0.9)	48.00(0.8)	51.79(0.6)	58.96(0.1)	51.98(0.2)	37.66(0.6)	39.57(0.6)	34.46(0.8)	47.60(0.1)
5-LoRA	55.22(0.9)	47.15(0.4)	51.97(0.7)	58.39(0.6)	51.28(1.0)	**38.25(0.2)**	39.47(0.8)	34.53(0.7)	47.63(0.2)
Base	48.67(0.5)	41.55(0.6)	49.78(0.3)	44.40(0.3)	37.82(0.4)	35.27(0.5)	29.40(0.1)	24.19(0.1)	46.49(0.5)
Single	54.92(0.5)	47.93(0.6)	49.88(0.2)	58.53(0.9)	53.08(0.7)	35.60(0.2)	39.42(0.1)	35.28(0.3)	47.19(0.3)
3-LoRA	56.26(0.1)	48.86(0.1)	**52.26(0.1)**	61.76(0.1)	56.43(0.5)	38.01(0.1)	39.97(0.2)	**35.82(0.3)**	46.51(0.3)
4-LoRA	56.28(0.3)	48.80(0.3)	52.25(0.1)	**61.92(0.1)**	**56.47(0.1)**	37.94(0.2)	39.93(0.2)	35.49(0.3)	**48.72(0.3)**
5-LoRA	**57.08(0.3)**	**49.34(0.3)**	52.08(0.3)	61.54(0.2)	55.81(0.3)	**38.19(0.1)**	**40.10(0.1)**	35.78(0.2)	48.70(0.1)

Table 2: Zero-shot performance comparison of different settings on Gemma-2-2B-it (top) and LLaMA3.1-8B-Instruct (bottom) models across various domain-specific tasks. "Single" to a LoRA model trained on D_{knowd}, "3-LoRA", "4-LoRA", and "5-LoRA" correspond to the first 3, 4, and 5 items in the sequence of "physics, chemistry, biology, finance, and medicine".

aging, Fisher Merging (Matena and Raffel, 2022), RegMean (Jin et al., 2023), Task Arithmetic (Ilharco et al.), TIES-Merging (Yadav et al., 2023), DARE, DARE_TIES (Yu et al., 2024), EMR-Merging (Huang et al., 2024b), GENOME (Zhang et al., 2025), and Model Swarms (Feng et al.). Detailed descriptions of these methods are provided in Appendix B.2. These methods are evaluated on multiple **base models**, including LLaMA3.1-8B-Instruct (Touvron et al., 2023), Gemma-2-2B-it (Team et al., 2024), and GPT-2 (Radford et al., 2019). In D_{gend}, we employ both full-parameter and LoRA fine-tuning. We obtain publicly available full-parameter checkpoints from Hugging Face following the setup of EMR-Merging (Huang et al., 2024b), and train LoRA experts on ten subdomains of the Tulu-v2-SFT-mixture dataset (Ivison et al., 2023) according to GENOME (Zhang et al., 2025). In D_{knowd}, we train LoRA models covering fields such as physics, chemistry, biology, medicine, and finance.

4.2 Upper Bound for Model Merging

Table 1 illustrates how the number of merged experts influences the performance of GENOME and Model Swarms under the DARE_TIES merging strategy. This experiment investigates the scaling dimension of expert quantity, an aspect that has received limited systematic attention in prior work. Specifically, GENOME primarily focuses on scaling the evolutionary population size from 10 to 40 models with a fixed expert pool, and Model Swarms explores expert diversity within a fixed-size pool, neither study systematically examines the impact of expert quantity on final performance. Our experimental results reveal a critical finding: contrary to the default configuration of 10 experts adopted by both methods, simply increasing the number of experts does not guarantee monotonic performance improvement. As shown in Table 1, optimal performance is typically achieved when merging about 6 experts for GENOME (and about 4 for Model Swarms), after which performance plateaus or even declines slightly.

This saturation phenomenon can be theoretically explained by Theorem 2, which establishes that the marginal contribution of each additional expert diminishes as the number of merged experts increases. Intuitively, the first few experts capture the dominant directions in the Hessian curvature space, those corresponding to large eigenvalues where the loss function is most sensitive to parameter changes. As more experts are added, they increasingly align with low-curvature directions (smaller

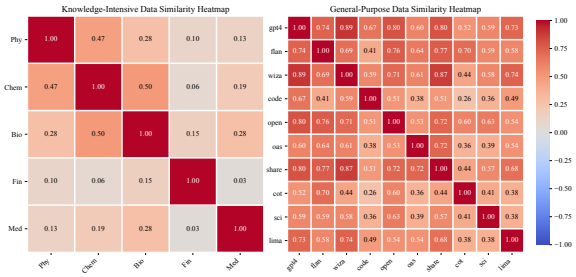

Figure 3: Heatmap of cosine similarities between sentence embeddings of D_{knowd} and D_{gend}.

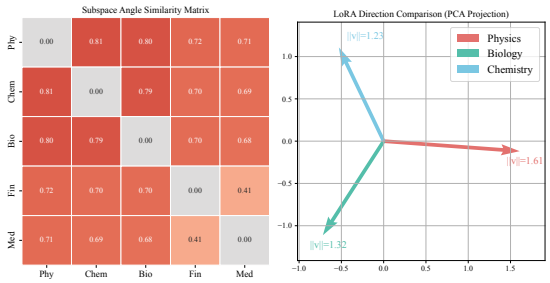

Figure 4: Analysis of domain model subspace orthogonality and PCA projections.

	Phy-title		Phy-trans	Chem-title		Chem-trans	Bio-title		Bio-trans
Model	F1	ROUGE	BLEU	F1	ROUGE	BLEU	F1	ROUGE	BLEU
Single-LoRA	54.92(0.5)	47.93(0.6)	49.88(0.2)	58.53(0.9)	53.08(0.7)	35.60(0.2)	39.42(0.1)	35.28(0.3)	47.19(0.3)
Phy+Chem	56.23(0.2)	48.99(0.3)	52.14(0.1)	61.14(0.2)	55.83(0.5)	37.98(0.2)	-	-	-
Phy+Bio	56.15(0.1)	48.48(0.2)	52.16(0.1)	-	-	-	40.05(0.1)	35.85(0.1)	46.15(0.1)
Chem+Bio	-	-	-	61.07(0.1)	55.48(0.1)	38.13(0.2)	39.99(0.2)	35.53(0.5)	46.61(0.4)
Phy1+Phy2	56.07(0.2)	48.74(0.5)	51.84(0.1)	-	-	-	-	-	-

Table 3: Performance of pairwise expert merging experiments across domains.

eigenvalues), where parameter modifications have minimal impact on the loss. Formally, since the Gaussian Width grows as $w(S_M) \propto \sqrt{\sum_{i=1}^{M} 1/\lambda_i}$, the concavity of the square root function ensures that $\Delta w_M = w(S_M) - w(S_{M-1})$ decreases as M grows. Consequently, beyond a certain threshold, adding more experts yields diminishing or even negative returns due to the introduction of redundant or conflicting parameters in directions that contribute negligibly to performance improvement. Additional evidence of this saturation behavior is provided in Appendix Table 7.

To further verify this phenomenon, we perform principal component analysis (PCA) (Wold et al., 1987) on the weights of various experts used in the experiments (see Figure 2). The results indicate that the number of principal components explaining approximately 95% of the total variance closely aligns with the number of experts at which model performance peaks. This indicates that the first few experts capture the dominant high-variance directions in the parameter space, while additional experts beyond this threshold contribute primarily along the remaining low-variance directions. This observation is consistent with Theorem 3: the performance degradation arises from parameter redundancy in these low-variance directions. Specifically, after the first few experts have captured the high-variance subspace, subsequent experts introduce parameters that lie in directions with small eigenvalues. Although DARE_TIES employs sparsification, the cumulative number of non-zero parameters k inevitably grows. Once k exceeds the theoretical bound, these redundant parameters in low-variance directions introduce conflicts rather than improvements, leading to performance degradation. In summary, our findings align with the theoretical analysis: expert model merging enhances performance within a certain range, but as the number of experts grows, the marginal benefit gradually decreases, and performance is ultimately limited by parameter redundancy.

4.3 Impact of Domain Similarity on Model Merging

Figure 3 shows the cosine similarity between the embeddings of D_{knowd} and D_{gend}, reflecting their correlation differences. Experiments on these corpus are presented in Tables 1 and 2, both exhibiting performance saturation.

According to Theorem 1, blindly increasing the number of experts does not always improve performance. Enhancing the quality and diversity of individual experts is often more effective than simply increasing their number. Table 3 shows the results of merging experts from different and same domains. Experiments in the physics domain indicate that merging models from different domains outperforms merging those from the same domain. Theorem 1 further indicates that the upper bound on the number of effectively mergeable models is constrained by inter-expert correlation, with higher correlation imposing a stricter bound. Therefore, prioritizing expert combinations with lower correlation tends to yield better performance gains.

SVD (Klema and Laub, 1980) is widely used to extract principal components from data matrices. Based on SVD, we analyze representation differ-

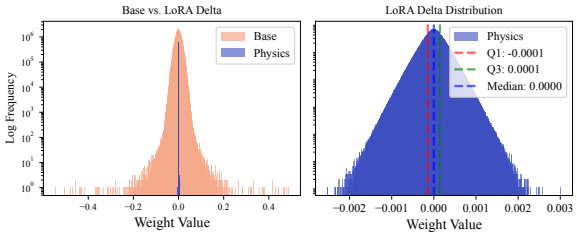

Figure 5: Weight distributions of the base model vs. LoRA fine-tuning on physics.

