Are we Merging the Right Models? Impact of Expert Training Duration on Model Merging for LLMs

Nikita Kozodoi, Zainab Afolabi, Jack Butler
AWS Forward Deployed Engineering

amazon | science

1 The Problem

- **Model merging** combines separately fine-tuned experts into one multi-task model
- It is a cheap alternative to pooling every domain's data or serving one model per skill
- Standard practice merges every expert at its own **best validation checkpoint T***
- The implicit claim: an expert contributes most when it is individually best

Research Question

We train experts from 0.25× to 5× T* and ask a simple question: **is the individual optimum the right place to stop for the merge?**

2 Overtraining Protocol

We train each expert well past T* and merge multiple models at every checkpoint.

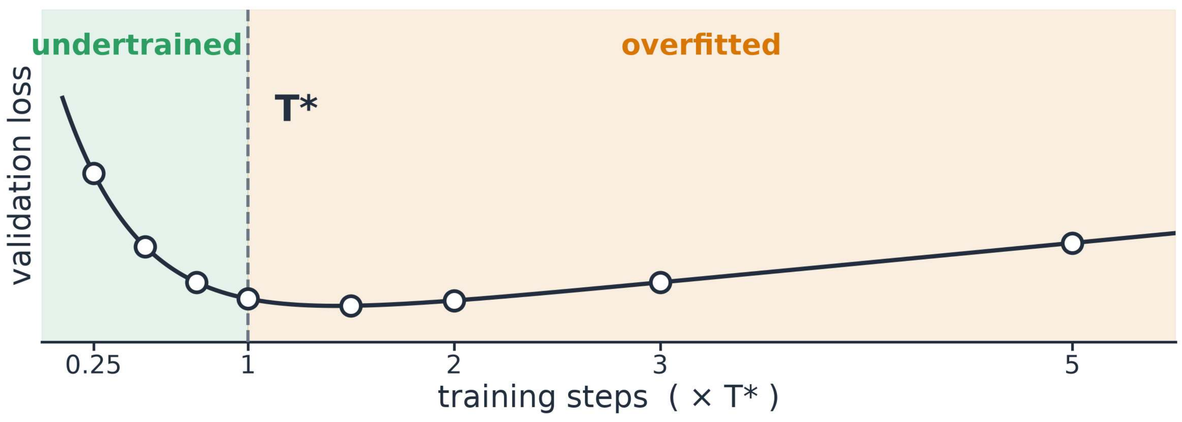

Figure 1. T* is found per domain and per model size, so every duration is a multiple of that expert's own optimum. We save 8 LoRA checkpoints spanning heavily undertrained to heavily overfitted.

3 Setup

Merging methods:

- **Simple Averaging:** equal-weight mean of all expert deltas
- **Task Arithmetic:** scales task vectors by one global coefficient
- **TIES:** trims small deltas, elects a sign, merges the agreeing values
- **DARE+TIES:** randomly drops and rescales deltas, then applies TIES
- **Greedy Soup:** adds an expert only if the held-out score improves

Experts and data:

- Qwen3.5 at 0.8B, 2B and 4B, fine-tuned with QLoRA (r = 16) on a 4-bit base
- 5 domains, one expert each, and all 5 adapters merged together
- Constant learning rate with no decay, so training stays active past T*
- 8 durations from 0.25× to 5× T*, scored per domain and averaged

Domain	Training data	Benchmark	Metric
Mathematics	DART-Math	GSM8K	Accuracy
Code	Magicoder	HumanEval	Pass@1
Instruction	TULU-3	IFEval	Accuracy
Multilingual	Aya	ARC	Accuracy
Safety	BeaverTails	HarmBench	Refusal rate

Table 1. Domains, datasets and benchmarks follow MergeBench.

Why overtraining might help:

- A random forest grows each tree deep on purpose: the trees are low-bias and high-variance, and averaging cancels the noise that depth adds
- TIES and DARE prune and resolve sign conflicts before averaging, so they have an explicit noise-suppression step. Simple Averaging does not

4 Where Each Method Wants to Stop

Method	0.8B	2B	4B	Regime
Simple Averaging	0.25–1.0×	0.25–0.5×	0.25–0.75×	undertrained
Greedy Soup	0.25–0.75×	0.25–2.0×	0.25–3.0×	undertrained
Task Arithmetic	0.25–3.0×	0.25–3.0×	0.25–2.0×	intermediate
TIES	1.5–3.0×	2.0–3.0×	3.0–5.0×	**overfitted**
DARE+TIES	2.0–5.0×	0.25–5.0×	3.0–5.0×	**overfitted**

Table 2. Optimal duration as a multiple of T*, across the four accuracy domains.

← **Read on arXiv**

Learn about our team →

5 Safety Follows the Same Pattern

Method	Best refusal rate	Peaks at
Simple Averaging	25.5%	1× T*
Greedy Soup	23.0%	1× T*
Task Arithmetic	50.0%	3× T*
DARE+TIES	61.5%	5× T*
TIES	**68.5%**	**3× T***

Table 3. Best refuse-to-answer rate on HarmBench, Qwen3.5-4B, across all eight durations.

The safety expert makes a localized parameter update. **Sparsification preserves it** even when other experts are heavily overtrained, while plain averaging diffuses it.

