



POLITECNICO
MILANO 1863

**SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE**

EXECUTIVE SUMMARY OF THE THESIS

DMS: LoRA Multi-Task Merging via Per-Module Magnitude Reallocation

LAUREA MAGISTRALE IN COMPUTER SCIENCE AND ENGINEERING - INGEGNERIA INFORMATICA

Author: NICOLÒ BOSSI

Advisor: PROF. MANUEL ROVERI

Co-advisor: MATTEO GAMBELLA

Academic year: 2025-2026

1. Introduction

Large Language Models (LLMs) are now routinely specialized to individual tasks through Parameter-Efficient Fine-Tuning, and Low-Rank Adaptation (LoRA) has become its dominant form [4]. LoRA freezes the pre-trained backbone and learns a small low-rank update for each task, which makes single-task specialization cheap and has produced a large public supply of ready-made adapters. The cost reappears when many of these specialists must be served at once. Handling T tasks means either swapping T adapters at inference time or keeping T copies of the model resident, and the storage and memory overhead grows linearly with the number of tasks.

Model merging offers a way around this. It consolidates several independently trained specialists into a single set of weights with no joint optimization, which sidesteps the data, the compute, and the catastrophic forgetting of multi-task or continual training, and can in principle run training-free. The obstacle is task interference. Task Arithmetic [5] represents each specialist by a task vector, the difference between its fine-tuned and base weights, and adds a scaled sum of these vectors to the base; sparsification

methods refine that sum, with TIES-Merging [7] trimming low-magnitude entries and electing a dominant sign per parameter before averaging the agreeing components, and DARE [8] randomly dropping a large fraction of the entries and rescaling the survivors. All of them degrade sharply relative to the specialists they are built from, and the effect is worse on LoRA, whose updates carry a large and uneven variance in magnitude. A few high-norm updates overwrite the contributions of the others, and when two tasks demand opposite changes to the same weight, averaging their updates cancels the learned signal.

The strongest existing remedies decompose the updates with a per-tensor Singular Value Decomposition (SVD). Task Singular Vectors (TSV-Merge) [2] decomposes each task update, keeps only its dominant singular directions, and decorrelates the resulting task subspaces before recombining them, while KnOTS [6] concatenates the updates and aligns them in a shared singular-vector basis before resolving them. Both are accurate and both are computationally heavy, and because they are closed-form spectral transforms they carry no per-task objective, so nothing in their construction pro-

tests a minority task from being traded away for a higher mean. A complementary, LoRA-specific line of work reads the failure in the adapters themselves. DO-Merging [10] locates the root cause in magnitude: it decouples each update into a magnitude and a direction factor, averages the magnitudes, orthogonalizes the directions through a data-free gradient procedure, and rescales the merged update with a single global coefficient. RobustMerge [9] targets direction robustness instead, pruning low-magnitude adapter entries and applying a per-rank scaling derived from the gaps between the adapters’ singular values, renormalized across tasks. The global rescaling fails, because one scalar cannot serve tasks that pull in opposite directions: raising the magnitude helps the tasks that want a stronger update and harms those that want a weaker one. The location of the added magnitude therefore matters more than its total amount.

This work presents **DMS (Deconflicting Magnitude Steering)**, a data-light framework that resolves LoRA interference at the level of individual modules rather than with one global coefficient. DMS decides its allocation from a direct measurement of where the specialists conflict, avoids the cost of a per-tensor SVD, and optimizes the worst task alongside the average. On an eight-task suite with the Llama-3.2-1B backbone it reaches a mean accuracy of 65.75%, a statistical tie with the best SVD methods, obtained without a single decomposition. A conflict analysis then traces the interference to the architecture rather than to the semantics of the tasks.