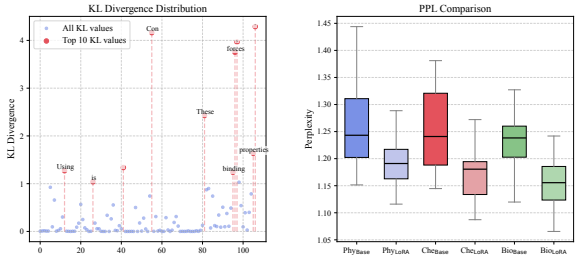

Figure 6: Comparison of KL divergence and perplexity. (Left) KL divergence between the base and LoRA fine-tuned models, with the Top 10 KL values marked in red corresponding to domain-independent tokens, indicating minimal adjustments to the overall output. (Right) Perplexity reduction of LoRA models across physics, chemistry, and biology, suggesting improved response accuracy and stability.

ences among five models in D_{knowd}, focusing on subspace similarity across domains. We measure the principal angles between these subspaces. As shown in the left part of Figure 4, the principal angles between the physics, chemistry, and biology subspaces are close to 90 degrees, indicating near orthogonality and minimal interference. In contrast, the smaller angles between finance and medical subspaces suggest significant overlap, which may cause conflicts in domain-specific information and affect merging. The 4LoRA and 5LoRA experiments in Table 2 further confirm this phenomenon: adding medical LoRA to 4LoRA degrades performance, indicating that subspace coupling between finance and medical models leads to negative interference, limiting merging effectiveness.

To better quantify differences between domain models, we perform PCA to examine the representations of physics, chemistry, and biology in reduced-dimensional space (see the right part of Figure 4). The projections show that vectors from these domains point in distinct directions along the first two principal components, revealing significant differences in variation patterns. Due to the approximate orthogonality of their subspaces, the vectors exhibit minimal overlap, effectively reducing interference during fusion and ensuring stability and independence in parameter integration. The relative balance in vector magnitudes indicates that each domain contributes comparably to the fusion, which facilitates overall improvement in the fused model's performance.

Table 3 demonstrates the advantages of approximate orthogonality between subspaces by comparing the performance of individual experts with fused pairs of approximately orthogonal expert models. The results show that fused pairs consistently outperform single-domain experts. This orthogonality ensures the relative independence of each adapter's update direction, effectively facilitating efficient knowledge integration across domains.

4.4 Limitations of Experts in Merging

Figure 5 presents the weight distribution histograms of the base model and the LoRA fine-tuned model on physics (Others in the Appendix). LoRA weights are highly sparse, indicating that the adjustments made to the original model parameters are extremely limited. Further quantification through SVD reveals a significantly heavy-tailed singular value distribution: only 0.195% lie between e^{-1} and e^{-2}, while most are below e^{-9}. This suggests LoRA fine-tuning concentrates parameter changes in a few directions, with minimal contribution to the overall behavior of the model.

To validate the effect of parameter constraints on model outputs, we perform token-level KL divergence analysis over concatenated input and answer sequences, comparing the output distributions of the base and LoRA models. The left of Figure 6 shows that tokens with the largest KL divergence are mostly domain-independent, suggesting LoRA improves task performance by adjusting a few parameters rather than reconstructing the output distribution with new knowledge.

To further validate these findings, we compare perplexity (PPL) across physics, chemistry, and biology. For each domain, we randomly select 40 questions and generate responses using both the LLaMA3.1-8B-Instruct base model and the LoRA fine-tuned model. As shown in the right part of Figure 6, the LoRA model generally exhibits lower perplexity than the base model, indicating enhanced accuracy and stability in the target domain. The results suggest that LoRA fine-tuning refines the model's inherent capabilities by optimizing a low-dimensional manifold in parameter space, rather than extending its knowledge boundary.

Method	CoLA	MNLI	MRPC	QNLI	QQP	RTE	SST-2	Avg.
Weight Averaging	55.0	55.1	51.0	57.6	76.7	44.8	52.5	56.1
Fisher Merging	54.8	58.0	39.5	63.3	81.5	49.1	64.7	58.7
RegMean	61.7	70.4	65.4	69.7	78.8	56.0	79.7	68.8
Ties-Merging	**68.4**	71.4	**68.4**	**69.6**	**82.4**	47.7	**81.8**	70.0
+RHT	68.2	**74.3**	68.1	69.4	**82.4**	**48.0**	**81.8**	**70.3**
Task Arithmetic	68.7	68.6	**69.6**	**70.5**	**81.8**	**47.3**	83.6	70.0
+RHT	**68.8**	**72.0**	**69.6**	70.4	81.7	**47.3**	**84.3**	**70.6**
EMR-Merging	72.8	**81.1**	79.2	84.8	**88.1**	66.5	90.3	80.4
+RHT	**73.7**	80.4	**79.4**	**87.1**	**88.1**	**67.5**	**90.6**	**81.0**

Table 4: Multi-task performance when merging GPT-2 models on seven text classification tasks.

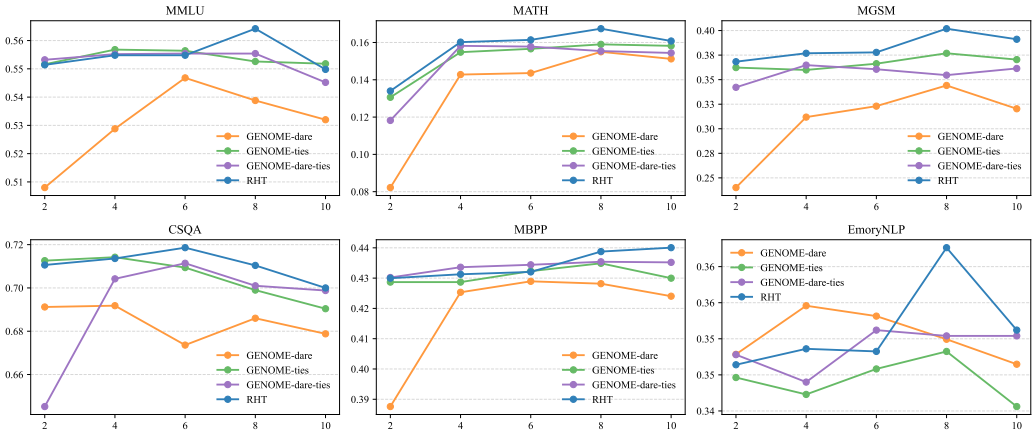

Figure 7: Performance trends across different numbers of merged expert models with RHT enhancement.

4.5 Results of Reparameterized Heavy-Tailed Method

Theorem 2 identifies the fundamental cause of performance saturation as the limited growth in the geometric complexity of the merged parameter set, measured via Gaussian Width. Theorem 4 establishes that, through reparameterization and scale mixing, the merged model inherently develops power-law heavy-tailed behavior, thereby overcoming the restrictive nature of Gaussian tails. Building on this result, Theorem 5 rigorously demonstrates that the parameter space associated with heavy-tailed distributions admits strictly greater coverage than that of exponential-tailed Gaussian counterparts. Such enhanced coverage endows the merged model with the capacity to represent a richer and more diverse class of functions, thereby strengthening its expressiveness and adaptability and enabling it to continuously integrate knowledge from additional experts without premature performance saturation.

To validate the effectiveness of RHT in practical model merging, we conduct experiments under both full-parameter fine-tuning and low-rank fine-tuning scenarios. As shown in Table 4, under the full-parameter fine-tuning setting, RHT effectively improves the performance of existing merging methods. Specifically, the average accuracy increases 0.3% compared to Ties-Merging, 0.6% compared to Task Arithmetic and EMR-Merging. Furthermore, in the scaling experiments based on LoRA fine-tuned experts (see Figure 7), RHT consistently outperforms other baseline methods across all tasks. Notably, when the performance of other methods approaches saturation, RHT continues to maintain steady improvement. In complex reasoning tasks such as MMLU, MATH, and MGSM, the performance gains of RHT are particularly pronounced, indicating its superior ability to leverage the growing number of experts and effectively avoid performance bottlenecks.

5 Conclusion

In this paper, we systematically investigate the fundamental limitations of model merging scalability through rigorous theoretical analysis and empirical evaluation. Our mathematical characterization, grounded in Gaussian Width, reveals an inherent pattern of concave diminishing returns in multi-expert ensembles, attributed to the saturation of the effective parameter space. The derived kinematic threshold provides a theoretical stopping criterion for the merging process. To address these limitations, we propose a Reparameterized Heavy-Tailed method that extends the coverage of merging parameters via heavy-tailed geometric reconstruction, resulting in sustained performance improvements.

Limitations

This paper focuses on merging experts derived from the same base architecture, which is the predominant setting in current model merging research. Heterogeneous merging across different architectures remains underexplored due to fundamental challenges such as parameter alignment, incompatible tokenization, and dimensional mismatches. As research in heterogeneous merging matures and standardized methodologies emerge, extending our theoretical framework to this setting presents a promising direction for future work. Our current analysis establishes foundational principles for homogeneous merging that may inform subsequent investigations into cross-architecture composition.

References

Takuya Akiba, Makoto Shing, Yujin Tang, Qi Sun, and David Ha. 2025. Evolutionary optimization of model merging recipes. *Nature Machine Intelligence*, pages 1–10.

Dennis Amelunxen, Martin Lotz, Michael B McCoy, and Joel A Tropp. 2014. Living on the edge: Phase transitions in convex programs with random data. *Information and Inference: A Journal of the IMA*, 3(3):224–294.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and 1 others. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Jiho Choi, Donggyun Kim, Chanhyuk Lee, and Seunghoon Hong. 2024. Revisiting weight averaging for model merging. *arXiv preprint arXiv:2412.12153*.

Zhuojun Ding, Wei Wei, and Chenghao Fan. 2025. Selecting and merging: Towards adaptable and scalable named entity recognition with large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9869–9886, Vienna, Austria. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. 2023. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19358–19369.

Shangbin Feng, Zifeng Wang, Yike Wang, Sayna Ebrahimi, Hamid Palangi, Lesly Miculicich, Achin Kulshrestha, Nathalie Rauschmayr, Yejin Choi, Yulia Tsvetkov, and 1 others. Model swarms: Collaborative search to adapt llm experts via swarm intelligence. In *Forty-second International Conference on Machine Learning*.

Lei Gao, Yue Niu, Tingting Tang, Salman Avestimehr, and Murali Annavaram. 2024. Ethos: Rectifying language models in orthogonal parameter space. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2054–2068.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague. Association for Computational Linguistics.

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*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. 2024a. Lorahub: Efficient cross-task generalization via dynamic lora composition. *Preprint*, arXiv:2307.13269.

Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. 2024b. Emr-merging: Tuning-free high-performance model merging. *Advances in Neural Information Processing Systems*, 37:122741–122769.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, and 1 others. 2023. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*.

Shankar Iyer, Nikhil Dandekar, Kornél Csernai, and 1 others. 2017. First quora dataset release: Question pairs.

