

# A Reproducible Within-Family SLERP Merge of Llama-3.1-8B-Instruct and DeepSeek-R1-Distill-Llama-8B

Norika Oda

ORCID: 0009-0006-6816-9891

Independent researcher

[norika@charenix.com](mailto:norika@charenix.com)

2026-05-27

## Abstract

We document a reproducible weight-level composition of two same-family checkpoints (Llama-3.1-8B-Instruct and DeepSeek-R1-Distill-Llama-8B) using Spherical Linear Interpolation (the SLERP recipe) at [this link](#). The merge is performed via the open-source mergekit toolkit on a 188 GB CPU server in approximately three minutes. The resulting 8B-parameter model loads cleanly through the Hugging Face transformers library and generates coherent English on five out-of-distribution prompts spanning explanation, arithmetic, poetry, factual recall, and algebra. We pair this result with a companion negative finding on activation-level function vector injection between the same two checkpoints : weight-level SLERP produces a usable composed model, while naive single-layer activation steering does not transfer the donor’s reasoning capability. This contrast motivates a layered framework in which different composition mechanisms address different sub-problems of the cross-architecture composition challenge.

## Introduction

Composition of pretrained language models is an active area in the open-weight community. The mergekit toolkit catalyzed widespread experimentation, producing hundreds of thousands of merged variants on public hubs. Within-family weight-scale merges, including SLERP, DARE , and TIES , are the most accessible technique because they require no per-model engineering once the architectural contract matches.

This paper has a single modest goal. We document one reproducible same-family spherical interpolation merge in enough detail that any third party with a CPU server and a half hour of compute can replicate it. The artifact is a 15 GB 8B-parameter model that loads through standard tooling and produces coherent text on a small held-out prompt set.

No new technique is introduced. The recipe is mergekit’s stock spherical interpolation at [this link](#).

We then place this result alongside a paired negative finding from activation-level transfer to motivate the larger research agenda.

# Method

## Models

Donor and host are both Llama-3 family 8B checkpoints. **Llama-3.1-8B-Instruct** (NousResearch mirror of the Meta release) is the general instruction-tuned base. **DeepSeek-R1-Distill-Llama-8B** is the Llama-3.1-8B base further distilled on reasoning traces from a stronger teacher.

Both share architecture (32 transformer blocks, hidden dimension 4096, 32 query heads, 8 key-value heads, RoPE base 500,000), tokenizer (Llama-3 byte-pair encoding, 128,256 vocabulary), and most of pretraining. Within-family merging has no dimensional, positional encoding, or tokenizer obstacles.

## Spherical Interpolation Recipe

We use SLERP (Spherical Linear Interpolation) with interpolation parameter  $\alpha = 0.5$ , applied uniformly to all 32 blocks. The recipe is expressed in mergekit’s YAML format:

```
slices:
  - sources:
    - model: llama-3.1-8b-instruct
      layer_range: [0, 32]
    - model: deepseek-r1-distill-llama-8b
      layer_range: [0, 32]
merge_method: slerp
base_model: llama-3.1-8b-instruct
parameters:
```

```
t: 0.5  
dtype: bfloat16
```

SLERP preserves the magnitude of parameter vectors along the great-circle path between source vectors, which empirically produces smoother behavior than naive linear interpolation when the two source models are not too far apart in weight space.

## Infrastructure

Merge execution: `mergekit-yaml` (version 0.1.4) on a Linux server with Intel i7-13700E (16 cores, 24 threads), 188 GB RAM, no GPU. The merge consists of 1457 graph operations and completes in approximately three minutes.

Output: 15 GB `safetensors` in four shards plus standard tokenizer and config files, written to a target directory in the format expected by Hugging Face `from_pretrained`.

## Validation Protocol

We load the merged network via `AutoModelForCausalLM.from_pretrained` with `bfloat16` precision and CPU placement. We generate 80 tokens with greedy decoding for each of five test prompts covering different task types: technical explanation, arithmetic, creative writing, factual recall, and algebraic problem solving.

The blend is judged to have produced a usable artifact if all five generations are non-empty, carry alphabetic content, and avoid pathological repetition or random gibberish.

## Results

The merge completed without error in 2 minutes and 11 seconds of wall-clock time on the described CPU server. The resulting model loaded into memory through standard transformers tooling.

Five-prompt validation results (greedy decoding, max 80 tokens, untrimmed):

1. “Explain in one sentence what a transformer model does.”  
Output: “A transformer model is a type of neural network architecture that uses self-attention mechanisms to process sequential data...”

2. “If I have 8 apples and give 3 to my friend, how many do I have left?”  
Output: “To find out how many apples you have left after giving 3 to your friend, subtract 3 from the original number...”
3. “Write a haiku about autumn.”  
Output: “Autumn’s golden hue / Leaves fall, drift to the ground / Nature’s final dance.”
4. “What is the capital of France?”  
Output: “The capital of France is Paris.”
5. “Solve for x:  $2x + 5 = 17$ . Show your work.”  
Output: “To solve for x, I’ll first subtract 5 from both sides of the equation to isolate the term with x...”

All five generations are non-empty, contain alphabetic content, follow the prompted task format, and do not exhibit repetition collapse or random sampling artifacts. The third prompt produces correct haiku form (5–7–5 syllabic structure). The fifth prompt correctly identifies the algebraic operation to apply.