2. The DMS Method

DMS treats each of the M linear modules of the backbone as an independent merging problem. For a module m , the T task updates $\Delta W_t^{(m)}$ are combined into one merged weight,

$$W_{\text{merged}}^{(m)} = W_0^{(m)} + \lambda_m \Phi\left(\{\Delta W_t^{(m)} : t \in M^{(m)}\}\right), \quad (1)$$

where $M^{(m)}$ is the subset of tasks that contribute at module m , Φ is a merge operator, and λ_m is a per-module scale. The method proceeds in three stages, summarized in Figure 1.

2.1. Profiling

The first stage measures, at each module, how much an unweighted average would cancel the magnitude of the specialists. The signal is a magnitude-cancellation ratio, computed on CPU from the adapter weights alone as a handful of Frobenius norms,

$$c_m = \frac{\|\frac{1}{T} \sum_{t=1}^T \Delta W_t^{(m)}\|_F}{\frac{1}{T} \sum_{t=1}^T \|\Delta W_t^{(m)}\|_F} \in (0, 1]. \quad (2)$$

A value near one means the task updates reinforce each other; a value near zero means they cancel. Alongside c_m , an inter-task activation-overlap score, computed from the specialists’ activations rather than their weights, records how similarly the specialists respond at each module, which separates the sites where tasks share features from those where they compete.

2.2. Search

The second stage searches over per-module merge policies with NSGA-II [1], a multi-objective evolutionary algorithm. Two objectives are optimized together: the eight-task mean accuracy and the worst-task accuracy. The genome encodes, per module, the participating tasks $M^{(m)}$ and the merge operator Φ , together with a single global scale $\lambda \in \{0.5, 1.0, 1.5\}$. The operator Φ decouples magnitude from direction, applying a magnitude rule (**sum**, **mean**, or **max**) and a direction rule (**mean** or **sum-renorm**) separately, which lets the search recover a low-norm task by choosing **max** where an average would suppress it.

2.3. Reallocation

The final stage is parameter-free. Reading the profile from Equation (2), it restores magnitude at exactly the modules where the average destroyed it,

$$\lambda_m = \frac{\lambda_{\text{base}}}{c_m}, \quad (3)$$

renormalized so that the mean of λ_m over all modules equals λ_{base} . Modules with heavy cancellation receive more scale, modules whose tasks already agree receive less, and the overall budget is preserved.