Dong-Hwan Jang, Sangdoo Yun, and Dongyoon Han. 2024. Model stock: All we need is just a few fine-tuned models. In *European Conference on Computer Vision*, pages 207–223. Springer.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. 2023. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. *arXiv preprint arXiv:2310.11564*.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. 2023. Dataless knowledge fusion by merging weights of language models. In *The Eleventh International Conference on Learning Representations*.

Virginia Klema and Alan Laub. 1980. The singular value decomposition: Its computation and some applications. *IEEE Transactions on automatic control*, 25(2):164–176.

Brett W Larsen, Stanislav Fort, Nic Becker, and Surya Ganguli. How many degrees of freedom do we need to train deep networks: a loss landscape perspective. In *International Conference on Learning Representations*.

Sanwoo Lee, Jiahao Liu, Qifan Wang, Jingang Wang, Xunliang Cai, and Yunfang Wu. 2025a. Dynamic fisher-weighted model merging via bayesian optimization. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4923–4935.

Yu-Ang Lee, Ching-Yun Ko, Tejaswini Pedapati, I-Hsin Chung, Mi-Yen Yeh, and Pin-Yu Chen. 2025b. Star: Spectral truncation and rescale for model merging. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 496–505.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics.

Deyuan Liu, Zecheng Wang, Bingning Wang, Weipeng Chen, Chunshan Li, Zhiying Tu, Dianhui Chu, Bo Li, and Dianbo Sui. 2025a. Checkpoint merging via bayesian optimization in llm pretraining. *Preprint*, arXiv:2403.19390.

Enshu Liu, Junyi Zhu, Zinan Lin, Xuefei Ning, Shuaiqi Wang, Matthew B. Blaschko, Sergey Yekhanin, Shengen Yan, Guohao Dai, Huazhong Yang, and Yu Wang. 2025b. Linear combination of saved checkpoints makes consistency and diffusion models better. In *The Thirteenth International Conference on Learning Representations*.

Shuqi Liu, Han Wu, Bowei He, Xiongwei Han, Mingxuan Yuan, and Linqi Song. 2025c. Sens-merging: Sensitivity-guided parameter balancing for merging large language models. In *The 63rd Annual Meeting of the Association for Computational Linguistics*. ACL Anthology.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2024a. Gpt understands, too. *AI Open*, 5:208–215.

Yongkang Liu, Yiqun Zhang, Qian Li, Tong Liu, Shi Feng, Daling Wang, Yifei Zhang, and Hinrich Schütze. 2024b. Hift: A hierarchical full parameter fine-tuning strategy. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18266–18287.

Jinliang Lu, Ziliang Pang, Min Xiao, Yaochen Zhu, Rui Xia, and Jiajun Zhang. 2024. Merge, ensemble, and cooperate! a survey on collaborative strategies in the era of large language models. *Preprint*, arXiv:2407.06089.

Tongxu Luo, Jiahe Lei, Fangyu Lei, Weihao Liu, Shizhu He, Jun Zhao, and Kang Liu. 2024. Moelora: Contrastive learning guided mixture of experts on parameter-efficient fine-tuning for large language models. *arXiv preprint arXiv:2402.12851*.

Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D Bagdanov, and Joost van de Weijer. No task left behind: Isotropic model merging with common and task-specific subspaces. In *Forty-second International Conference on Machine Learning*.

Michael S Matena and Colin A Raffel. 2022. Merging models with fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35:17703–17716.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Ryan Po, Guandao Yang, Kfir Aberman, and Gordon Wetzstein. 2024. Orthogonal adaptation for modular customization of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7964–7973.

Akshara Prabhakar, Yuanzhi Li, Karthik Narasimhan, Sham Kakade, Eran Malach, and Samy Jelassi. 2025. Lora soups: Merging loras for practical skill composition tasks. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 644–655.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Alexandre Rame, Matthieu Kirchmeyer, Thibaud Rahier, Alain Rakotomamonjy, Patrick Gallinari, and Matthieu Cord. 2022. Diverse weight averaging for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 35:10821–10836.

Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. 2024. Ziplora: Any subject in any style by effectively merging loras. In *European Conference on Computer Vision*, pages 422–438. Springer.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, and 1 others. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations*.

Chongjie Si, Jingjing Jiang, and Wei Shen. 2025a. Unveiling the mystery of weight in large foundation models: Gaussian distribution never fades. *Preprint*, arXiv:2501.10661.

Chongjie Si, Kangtao Lv, Jingjing Jiang, Yadao Wang, Yongwei Wang, Xiaokang Yang, Wenbo Su, Bo Zheng, and Wei Shen. 2025b. Nan: A training-free solution to coefficient estimation in model merging. *Preprint*, arXiv:2505.16148.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with svd to tie the knots. In *The Thirteenth International Conference on Learning Representations*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158.

Mingxu Tao, Chen Zhang, Quzhe Huang, Tianyao Ma, Songfang Huang, Dongyan Zhao, and Yansong Feng. 2024. Unlocking the potential of model merging for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8705–8720.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Roman Vershynin. 2015. Estimation in high dimensions: a geometric perspective. In *Sampling Theory, a Renaissance: Compressive Sensing and Other Developments*, pages 3–66. Springer.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, page 353. Association for Computational Linguistics.

Hanqing Wang, Bowen Ping, Shuo Wang, Xu Han, Yun Chen, Zhiyuan Liu, and Maosong Sun. 2024. Loraflow: Dynamic lora fusion for large language models in generative tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12871–12882.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and 1 others. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pages 23965–23998. PMLR.

Xun Wu, Shaohan Huang, and Furu Wei. Mixture of lora experts. In *The Twelfth International Conference on Learning Representations*.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36:7093–7115.

Prateek Yadav, Tu Vu, Jonathan Lai, Alexandra Chronopoulou, Manaal Faruqui, Mohit Bansal, and Tsendsuren Munkhdalai. 2024. What matters for model merging at scale? *arXiv preprint arXiv:2410.03617*.

Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. 2024. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *Preprint*, arXiv:2408.07666.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. In *The Twelfth International Conference on Learning Representations*.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*.

Sayyed M Zahiri and Jinho D Choi. 2018. Emotion detection on tv show transcripts with sequence-based convolutional neural networks. In *AAAI Workshops*, volume 18, pages 44–52.

Fanhu Zeng, Haiyang Guo, Fei Zhu, Li Shen, and Hao Tang. 2025. Parameter efficient merging for multimodal large language models with complementary parameter adaptation. *Preprint*, arXiv:2502.17159.

Haobo Zhang and Jiayu Zhou. 2025. Unraveling lora interference: Orthogonal subspaces for robust model merging. *Preprint*, arXiv:2505.22934.

Jinghan Zhang, Junteng Liu, Junxian He, and 1 others. 2023. Composing parameter-efficient modules with arithmetic operation. *Advances in Neural Information Processing Systems*, 36:12589–12610.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Yiqun Zhang, Peng Ye, Xiaocui Yang, Shi Feng, Shufei Zhang, Lei Bai, Wanli Ouyang, and Shuyue Hu. 2025. Nature-inspired population-based evolution of large language models. *arXiv preprint arXiv:2503.01155*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, YeYanhan YeYanhan, and Zheyan Luo. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410.

A Proofs

Definition 1 (Gaussian Width (Vershynin, 2015)). *Let $S \subseteq \mathbb{R}^D$ be a subset of the D-dimensional Euclidean space. The Gaussian Width $w(S)$ of S is defined as:*

$$w(S) = \frac{1}{2}\mathbb{E}\left[\sup_{x,y\in S} \langle g, x-y\rangle\right], \tag{10}$$

where $g \sim \mathcal{N}(0, I_D)$ is a standard Gaussian random vector, and $\langle g, x-y\rangle$ represents the inner product between g and the difference $x-y$ between any two points x and y in S.

The Gaussian Width quantifies the extent to which the set S spans in random directions, thereby reflecting its geometric complexity.

Definition 2 (Statistical Dimension (Amelunxen et al., 2014)). *For a closed convex cone $C \subseteq \mathbb{R}^D$, its statistical dimension is expressed as:*

$$\delta(C) = \mathbb{E}\left[\|\Pi_C(g)\|_2^2\right], \tag{11}$$

where $g \sim \mathcal{N}(0, I_D)$ is a standard Gaussian random vector, $\Pi_C(g)$ is the projection of g onto the convex cone C.

Lemma 1 (Approximate Kinematics Theory (Amelunxen et al., 2014)). *For a closed convex cone $C \subseteq \mathbb{R}^D$, any k-dimensional subspace $S_k \subseteq \mathbb{R}^D$, and a Haar-distributed random orthogonal matrix Q:*

$$\begin{aligned} \delta(C) + k \lesssim D &\implies \Pr\{C \cap QS_k = \phi\} \approx 1, \\ \delta(C) + k \gtrsim D &\implies \Pr\{C \cap QS_k = \phi\} \approx 0. \end{aligned} \tag{12}$$

A.1 Proof of the Upper Bound Mode Merging

Proof of Theorem 1. According to the linear combination properties of Gaussian random variables, the merge parameter distribution is:

$$\theta_{\text{merge}} \sim \mathcal{N}(\mu_{\text{merge}}, \Sigma_{\text{merge}}). \tag{13}$$

The mean vector is:

$$\mu_{\text{merge}} = \sum_{i=1}^{n} \alpha_i \, \mu_i. \tag{14}$$

Covariance matrix:

$$\Sigma_{\text{merge}} = \sum_{i=1}^{n} \alpha_i^2 \, \sigma_i^2 \, I + \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} \alpha_i \, \alpha_j \, \mathrm{Cov}(\theta_i, \theta_j). \tag{15}$$

Define the covariance between experts i and j as:

$$\mathrm{Cov}(\theta_i, \theta_j) = \rho_{ij} \, \sigma_i \, \sigma_j \, I, \quad |\rho_{ij}| \leq 1. \tag{16}$$

Substituting the covariance into the covariance matrix expression above:

$$\Sigma_{\text{merge}} = \Big(\sum_{i=1}^{n} \alpha_i^2 \, \sigma_i^2 + \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} \alpha_i \, \alpha_j \, \rho_{ij} \, \sigma_i \, \sigma_j\Big) I. \tag{17}$$

Simplified to scalar variance:

$$\sigma^2_{\text{merge}} = \sum_{i=1}^{n} \alpha_i^2 \, \sigma_i^2 + \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} \alpha_i \alpha_j \, \rho_{ij} \, \sigma_i \, \sigma_j. \tag{18}$$

The merged variance in the simplified case is:

$$\sigma^2_{\text{merge}} = \sigma^2 \Big(\sum_{i=1}^{n} \alpha_i^2 + \rho \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} \alpha_i \alpha_j\Big). \tag{19}$$

Noting $(\sum_{i=1}^{n} \alpha_i)^2 = \sum_{i=1}^{n} \alpha_i^2 + \sum_{i=1}^{n} \sum_{j=1, i\neq j} \alpha_i \alpha_j = 1$, we get:

$$\sigma^2_{\text{merge}} = \sigma^2 \big(\rho + (1-\rho) \sum_{i=1}^{n} \alpha_i^2\big). \tag{20}$$

In the uniform weight case $\alpha_i = 1/n$, the variance is:

$$\sigma^2_{\text{merge}} = \sigma^2 \Big(\rho + \tfrac{1-\rho}{n}\Big). \tag{21}$$

When the number of experts $n \to \infty$, the variance after merging tends to:

$$\lim_{n\to\infty} \sigma^2_{\text{merge}} = \sigma^2 \, \rho. \tag{22}$$

This shows that no matter how many models are merged, the variance cannot be lower than $\sigma^2 \, \rho$, that is, there is a theoretical lower bound $\sigma^2 \, \rho$. When the models are completely independent ($\rho = 0$), theoretically increasing the number of models can reduce the variance infinitely. However, in reality, there is usually a correlation between models ($\rho > 0$), so there is an upper bound to merge.