6 The Optimum is Method-Dependent

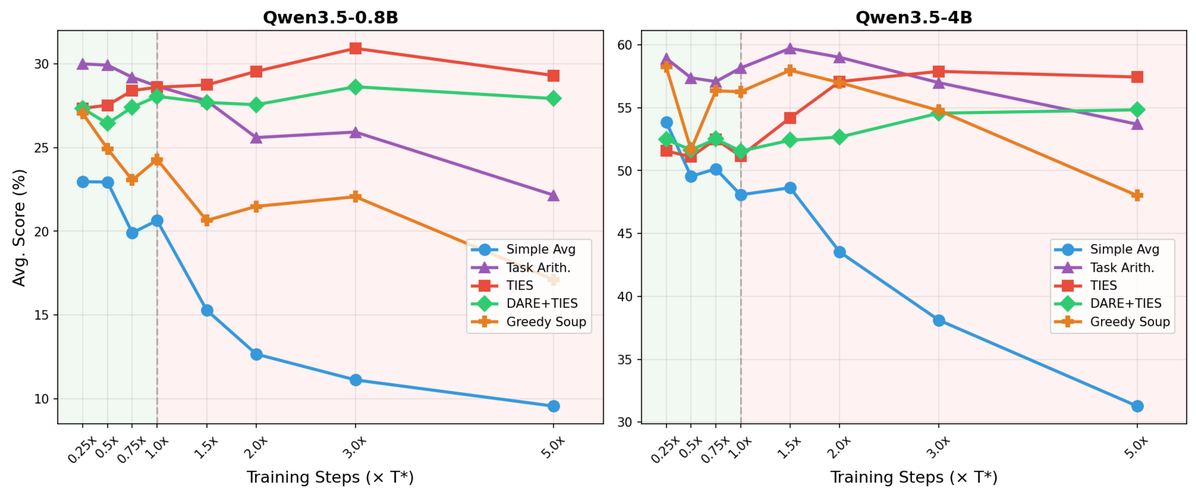

Figure 2. Merged accuracy against expert training duration, averaged over the four accuracy domains. Green band is undertrained, red is overfitted, dashed line is T*.

Simple Averaging falls past T*, losing 14 to 23 points by 5×. **TIES** and **DARE+TIES** peak well past it, beating their own T* score by 2.5 to 8.7 points.

Math shows the widest spread: Simple Averaging drops from 64.8% at 0.25× to 20.3% at 5×, while TIES climbs to 66.7% at 3×.

7 Why Overtraining Helps

Sparsification filters the added variance

Past T*, bias stabilizes as experts capture more task-specific signal, while variance and covariance climb. TIES and DARE trim that variance on the filtering step.

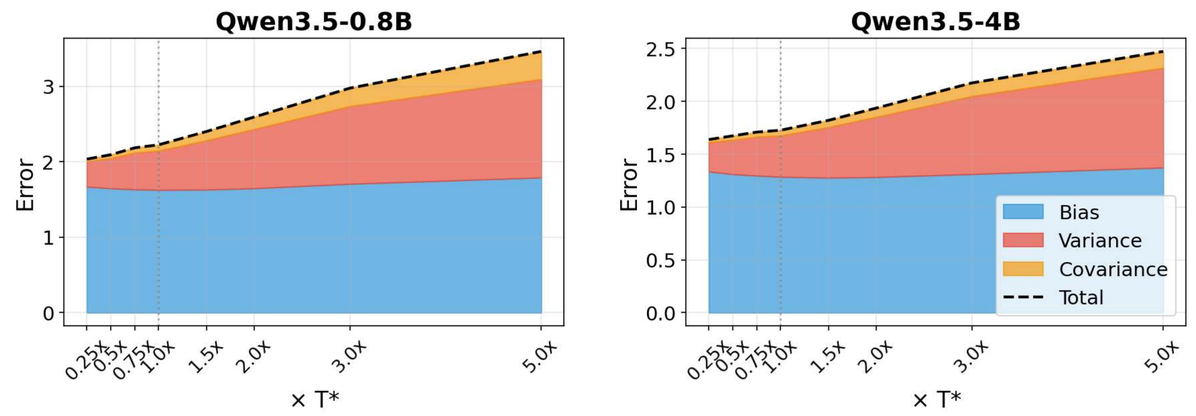

Figure 3. Bias-variance-covariance decomposition of the merged model error for 0.8B and 4B. The 2B panel in the paper shows the same trend.

Merging stays inside one basin

The loss barrier between expert pairs grows with training duration, but stays under 0.35 even at 5× T*. LoRA's low-rank constraint keeps every expert in the pre-trained basin.

8 Customer Impact

- Customers need several fine-tuned skills in **one deployable model**
- The fix changes only the **training steps per expert**: same architecture, deployment and inference cost
- Applies on Amazon SageMaker AI, Amazon Bedrock and Amazon Nova Forge

9 Take-Aways

Key Findings

- T* is almost never the best place to stop for the merge
- Simple Averaging wants undertrained experts, TIES and DARE want overfitted ones
- Using sparsification merging methods? Train experts past the validation optimum

Contact: kozodoi@amazon.com