## Quantitative benchmark

We evaluate the merged model on the same 10-prompt GSM8K held-out subset used in the companion negative paper , with greedy decoding capped at 200 new tokens. Three-way comparison:

GSM8K accuracy on 10 held-out problems (m11 through m20), greedy decoding, 200-token cap. Reasoning-marker rate counts occurrences of explicit reasoning cues (“step by step”, “therefore”, “first”, and similar) per generation.

| Model                                 | Accuracy        | Mean markers/gen | Mean gen tokens |
|---------------------------------------|-----------------|------------------|-----------------|
| Llama-3.1-8B-Instruct (base A)        | 6/10<br>(60.0%) | 2.45             | —               |
| DeepSeek-R1-Distill-Llama-8B (base B) | 1/10<br>(10.0%) | 3.50             | 198             |
| SLERP (this paper)                    | 6/10<br>(60.0%) | 4.30             | 200             |

Three observations from the table:

**Zero merge tax in this configuration.** The merged artifact matches Llama-3.1-8B-Instruct exactly on accuracy. The SLERP at neither hurt nor helped the host’s mathematical problem-solving rate on this held-out set.

**Stylistic transfer from donor to merged.** The merged generation shows a reasoning-marker rate of 4.30 per output, substantially above Llama’s 2.45 baseline and above DeepSeek’s own 3.50. The merged artifact has acquired the linguistic surface features of the reasoning-distilled donor without inheriting the base accuracy gap that would have appeared if the merge had been destructive.

**Caveat on the DeepSeek-R1-Distill comparison.** DeepSeek-R1-Distill scored only 10% in our setup, with mean generation length 198 tokens against a 200-token cap. The R1 distillation family typically produces reasoning chains in the 500 to 2000 token range; our token cap truncates these chains before they conclude. The 10% figure therefore should not be read as evidence about DeepSeek-R1-Distill’s underlying mathematical capability, only about its behavior under a 200-token decoding budget. A longer-budget rerun is reserved for a follow-up paper.

## Discussion

### Contrast with activation-level transfer

A companion paper from the same author reports a negative result on activation-granularity cross-checkpoint transfer between the identical donor-host pair: a contrastive function vector extracted from the donor’s layer 12 last-token activation, injected into the host’s residual stream at the same layer with scalar , produces monotonically decreasing accuracy on GSM8K, with no value of improving over baseline.

The two results together suggest the following preliminary observation, which we offer not as a theorem but as a working hypothesis subject to broader empirical test: *Weight-space spherical interpolation between same-family checkpoints produces a model with sensible surface behavior, while activation-level last-token contrastive injection does not transfer the donor’s task-specific capability.* The mechanisms differ in scope and depth. SLERP modifies the full forward computation by interpolating every weight tensor in both source models along the great-circle path between their respective parameter values, producing a third model whose computational graph is fully shared with both ancestors but whose weight values differ uniformly. Activation injection, by contrast, adds a constant offset to one

position in the residual stream of the unchanged receiver, modifying neither the host’s weights nor its computational graph but only the value flowing through one tap point during one forward pass. The former operation produces coherence because every transformer block downstream of the change sees consistently perturbed input drawn from the interpolated parameter distribution. The latter operation cannot produce coherence at the level required for multi-step reasoning because subsequent transformer blocks expect to read from the host’s natural residual stream distribution, not from a constant-offset version of it, and the donor’s information content is not delivered through the natural inference path that the host’s downstream blocks were trained to consume.

This contrast suggests that cross-checkpoint capability transfer at the activation level is harder than the corresponding weight-level operation, even in the easiest possible setting (matching architecture, same tokenizer, donor is a fine-tune of the host’s pretraining checkpoint).

## Limitations

We document a single recipe (SLERP ) on a single donor-host pair. We do not compare against TIES, DARE, or frankenmerge variants in this paper; those are reserved for a follow-up batch comparison. We do not yet report quantitative benchmark accuracy beyond the qualitative observation that outputs are non-pathological; benchmark results will be appended when complete.

The merged model has not been evaluated for instruction-following preservation, safety alignment retention, or task-specific degradation against either base. These evaluations are deferred to a downstream paper that will compare the SLERP variant against multiple merge recipes on a battery of benchmarks.

## Conclusion

A reproducible same-family SLERP learned tensor merge of Llama-3.1-8B-Instruct and DeepSeek-R1-Distill-Llama-8B produces a usable 8B model in three minutes of CPU compute. The artifact is small enough to recipient model locally.

It ships with standard transformers tooling.

Paired with a companion negative result on activation–level transfer between the same checkpoints, this finding suggests weight–level merging is a substantially more reliable composition technique than naive activation–level injection in the easiest tested configuration. Further recipes (DARE, TIES, frankenmerge depth–stack) will be reported in a downstream batch comparison.

## Code and Data Availability

Merge recipe, validation prompts, generation outputs, and reproduction script are released at <https://github.com/norika1207-lab/frankenstein-skeleton> (forthcoming). The merged model artifact itself is held private at the author’s discretion pending downstream evaluation.

9 C. Goddard et al., *Arcee’s MergeKit: A Toolkit for Merging Large Language Models*, arXiv:2403.13257, 2024.  
<https://arxiv.org/abs/2403.13257>

L. Yu et al., *Language Models are Super Mario: Absorbing Abilities from Homologous Models as a Free Lunch*, arXiv:2311.03099, 2024.  
<https://arxiv.org/abs/2311.03099>

P. Yadav et al., *TIES–Merging: Resolving Interference When Merging Models*, arXiv:2306.01708, 2023. <https://arxiv.org/abs/2306.01708>

DeepSeek–AI, *DeepSeek–R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*, arXiv:2501.12948, 2025.  
<https://arxiv.org/abs/2501.12948>

H. Y. Chen (Norika Oda), *Naive Contrastive Function Vector Injection Does Not Transfer Reasoning Capability Within the Llama–3 Family*, Zenodo, 2026. <https://doi.org/10.5281/zenodo.20394207>