Because the variance has a lower bound, we consider an adaptive stopping condition based on the marginal variance drop when adding an additional expert. Specifically, we require that the drop from $n-1$ to n experts satisfies:

$$\sigma^2_{\text{merge}}(n-1) - \sigma^2_{\text{merge}}(n) \geq \Delta, \tag{23}$$

where $\Delta > 0$ is a pre-defined threshold.

According to Equation 21, we can get:

$$\sigma^2_{\text{merge}}(n-1) - \sigma^2_{\text{merge}}(n) = \frac{\sigma^2(1-\rho)}{n(n-1)}. \quad (24)$$

Thus, the stopping condition yields:

$$n(n-1) \leq \frac{\sigma^2(1-\rho)}{\Delta}. \quad (25)$$

Therefore, the maximum number of models that can be merged is:

$$n_{\max} = \left\lfloor \sqrt{\frac{\sigma^2(1-\rho)}{\Delta} + \frac{1}{4}} + \frac{1}{2} \right\rfloor. \quad (26)$$

This indicates that there exists an upper bound on the number of models that can be merged, and this bound is mainly determined by the variance σ^2, the correlation ρ between models, and the desired variance drop threshold Δ.

□

A.2 Proof of the Gaussian Width of the Merged Model Subspace

Proof of Theorem 2. Let $L(\theta^*)$ denote the loss of the model obtained by merging all available experts (i.e., n experts). The objective of model merging is to determine the minimal merging size M such that the merged model achieves optimal merging performance. This objective is formulated as the following constrained minimization problem:

$$\min_{M \in \mathbb{Z}^+} M \quad \text{s.t.} \exists \theta \in S(\epsilon), L(\theta) \leq L(\theta^*) + \epsilon, \quad (27)$$

where $S(\epsilon)$ is the set of admissible parameter configurations for an M-expert model. This set is formally defined by the performance constraint:

$$S(\epsilon) = \{\theta \in \mathbb{R}^D : L(\theta) \leq L(\theta^*) + \epsilon\}. \quad (28)$$

Here, ϵ is the performance tolerance threshold. Consider merging n expert models with weights θ^* and a loss function $L(\theta)$, where θ lies in a small neighborhood of θ^*, we approximate $L(\theta)$ using a second-order Taylor expansion. Given that the first derivative of $L(\theta)$ at θ^* is zero, Equation 28 can be reformulated as:

$$S(\epsilon) = \left\{\theta \in \mathbb{R}^D : (\theta - \theta^*)^T H (\theta - \theta^*) \leq 2\epsilon\right\}, \quad (29)$$

where H is the Hessian matrix of $L(\theta)$ at θ^*. Since H is positive definite, $S(\epsilon)$ forms an ellipsoid centered at θ^*.

We then perform a linear transformation $z = H^{\frac{1}{2}}(\theta - \theta^*)$ to express:

$$S(\epsilon) = \left\{z \in \mathbb{R}^D \mid \|z\|^2 \leq 2\epsilon\right\}. \quad (30)$$

From Equation 28, we have:

$$\sup_{\theta \in S(\epsilon)} \langle g, \theta - \theta^* \rangle = \sup_z \langle g, H^{-\frac{1}{2}} z \rangle \text{s.t.} \|z\|^2 \leq 2\epsilon, \quad (31)$$

which is maximized by:

$$z^* = \sqrt{2\epsilon} \cdot \frac{H^{-\frac{1}{2}} g}{\|H^{-\frac{1}{2}} g\|}. \quad (32)$$

Thus, the Gaussian Width becomes:

$$w(S(\epsilon)) = \mathbb{E}\left[\sqrt{2\epsilon} \cdot \|H^{-\frac{1}{2}} g\|\right]. \quad (33)$$

By applying Jensen's inequality, we approximate the expected value as:

$$\mathbb{E}\left[\|H^{-\frac{1}{2}} g\|\right] \approx \sqrt{\text{Tr}(H^{-1})}. \quad (34)$$

Hence, the final Gaussian Width is:

$$w(S(\epsilon)) \approx \sqrt{2\epsilon \cdot \text{Tr}(H^{-1})}. \quad (35)$$

For the number of experts M, the Gaussian Width becomes:

$$w(S_M) \approx \sqrt{2\epsilon \cdot \sum_{i=1}^{M} \frac{1}{\lambda_i}}, \quad (36)$$

where λ_i is the i-th eigenvalue of H. The marginal contribution of adding the M-th expert is:

$$\begin{aligned} \Delta w_M &= w(S_M) - w(S_{M-1}) \\ &= \sqrt{2\epsilon \cdot \sum_{i=1}^{M} \frac{1}{\lambda_i}} - \sqrt{2\epsilon \cdot \sum_{i=1}^{M-1} \frac{1}{\lambda_i}}. \end{aligned} \quad (37)$$

Since the square root function is concave, the marginal gain decreases as M increases:

$$\Delta w_M > \Delta w_{M+1}. \quad (38)$$

Thus, diminishing marginal return arises from the concavity of the square root function, leading to progressively smaller contributions from each additional expert to the overall Gaussian Width. □

A.3 Proof of Parameter Redundancy Effects

Proof of Theorem 3. Let the weight of the merged model of M experts be θ^k, which represents a k-sparse vector containing exactly k non-zero parameters. Let θ^* be the weight vector obtained by merging all expert models. We decompose θ^* into two parts:

- $\theta^k = [\theta_1^*, \theta_2^*, \ldots, \theta_k^*]$, which represents the parameters contributed by M expert models ($M \leq N$).
- $\theta' = [\theta_{k+1}^*, \theta_{k+2}^*, \ldots, \theta_d^*]$, which represents the parameters contributed by the remaining $N - M$ expert models.

Given θ^k, the sublevel set of the loss function is defined by:

$$S(\theta', \epsilon) = \left\{ \theta' \in \mathbb{R}^{d-k} : L([\theta^k, \theta']) \leq L(\theta^*) + \epsilon \right\}. \tag{39}$$

To demonstrate the existence of parameter redundancy in the model merging process, we need to show that there exists a θ^k such that the zero vector $0 \in \mathbb{R}^{d-k}$ belongs to $S(\theta', \epsilon)$.

Next, consider the statistical dimension of the projection cone of the set $S(\theta', \epsilon)$. The statistical dimension of the projection cone is closely related to the geometric structure of the set. Using Lemma 1, we aim to prove that the statistical dimension of the projection cone of $S(\theta', \epsilon)$ is full, meaning its dimension is $d - k$.

Let $C = p(S(\theta', \epsilon))$ represent the result of projecting the set $S(\theta', \epsilon)$ onto the unit sphere S^{d-1}. According to existing research (Amelunxen et al., 2014), there is the following relationship between statistical dimension and Gaussian Width:

$$w^2(C) \leq \delta(C) \leq w^2(C) + 1. \tag{40}$$

Therefore, the relationship between the projected Gaussian Width $w(p(S(\theta', \epsilon)))$ and statistical dimension is:

$$w(p(S(\theta', \epsilon)))^2 \gtrsim d - k. \tag{41}$$

From Equation 29, we know that $S(\theta', \epsilon)$ is an ellipsoid, and all points $x \in S(\theta', \epsilon)$ are projected onto the unit sphere S^{d-1}, with the projection operation given by:

$$p(S(\theta', \epsilon)) = \left\{ \frac{x - \theta^k}{\|x - \theta^k\|} : x \in S(\theta', \epsilon) \right\}. \tag{42}$$

According to Equation 35, the Gaussian Width of the ellipsoid $w(S(\epsilon))$ is approximately:

$$w(S(\epsilon))^2 \approx 2\epsilon \mathrm{Tr}(H^{-1}) = 2\epsilon \sum_{i=1}^{d} \frac{1}{\lambda_i} = \sum_{i=1} r_i^2, \tag{43}$$

From (Larsen et al.), we modify r_i^2 to:

$$\frac{r_i^2}{\|\theta^* - \theta^k\|_2^2 + r_i^2}. \tag{44}$$

Therefore, the projected Gaussian Width is given by:

$$w(p(S(\theta', \epsilon)))^2 = \sum_{i=1}^{d-k} \frac{r_i^2}{\|\theta^* - \theta^k\|_2^2 + r_i^2}. \tag{45}$$

Here, $r_i = \sqrt{\frac{2\epsilon}{\lambda_i}}$ is the radius of the ellipsoid, and λ_i is the eigenvalue of the Hessian matrix of the loss function $L([\theta^k, \theta'])$ with respect to θ'.

From formulas 41 and 45, it can be observed that as the number of expert models increases, the number of non-zero parameters k in the network also increases, and the parameter θ^k approaches θ^*, which makes:

$$\frac{r_i^2}{\|\theta^* - \theta^k\|_2^2 + r_i^2} \approx 1. \tag{46}$$

In this case, the projected Gaussian Width will approach $d - k$, that is: $w(p(S(\theta', \epsilon)))^2 \approx d - k$. When each fraction $\frac{r_i^2}{\|\theta^* - \theta^k\|_2^2 + r_i^2}$ approaches 1, it means that the contribution from each direction is close to 1. At this point, the projected Gaussian Width will be close to:

$$w(p(S(\theta', \epsilon)))^2 = \sum_{i=1}^{d-k} 1 = d - k. \tag{47}$$

Thus, $0 \in S(\theta', \epsilon)$, meaning all the unmerged parameters become redundant. □

A.4 Proof of the Difference of Gaussian Distributions

Proof of Section 3.3. According to the properties of independent Gaussian random variables, their linear combination is still Gaussian, with the mean and variance given by the linear combination of the means and variances, respectively. Therefore,

$$\begin{aligned} \mathbb{E}[\mathbf{w}'] &= \mathbb{E}[\mathbf{w}] - \mathbb{E}[\mathbf{g}] = \boldsymbol{\mu} - \boldsymbol{\mu} = \mathbf{0}, \\ \mathrm{Var}[\mathbf{w}'] &= \mathrm{Var}[\mathbf{w}] + \mathrm{Var}[\mathbf{g}] \\ &= \sigma^2 \mathbf{I} + \sigma_g^2 \mathbf{I} = (\sigma^2 + \sigma_g^2)\mathbf{I}. \end{aligned} \tag{48}$$

Thus, $\mathbf{w}' \sim \mathcal{N}(\mathbf{0}, (\sigma^2 + \sigma_g^2)\mathbf{I})$. □

A.5 Proof of Heavy-Tailed Emergence From Expert Merging

Proof of Theorem 4. Let $\mathbf{w}' \sim \mathcal{N}(\mathbf{0}, (\sigma^2 + \sigma_g^2)\mathbf{I})$ be a zero-mean multivariate Gaussian distribution. Consider the nonlinear transformation:

$$T(w_i') = \text{sign}(w_i') \cdot |w_i'|^\gamma \cdot \left(1 + \alpha e^{-\beta |w_i'|}\right), \tag{49}$$

where $0 < \gamma < 1$, $\alpha > 0$, and $\beta > 0$. For simplicity, denote:

$$T(x) = \text{sign}(x) \cdot |x|^\gamma \cdot \left(1 + \alpha e^{-\beta |x|}\right). \tag{50}$$

When $|x|$ is sufficiently large, we have $e^{-\beta|x|} \to 0$, hence:

$$T(x) \approx \text{sign}(x) \cdot |x|^\gamma. \tag{51}$$

Consequently, for $Y = T(X)$,

$$|Y| \approx |X|^\gamma, \quad \text{i.e.} \quad |X| \approx |Y|^{1/\gamma}. \tag{52}$$

Therefore, for large y,

$$P(|Y| > y) \approx P(|X| > y^{1/\gamma}). \tag{53}$$

For a standard Gaussian random variable $X \sim \mathcal{N}(0, \sigma^2)$, the tail probability satisfies the well-known asymptotic relation:

$$P(|X| > t) \sim \frac{2\sigma}{\sqrt{2\pi}\, t} \exp\left(-\frac{t^2}{2\sigma^2}\right), \quad t \to \infty. \tag{54}$$

Setting $t = y^{1/\gamma}$ yields:

$$P(|X| > y^{1/\gamma}) \sim \frac{2\sigma}{\sqrt{2\pi}} y^{-1/\gamma} \exp\left(-\frac{y^{2/\gamma}}{2\sigma^2}\right). \tag{55}$$

Let $C = 2\sigma/\sqrt{2\pi}$, the tail probability for a single expert becomes:

$$P(|Y| > y) \sim C\, y^{-1/\gamma} \exp\left(-\frac{y^{2/\gamma}}{2\sigma^2}\right). \tag{56}$$

Now consider the merging of multiple experts with heterogeneous scale parameters σ. Define:

$$\begin{aligned} \theta &:= \frac{1}{2\sigma^2}, \Rightarrow \exp\left(-y^{2/\gamma}/2\sigma^2\right) \\ &= \exp(-\theta y^{2/\gamma}). \end{aligned} \tag{57}$$

Let the probability density of θ be $g_\theta(\theta)$. Then the merged model's survival function (tail probability) can be approximated by:

$$\begin{aligned} S(y) &= P(|Y| > y) \\ &\approx y^{-1/\gamma} \int_0^\infty \frac{1}{\sqrt{\pi\theta}} e^{-\theta y^{2/\gamma}}\, g_\theta(\theta)\, d\theta. \end{aligned} \tag{58}$$

Define:

$$\varphi(\theta) := \frac{1}{\sqrt{\pi\theta}}\, g_\theta(\theta), \quad u := y^{2/\gamma}, \tag{59}$$

thus:

$$\begin{aligned} S(y) &= y^{-1/\gamma}\, \mathcal{L}\{\varphi\}(u), \\ \mathcal{L}\{\varphi\}(u) &:= \int_0^\infty e^{-\theta u} \varphi(\theta)\, d\theta. \end{aligned} \tag{60}$$

By the Tauberian theorem, as $y \to \infty$ (i.e., $u \to \infty$), the asymptotic behavior of $\mathcal{L}\{\varphi\}(u)$ is determined by the behavior of $\varphi(\theta)$ as $\theta \to 0^+$. If there exist $\delta > 0$ and a constant $C' > 0$ such that:

$$\varphi(\theta) \sim C'\, \theta^{\delta - 1}, \quad \theta \to 0^+, \tag{61}$$

then:

$$\mathcal{L}\{\varphi\}(u) \sim C'\, \Gamma(\delta)\, u^{-\delta}, \quad u \to \infty. \tag{62}$$

Therefore,

$$\begin{aligned} S(y) &\sim C'\, \Gamma(\delta)\, y^{-1/\gamma}\, (y^{2/\gamma})^{-\delta} \\ &= \big(C'\, \Gamma(\delta)\big)\, y^{-(1+2\delta)/\gamma}. \end{aligned} \tag{63}$$

That is, the tail probability obeys a power-law decay:

$$P(|Y| > y) \propto y^{-\kappa}, \quad \kappa = \frac{1 + 2\delta}{\gamma}. \tag{64}$$

This establishes that if the mixing distribution $g_\theta(\theta)$ behaves as a power law near $\theta = 0$, then the merged model exhibits heavy-tailed behavior. □

A.6 Proof of Heavy-Tailed Distributions Expanding the Model Function Space

Proof of Theorem 5. In the model merging paradigm, standard weighted averaging (i.e., Gaussian mixtures) yields light-tailed models with exponential decay. By contrast, introducing a two-stage nonlinear transformation with scale mixing leads to heavy-tailed power-law behavior (Equation 64). We next prove that heavy-tailed distributions cover a much broader parameter space than exponential-tailed ones, offering greater applicability and expressive power.

Let X be a random variable with survival function:

$$S(x) = P(|X| > x). \tag{65}$$

Exponential-tailed distributions are those whose tail decays at least as fast as an exponential function.

A prototypical example is the Gaussian distribution, whose survival function satisfies:

$$S(x) \sim \exp(-ax^b), \quad a > 0,\, b \geq 1. \tag{66}$$

Heavy-tailed distributions decay slower than any exponential function and often follow a power-law decay:

$$S(x) \sim cx^{-\alpha}, \quad \alpha > 0, \tag{67}$$

where α is the tail index. A natural distribution family interpolating between these behaviors is the Student's t-distribution, parameterized by its degrees of freedom $\nu > 0$:

(a) **Heavy tail for finite ν**

For any finite $\nu \in (0, \infty)$, when $x \to \infty$, the probability density function (PDF) satisfies:

$$f(x;\nu) \propto \left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} \sim x^{-(\nu+1)}. \tag{68}$$

Integrating the PDF, the survival function satisfies:

$$S(x;\nu) \sim cx^{-\nu}, \tag{69}$$

which is a power-law decay, confirming that any Student's t-distribution with finite degrees of freedom is heavy-tailed.

(b) **Exponential tail in the limit $\nu \to \infty$**

It is a classical result that as $\nu \to \infty$, the Student's t-distribution converges in distribution to a standard normal distribution:

$$\lim_{\nu\to\infty} f(x;\nu) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}. \tag{70}$$

The Gaussian distribution is a canonical example of an exponential-tailed distribution.

In the family of t-distributions, the set of parameters corresponding to heavy-tailed behavior is the entire positive real axis: $P_{\text{Heavy}} = (0, \infty)$, which is a continuous open interval containing infinitely many points. In contrast, the parameter corresponding to an exponential-tailed distribution corresponds only to the limiting case of the degrees of freedom $\nu \to \infty$, which is a boundary point of the parameter space: $P_{\text{Exponential}} = \{\infty\}$. Therefore, compared to exponential-tailed distributions, heavy-tailed distributions have a significantly larger coverage in the parameter space, offering greater expressiveness and applicability.

□

B Experimental Setup

To obtain the expert models, we fine-tune the base models using LoRA with the LLaMA-Factory framework (Zheng et al., 2024). The complete training hyperparameters and configurations for this process are provided in Table 5.

B.1 Evaluation Metrics

To comprehensively assess the performance of our model across various tasks, we employ a set of widely adopted evaluation metrics tailored to different data categories, as summarized in Table 6. For **general-purpose** tasks such as question answering, classification, and coding, accuracy, weighted-F1, and pass@1 are used to evaluate correctness and robustness.

For **knowledge-intensive** tasks, including title generation and text translation, we adopt several text-level evaluation metrics to capture fluency, faithfulness, and semantic similarity. Specifically, BLEU (Papineni et al., 2002) measures n-gram overlap between generated and reference texts; ROUGE (Fang et al., 2023) evaluates recall-oriented summarization quality; BERTScore (Zhang et al.) leverages contextual embeddings to assess semantic similarity; and F1 score balances precision and recall.

B.2 Baselines

Weight Averaging takes a simple linear average of the weights from each expert model, where each model is assigned an equal coefficient.

Fisher Merging (Matena and Raffel, 2022) leverages the Fisher information matrix to assign importance weights to parameters during merging. Parameters with higher Fisher information scores, which indicate greater sensitivity to model outputs, receive proportionally larger influence in the final merged model.

RegMean (Jin et al., 2023) formulates model merging as a regularization-based optimization problem that operates without training data. It regularizes the merged weights to stay close to the original expert models while minimizing prediction disagreements, effectively balancing contributions from each expert.

Task Arithmetic (Ilharco et al.) is a widely used training-free model merging baseline. Its core idea is to compute the weight differences between a fine-tuned model and a pretrained model (i.e., the "task

vectors") and linearly combine these task vectors to enable model editing and multi-task fusion.

TIES-Merging (Yadav et al., 2023) aims to introduce sparsification and sign-alignment strategies in the process of model parameter merging to mitigate weight conflicts and information cancellation. It employs a "Trim, Elect Sign, Disjoint Merge" procedure: first removing redundant parameters, then averaging only the weights with consistent signs, and selecting a dominant direction for weights with inconsistent signs, thereby effectively improving the stability and performance of the merged model.

DARE (Yu et al., 2024) operates by randomly dropping a large portion of weights within the task vector and subsequently applying a proportional scaling to the remaining weights.

EMR-Merging (Huang et al., 2024b) represents a recent advancement in tuning-free model merging. While traditional methods like Task Arithmetic and TIES-Merging directly merge model weights, EMR-Merging takes a different approach by decomposing the merging process into unified task vectors and task-specific modulators, allowing for dynamic adjustment during inference without additional training overhead.

Model Swarms (Feng et al.) A collaborative search algorithm designed to adapt large language model (LLM) experts using principles of swarm intelligence. Inspired by Particle Swarm Optimization (PSO), the method treats each LLM as a "particle" navigating the model weight space. Guided by a utility function and influenced by personal best, global best, and worst checkpoints, these expert models iteratively update their weights and directions to optimize for a target objective.

GENOME (Zhang et al., 2025) A population-based evolutionary framework for adapting large language models (LLMs) based on genetic optimization. Inspired by biological evolution, the method treats each LLM as an "individual" with parameters functioning as digital genes. A population of expert models evolves through three key operations: crossover, which merges weights from parent models; mutation, which introduces random perturbations to enhance diversity; and selection, which prioritizes high-performing individuals based on a fitness function.

C More Detailed Results

This section provides comprehensive experimental results that complement the main findings presented in the paper.

Table 7 presents the detailed performance of GENOME under DARE and TIES merging strategies when merging 2 to 10 LoRA experts on the D_{gend} corpus, demonstrating the existence of an upper bound on the number of experts that can be effectively merged. Table 8 reports complete results across five model merging methods (GENOME-DARE, GENOME-TIES, GENOME-DARE-TIES, Model Swarms, and RHT) over five random seeds on six datasets (MMLU, MATH, MGSM, CSQA, MBPP, and EmoryNLP).

Figure 8 visualizes the weight distributions of expert models from three representative methods: LoRA-Flow (Wang et al., 2024), LoraHub (Huang et al., 2024a), and GENOME (Zhang et al., 2025). The histograms demonstrate that expert weights across different domains consistently follow Gaussian distributions, which supports the theoretical assumptions underlying our analysis.

For the D_{knowd} corpus experiments, Tables 9 and 10 report the performance of Gemma-2-2B-it and LLaMA3.1-8B-Instruct on domain-specific title generation tasks (physics, chemistry, and biology) across five random seeds, evaluated using F1, ROUGE, BLEU, and BERT Score metrics. Table 11 presents the corresponding results for translation tasks using BLEU as the evaluation metric.

To investigate cross-domain merging effects, Tables 12, 13, and 14 analyze pairwise LoRA fusion experiments. Specifically, Table 12 examines physics title generation when merging physics experts with chemistry, biology, or other physics expert. Table 13 extends this analysis to chemistry and biology title generation tasks with cross-domain expert combinations. Table 14 reports the translation task results for all pairwise domain combinations (physics, chemistry, and biology), evaluated using BLEU scores across five random seeds.

Dataset	Samples	LoRA rank/alpha	Learning rate	Warmup ratio	Batchsize	Epochs
CoT	49747	8/16	2.00E-04	0.1	32	5
Code Alpaca	20016	8/16	2.00E-04	0.1	32	5
Flan v2	49123	8/16	2.00E-04	0.1	32	5
GPT4 Alpaca	19906	8/16	2.00E-04	0.1	32	5
Open Assistant 1	7331	8/16	2.00E-04	0.1	32	5
Open-Orca	29683	8/16	2.00E-04	0.1	32	5
Science Literature	7468	8/16	2.00E-04	0.1	32	5
ShareGPT	111912	8/16	2.00E-04	0.1	32	5
WizardLM	29810	8/16	2.00E-04	0.1	32	5
LIMA	1018	8/16	2.00E-04	0.1	32	5
Physics	19999	8/16	2.00E-04	0.1	32	5
Chemistry	19999	8/16	2.00E-04	0.1	32	5
Biology	20000	8/16	2.00E-04	0.1	32	5
Finance	100000	8/16	5.00E-05	0.1	32	5
Medicine	22000	8/16	2.00E-04	0.1	32	5

Table 5: Training configurations for different datasets. The learning rate decay follows a cosine schedule.

	Dataset	Category	Metrics	Size	
				vaild	test
General Purpose Data	CSQA	Question Answering	accuracy	200	1000
	EmoryNLP	Affective Computing	weighted-F1	200	697
	MATH	Mathematics	accuracy	200	1000
	MBPP	Code Generation	pass@1	200	774
	MGSM	Multilingual Processing	accuracy	200	2637
	MMLU	General Knowledge	accuracy	200	1000
	SST-2	Text Classification	accuracy	-	872
	QQP	Text Classification	accuracy	-	40430
	CoLA	Text Classification	accuracy	-	1043
	MNLI	Text Classification	accuracy	-	9832
	MRPC	Text Classification	accuracy	-	408
	QNLI	Text Classification	accuracy	-	5463
	RTE	Text Classification	accuracy	-	277
Knowledge Intensive Data	Physics_title	Title Generation	BERT Score, F1, ROUGE, BLEU	150	350
	Physics_trans	Text Translation	BLEU	150	350
	Chemistry_title	Title Generation	BERT Score, F1, ROUGE, BLEU	150	350
	Chemistry_trans	Text Translation	BLEU	150	350
	Biology_title	Title Generation	BERT Score, F1, ROUGE, BLEU	200	1077
	Biology_trans	Text Translation	BLEU	150	350

Table 6: Datasets and Evaluation Metrics for Benchmarking.

Model	MMLU	MATH	MGSM	CSQA	MBPP	EmoryNLP
DARE-2LoRA	50.80(1.1)	08.22(0.5)	24.01(1.0)	69.12(0.8)	38.75(1.3)	34.78(0.1)
DARE-4LoRA	52.88(0.9)	14.28(0.5)	31.19(1.6)	**69.18(0.9)**	42.53(0.7)	**35.45(0.7)**
DARE-6LoRA	**54.68(1.0)**	14.36(0.8)	32.30(1.0)	67.36(2.5)	**42.89(0.5)**	35.31(0.5)
DARE-8LoRA	53.88(0.9)	**15.50(0.5)**	**34.41(2.7)**	68.60(1.2)	42.81(0.5)	34.99(0.4)
DARE-10LoRA	53.20(1.6)	15.12(0.8)	32.04(1.2)	67.88(2.2)	42.40(0.4)	34.65(0.6)
TIES-2LoRA	55.16(0.5)	13.06(0.3)	36.23(0.2)	71.26(1.0)	42.87(0.2)	34.46(0.8)
TIES-4LoRA	**55.68(0.9)**	15.48(0.8)	35.98(0.7)	**71.42(0.8)**	42.87(0.3)	34.23(0.5)
TIES-6LoRA	55.64(0.7)	15.66(0.9)	36.62(0.7)	70.94(1.2)	43.36(0.3)	34.58(0.4)
TIES-8LoRA	55.26(1.0)	**15.90(0.3)**	**37.68(2.1)**	69.90(2.3)	**43.44(0.3)**	**34.82(0.4)**
TIES-10LoRA	55.18(1.1)	15.82(0.1)	37.04(1.6)	69.04(0.8)	42.99(2.0)	34.06(0.3)

Table 7: Performance of GENOME with 2–10 LoRA fusions on the D_{gend} corpus under the DARE and TIES merging strategy. Results averaged over 5 random seeds with standard deviations.

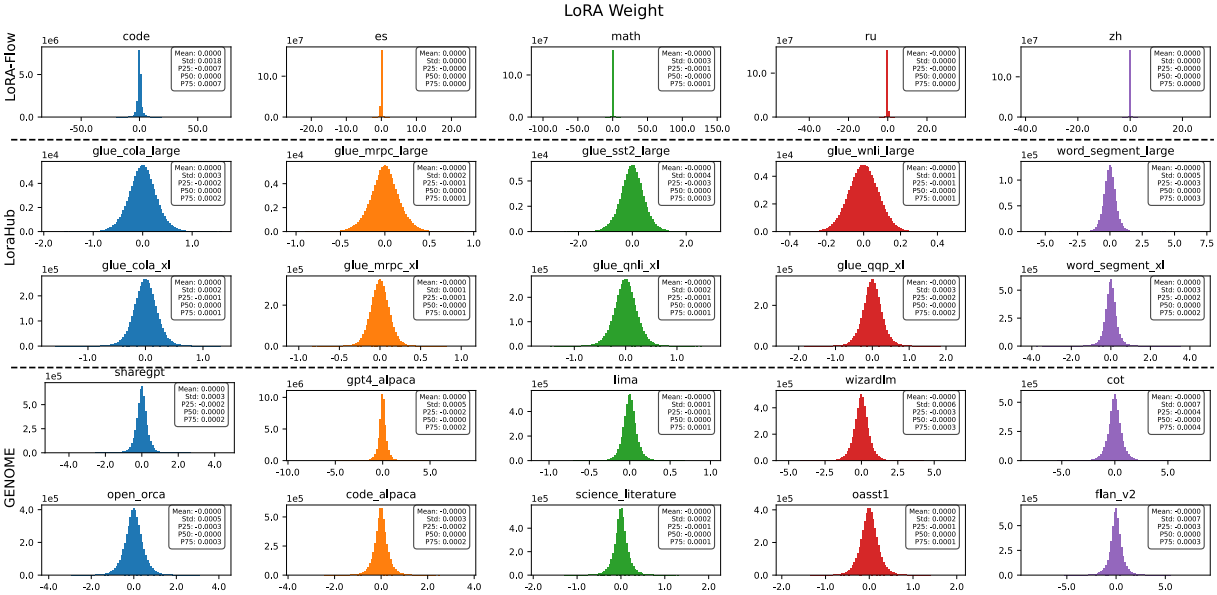

Figure 8: Histogram of expert model weights from LoRA-Flow (Wang et al., 2024), LoraHub (Huang et al., 2024a), GENOME (Zhang et al., 2025), demonstrating a Gaussian distribution.

Dataset	MMLU					MATH					MGSM				
Setting	2-LoRA	4-LoRA	6-LoRA	8-LoRA	10-LoRA	2-LoRA	4-LoRA	6-LoRA	8-LoRA	10-LoRA	2-LoRA	4-LoRA	6-LoRA	8-LoRA	10-LoRA
GENOME-DARE	0.5030	0.5370	0.5460	0.5300	0.5480	0.0880	0.1370	0.1360	0.1510	0.1480	0.2241	0.3026	0.3242	0.3136	0.3178
	0.5280	0.5300	0.5640	0.5280	0.5070	0.0800	0.1460	0.1340	0.1610	0.1540	0.2397	0.3003	0.3151	0.3735	0.3409
	0.5070	0.5180	0.5470	0.5480	0.5300	0.0770	0.1380	0.1460	0.1570	0.1460	0.2473	0.3140	0.3345	0.3303	0.3117
	0.5000	0.5390	0.5390	0.5460	0.5440	0.0860	0.1480	0.1520	0.1500	0.1440	0.2480	0.3386	0.3117	0.3311	0.3212
	0.5020	0.5200	0.5380	0.5420	0.5310	0.0800	0.1450	0.1500	0.1560	0.1640	0.2416	0.3038	0.3295	0.3720	0.3102
GENOME-TIES	0.5430	0.5610	0.5620	0.5680	0.5500	0.1340	0.1610	0.1650	0.1560	0.1590	0.3584	0.3694	0.3546	0.3883	0.3644
	0.5520	0.5630	0.5530	0.5510	0.5540	0.1320	0.1420	0.1650	0.1570	0.1570	0.3644	0.3595	0.3641	0.3925	0.3565
	0.5550	0.5570	0.5620	0.5400	0.5630	0.1310	0.1570	0.1480	0.1610	0.1570	0.3637	0.3489	0.3686	0.3955	0.3970
	0.5530	0.5420	0.5590	0.5530	0.5570	0.1250	0.1620	0.1470	0.1590	0.1580	0.3625	0.3587	0.3735	0.3557	0.3659
	0.5550	0.5610	0.5460	0.5510	0.5350	0.1310	0.1520	0.1580	0.1620	0.1600	0.3625	0.3629	0.3705	0.3523	0.3682
GENOME-DARE-TIES	0.5470	0.5640	0.5580	0.5630	0.5530	0.1130	0.1530	0.1630	0.1530	0.1590	0.3440	0.3510	0.3570	0.3640	0.3610
	0.5520	0.5480	0.5530	0.5630	0.5310	0.1160	0.1490	0.1500	0.1520	0.1410	0.3490	0.3750	0.3610	0.3500	0.3650
	0.5590	0.5580	0.5600	0.5490	0.5470	0.1250	0.1650	0.1590	0.1540	0.1590	0.3380	0.3630	0.3730	0.3470	0.3480
	0.5560	0.5530	0.5520	0.5450	0.5510	0.1210	0.1530	0.1660	0.1580	0.1570	0.3460	0.3630	0.3650	0.3590	0.3520
	0.5520	0.5530	0.5540	0.5570	0.5440	0.1160	0.1710	0.1510	0.1600	0.1560	0.3340	0.3720	0.3470	0.3530	0.3810
Swarms	0.5530	0.5560	0.5550	0.5510	0.5590	0.0940	0.1630	0.1570	0.1610	0.1620	0.3360	0.3720	0.3730	0.3560	0.3760
	0.5520	0.5570	0.5510	0.5510	0.5470	0.1010	0.1590	0.1530	0.1500	0.1400	0.3390	0.3740	0.3390	0.3360	0.3650
	0.5460	0.5570	0.5590	0.5430	0.5510	0.1030	0.1650	0.1600	0.1430	0.1430	0.3420	0.3770	0.3660	0.3530	0.3580
	0.5510	0.5720	0.5560	0.5700	0.5590	0.1040	0.1820	0.1550	0.1570	0.1530	0.3410	0.3510	0.3600	0.3860	0.3600
	0.5460	0.5430	0.5520	0.5410	0.5570	0.1030	0.1420	0.1550	0.1540	0.1540	0.3420	0.3590	0.3700	0.3340	0.3470
RHT	0.5540	0.5550	0.5560	0.5650	0.5450	0.1270	0.1580	0.1580	0.1590	0.1590	0.3641	0.3766	0.3769	0.3917	0.3970
	0.5520	0.5480	0.5540	0.5750	0.5600	0.1290	0.1480	0.1660	0.1680	0.1690	0.3743	0.3747	0.3542	0.3940	0.3944
	0.5500	0.5480	0.5540	0.5680	0.5600	0.1430	0.1630	0.1540	0.1690	0.1560	0.3713	0.3777	0.3557	0.4042	0.3834
	0.5510	0.5590	0.5570	0.5640	0.5430	0.1370	0.1680	0.1630	0.1740	0.1600	0.3584	0.3853	0.4050	0.3970	0.3970
	0.5500	0.5640	0.5530	0.5500	0.5410	0.1340	0.1640	0.1660	0.1670	0.1600	0.3735	0.3701	0.3970	0.4224	0.3838
Dataset	CSQA					MBPP					EmoryNLP				
Setting	2-LoRA	4-LoRA	6-LoRA	8-LoRA	10-LoRA	2-LoRA	4-LoRA	6-LoRA	8-LoRA	10-LoRA	2-LoRA	4-LoRA	6-LoRA	8-LoRA	10-LoRA
GENOME-DARE	0.7000	0.6950	0.6840	0.6720	0.6900	0.3837	0.4251	0.4251	0.4315	0.4238	0.3475	0.3589	0.3458	0.3455	0.3518
	0.6800	0.6830	0.6530	0.6900	0.6910	0.3979	0.4341	0.4302	0.4315	0.4225	0.3488	0.3612	0.3533	0.3479	0.3434
	0.6950	0.7060	0.6860	0.6750	0.6870	0.3708	0.4238	0.4367	0.4315	0.4315	0.3480	0.3541	0.3596	0.3555	0.3426
	0.6960	0.6890	0.6420	0.6960	0.6870	0.3811	0.4160	0.4276	0.4199	0.4225	0.3456	0.3558	0.3533	0.3503	0.3543
	0.6850	0.6860	0.7030	0.6970	0.6390	0.4044	0.4276	0.4251	0.4264	0.4199	0.3493	0.3428	0.3537	0.3504	0.3403
GENOME-TIES	0.7070	0.7110	0.7160	0.6920	0.6830	0.4276	0.4315	0.4315	0.4367	0.4354	0.3462	0.3373	0.3416	0.3413	0.3376
	0.7230	0.7180	0.7080	0.7090	0.6900	0.4276	0.4315	0.4380	0.4367	0.3967	0.3501	0.3496	0.3437	0.3477	0.3396
	0.7230	0.7030	0.7240	0.6630	0.6890	0.4315	0.4289	0.4328	0.4354	0.4496	0.3372	0.3392	0.3452	0.3534	0.3382
	0.7060	0.7150	0.7060	0.7210	0.7030	0.4276	0.4238	0.4341	0.4302	0.4367	0.3546	0.3407	0.3505	0.3494	0.3461
	0.7040	0.7240	0.6930	0.7100	0.6870	0.4289	0.4276	0.4315	0.4328	0.4315	0.3351	0.3446	0.3481	0.3494	0.3415
GENOME-DARE-TIES	0.6560	0.7030	0.7220	0.6970	0.7070	0.4320	0.4330	0.4340	0.4350	0.4320	0.3480	0.3330	0.3460	0.3510	0.3550
	0.6500	0.7010	0.7040	0.7030	0.7050	0.4240	0.4280	0.4340	0.4350	0.4370	0.3500	0.3490	0.3450	0.3530	0.3510
	0.6630	0.7110	0.6980	0.7110	0.6970	0.4330	0.4350	0.4370	0.4380	0.4350	0.3500	0.3470	0.3590	0.3500	0.3550
	0.6360	0.6950	0.7160	0.6990	0.6910	0.4300	0.4390	0.4320	0.4350	0.4300	0.3450	0.3420	0.3550	0.3550	0.3410
	0.6210	0.7110	0.7170	0.6950	0.6940	0.4320	0.4330	0.4350	0.4340	0.4420	0.3460	0.3490	0.3510	0.3430	0.3500
Swarms	0.6540	0.6920	0.6860	0.6770	0.7050	0.4320	0.4410	0.4430	0.4370	0.4340	0.3450	0.3330	0.3470	0.3510	0.3500
	0.6440	0.7110	0.6980	0.6860	0.7000	0.4350	0.4330	0.4300	0.4420	0.4300	0.3490	0.3590	0.3560	0.3570	0.3430
	0.6530	0.7160	0.7090	0.7120	0.7070	0.4370	0.4320	0.4340	0.4380	0.4330	0.3390	0.3460	0.3420	0.3600	0.3670
	0.6510	0.7000	0.6960	0.6800	0.6700	0.4380	0.4350	0.4280	0.4420	0.4370	0.3490	0.3420	0.3460	0.3490	0.3740
	0.6450	0.7030	0.6990	0.6830	0.6970	0.4330	0.4330	0.4300	0.4340	0.4420	0.3450	0.3480	0.3480	0.3480	0.3590
RHT	0.7080	0.7180	0.7210	0.7140	0.7100	0.4289	0.4315	0.4302	0.4432	0.4380	0.3470	0.3482	0.3524	0.3688	0.3597
	0.7080	0.6930	0.7150	0.7060	0.7190	0.4328	0.4289	0.4289	0.4406	0.4406	0.3436	0.3454	0.3435	0.3582	0.3514
	0.7120	0.7150	0.7300	0.7070	0.7050	0.4276	0.4328	0.4354	0.4315	0.4406	0.3468	0.3452	0.3477	0.3639	0.3456
	0.7130	0.7200	0.7120	0.7070	0.6530	0.4289	0.4289	0.4302	0.4419	0.4457	0.3465	0.3491	0.3537	0.3575	0.3531
	0.7120	0.7220	0.7150	0.7180	0.7130	0.4315	0.4341	0.4354	0.4367	0.4354	0.3480	0.3552	0.3440	0.3647	0.3463

Table 8: Performance of different model merging methods (GENOME-DARE, GENOME-TIES, GENOME-DARE-TIES, Model Swarms, and RHT) when merging 2 to 10 LoRA experts on MMLU, MATH, MGSM, CSQA, MBPP, and EmoryNLP. Results are reported over 5 experimental runs.

Phy-title											
3-LoRA				4-LoRA				5-LoRA			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.5671	0.4832	0.2177	0.8975	0.5662	0.4818	0.2221	0.8974	0.5534	0.4719	0.2067	0.8943
0.5717	0.4891	0.2241	0.8984	0.5729	0.4882	0.2229	0.8981	0.5626	0.4787	0.2175	0.8969
0.5588	0.4786	0.2141	0.8954	0.5459	0.4654	0.2084	0.8950	0.5464	0.4672	0.2209	0.8960
0.5673	0.4841	0.2194	0.8974	0.5629	0.4836	0.2199	0.8974	0.5521	0.4711	0.2132	0.8967
0.5705	0.4885	0.2225	0.8981	0.5621	0.4815	0.2225	0.8967	0.5466	0.4689	0.2126	0.8959
Chem-title											
3-LoRA				4-LoRA				5-LoRA			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.5917	0.5200	0.2469	0.9121	0.5881	0.5181	0.2398	0.9120	0.5848	0.5183	0.2356	0.9102
0.5911	0.5224	0.2420	0.9122	0.5903	0.5206	0.2343	0.9107	0.5909	0.5222	0.2449	0.9127
0.5907	0.5201	0.2426	0.9125	0.5914	0.5234	0.2404	0.9113	0.5904	0.5199	0.2457	0.9124
0.5926	0.5191	0.2457	0.9125	0.5902	0.5194	0.2431	0.9124	0.5765	0.5050	0.2258	0.9097
0.5892	0.5226	0.2459	0.9120	0.5881	0.5181	0.2398	0.9120	0.5770	0.4989	0.2273	0.9085
Bio-title											
3-LoRA				4-LoRA				5-LoRA			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.3973	0.3495	0.0627	0.8625	0.3974	0.3498	0.0622	0.8626	0.4016	0.3506	0.0644	0.8631
0.3983	0.3488	0.0630	0.8628	0.4015	0.3504	0.0653	0.8630	0.3954	0.3484	0.0623	0.8625
0.3978	0.3478	0.0632	0.8626	0.4010	0.3511	0.0658	0.8633	0.3980	0.3488	0.0628	0.8626
0.3988	0.3512	0.0653	0.8634	0.3932	0.3441	0.0628	0.8621	0.3792	0.3328	0.0567	0.8551
0.3978	0.3464	0.0633	0.8624	0.3857	0.3280	0.0563	0.8592	0.3979	0.3460	0.0619	0.8620

Table 9: The performance comparison of different LoRA fusion settings on Gemma-2-2B-it across various domain-specific tasks.

Phy-title											
3-LoRA				4-LoRA				5-LoRA			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.5620	0.4895	0.2508	0.8886	0.5614	0.4886	0.2562	0.8864	0.5752	0.4962	0.2568	0.8898
0.5620	0.4890	0.2506	0.8884	0.5588	0.4861	0.2489	0.8879	0.5666	0.4918	0.2536	0.8893
0.5629	0.4905	0.2533	0.8903	0.5683	0.4917	0.2536	0.8892	0.5721	0.4948	0.2540	0.8895
0.5641	0.4856	0.2510	0.8900	0.5639	0.4903	0.2534	0.8896	0.5687	0.4890	0.2541	0.8890
0.5620	0.4888	0.2517	0.8904	0.5620	0.4833	0.2503	0.8858	0.5716	0.4956	0.2574	0.8900
Chem-title											
3-LoRA				4-LoRA				5-LoRA			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.6206	0.5688	0.3193	0.9070	0.6192	0.5655	0.3122	0.9064	0.6120	0.5537	0.2950	0.9030
0.6176	0.5676	0.3166	0.9070	0.6181	0.5665	0.3116	0.9065	0.6153	0.5611	0.3081	0.9057
0.6165	0.5571	0.3034	0.9046	0.6193	0.5632	0.3091	0.9052	0.6163	0.5561	0.3002	0.9037
0.6155	0.5601	0.3103	0.9055	0.6196	0.5652	0.3103	0.9066	0.6153	0.5600	0.3066	0.9047
0.6180	0.5680	0.3182	0.9072	0.6199	0.5635	0.3080	0.9060	0.6180	0.5598	0.3048	0.9050
Bio-title											
3-LoRA				4-LoRA				5-LoRA			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.4029	0.3614	0.0712	0.8568	0.3977	0.3538	0.0713	0.8587	0.4000	0.3582	0.0710	0.8531
0.3996	0.3588	0.0713	0.8553	0.3975	0.3557	0.0688	0.8548	0.4008	0.3535	0.0696	0.8540
0.3965	0.3542	0.0712	0.8547	0.3986	0.3525	0.0701	0.8559	0.4004	0.3565	0.0695	0.8536
0.4013	0.3610	0.0715	0.8567	0.4044	0.3612	0.0728	0.8593	0.4015	0.3602	0.0716	0.8552
0.3983	0.3561	0.0705	0.8552	0.3987	0.3518	0.0697	0.8537	0.4023	0.3607	0.0703	0.8546

Table 10: The performance comparison of different LoRA fusion settings on LLaMA3.1-8B-Instruct across various domain-specific tasks.

Model	Phy-trans			Chem-trans			Bio-trans		
	3-LoRA	4-LoRA	5-LoRA	3-LoRA	4-LoRA	5-LoRA	3-LoRA	4-LoRA	5-LoRA
Gemma	0.5189	0.5199	0.5250	0.3872	0.3857	0.3816	0.4772	0.4772	0.4753
	0.5208	0.5088	0.5094	0.3820	0.3678	0.3840	0.4762	0.4781	0.4777
	0.5207	0.5250	0.5248	0.3843	0.3735	0.3807	0.4755	0.4748	0.4782
	0.5161	0.5233	0.5152	0.3808	0.3782	0.3856	0.4781	0.4753	0.4733
	0.5177	0.5129	0.5241	0.3770	0.3778	0.3807	0.4775	0.4746	0.4769
LLaMA	0.5218	0.5247	0.5251	0.3833	0.3805	0.3824	0.4629	0.4911	0.4860
	0.5237	0.5221	0.5213	0.3782	0.3821	0.3839	0.4627	0.4825	0.4857
	0.5240	0.5212	0.5223	0.3792	0.3763	0.3807	0.4635	0.4863	0.4870
	0.5219	0.5223	0.5152	0.3792	0.3766	0.3819	0.4702	0.4891	0.4888
	0.5218	0.5226	0.5203	0.3807	0.3818	0.3807	0.4668	0.4901	0.4879

Table 11: The performance comparison of different LoRA fusion settings on Gemma-2-2B-it and LLaMA3.1-8B-Instruct across various domain-specific tasks. The evaluation metric is BLEU.

Phy-title											
Phy+Chem				Phy+Bio				Phy1+Phy2			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.5607	0.4887	0.2533	0.8861	0.5610	0.4860	0.2503	0.8890	0.5608	0.4913	0.2564	0.8899
0.5618	0.4892	0.2549	0.8861	0.5622	0.4862	0.2497	0.8887	0.5581	0.4867	0.2579	0.8898
0.5662	0.4953	0.2607	0.8888	0.5587	0.4814	0.2465	0.8849	0.5635	0.4899	0.2552	0.8883
0.5607	0.4882	0.2538	0.8863	0.5634	0.4859	0.2515	0.8904	0.5625	0.4911	0.2582	0.8895
0.5622	0.4885	0.2549	0.8861	0.5623	0.4846	0.2513	0.8899	0.5590	0.4785	0.2471	0.8857

Table 12: Performance of pairwise LoRA fusion experiments across domains (Physics).

Chem-title							
Phy+Chem				Chem+Bio			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.6073	0.5491	0.2982	0.9021	0.6121	0.5532	0.3001	0.9052
0.6128	0.5579	0.3063	0.9051	0.6119	0.5556	0.3036	0.9056
0.6114	0.5620	0.3107	0.9053	0.6084	0.5552	0.3139	0.9063
0.6119	0.5580	0.3107	0.9047	0.6101	0.5551	0.3002	0.9055
0.6137	0.5650	0.3122	0.9057	0.6110	0.5552	0.3015	0.9051
Bio-title							
Phy+Bio				Chem+Bio			
F1	ROUGE	BLEU	BERT Score	F1	ROUGE	BLEU	BERT Score
0.4006	0.3581	0.0722	0.8570	0.3984	0.3539	0.0709	0.8587
0.4010	0.3591	0.0720	0.8572	0.4005	0.3575	0.0727	0.8587
0.3996	0.3577	0.0715	0.8583	0.4022	0.3594	0.0726	0.8589
0.4009	0.3586	0.0709	0.8570	0.4014	0.3588	0.0719	0.8594
0.4006	0.3593	0.0715	0.8571	0.3970	0.3473	0.0695	0.8564

Table 13: Performance of pairwise LoRA fusion experiments across domains (Chemistry and Biology).

Phy-trans			Chem-trans		Bio-trans	
Phy+Chem	Phy+Bio	Phy1+Phy2	Phy+Chem	Chem+Bio	Phy+Bio	Chem+Bio
0.5216	0.5215	0.5190	0.3762	0.3813	0.4614	0.4633
0.5205	0.5212	0.5185	0.3810	0.3811	0.4614	0.4705
0.5207	0.5212	0.5185	0.3805	0.3800	0.4637	0.4612
0.5215	0.5234	0.5162	0.3788	0.3848	0.4613	0.4655
0.5231	0.5207	0.5201	0.3827	0.3796	0.4599	0.4703

Table 14: Performance of pairwise LoRA fusion experiments across domains. The evaluation metric is BLEU.