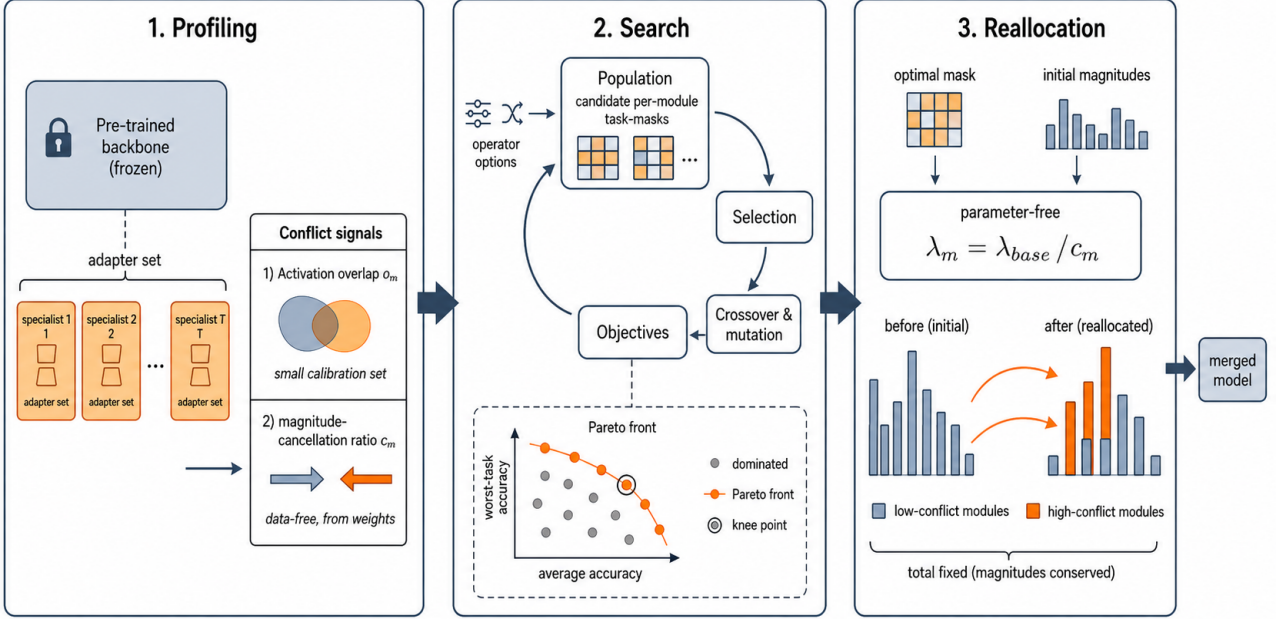


Figure 1: The DMS pipeline. Profiling extracts two per-module conflict signals (activation overlap o_m and the data-free magnitude-cancellation ratio c_m); an NSGA-II search selects the per-module masks and operators on the mean and worst-task Pareto front; and a parameter-free reallocation $\lambda_m = \lambda_{\text{base}}/c_m$ restores magnitude at the high-conflict modules while conserving the total budget.

3. Experimental Setup

The backbone is Llama-3.2-1B [3], a decoder of 16 layers with 112 linear modules in total. Each task specialist is a LoRA adapter of rank $r = 32$ and scaling $\alpha = 16$ with dropout 0.05, trained for 3000 optimizer steps and covering all 112 modules. All experiments run on a single NVIDIA L4 GPU with 24 GB of memory, which fixes the deployment budget the work targets. The suite has eight multiple-choice tasks from two families. Commonsense reasoning is covered by `piqa`, `winogrande`, `hellaswag`, and `siqa`; question answering by `boolq`, `openbookqa`, `arc_easy`, and `arc_challenge`. Every method is evaluated zero-shot on the full held-out test split of each task, for 22,419 test examples in total, and reported as an equally weighted eight-task average. The uncertainty band is the binomial standard deviation of that macro-average, ± 0.44 accuracy points, and is the same for every method because it depends only on the test-set sizes. The baselines are the strongest Model-Merging and LoRA-Merging methods available; their published configurations transfer poorly to this backbone, so each was retuned on a valida-

tion split for a fair comparison.

4. Results

Table 1 reports the eight-task average and the worst per-task accuracy for every method. DMS reaches 65.75%, which is +10.9 points over the un-adapted backbone (54.8%) and +2.1 points over a plain linear merge. It lands in a statistical tie with the two strongest SVD methods, TSV-Merge (65.64%) and KnOTS (65.53%), and matches the 65.01% ceiling set by keeping eight separate specialists. This parity is obtained with no per-tensor SVD: the merge is a closed-form pass whose magnitude signal costs a few Frobenius norms on CPU, so the merged model is as cheap to apply and deploy as a linear merge. The explicit worst-task objective pays off in the last column. DMS holds the lowest-scoring task at 47.9%, above every competitor, and the per-module operator alone recovers `boolq`, the lowest-norm task, by +16 points, by selecting the `max` magnitude rule where a global average would have drowned it out. This advantage sits within the noise band and is best read as a guarantee against collapse rather than a measured

Method	Avg.	Worst
Pre-trained backbone	54.80	34.5
Uniform Averaging	59.89	45.8
TIES w/DARE	62.48	45.1
TIES-Merging	62.77	45.7
RobustMerge	62.70	44.2
Task Arithmetic	63.63	46.1
TA w/DARE	63.66	46.2
DO-Merging	64.95	46.7
KnOTS	65.53	47.4
TSV-Merge	65.64	47.3
DMS (ours)	65.75	47.9

Table 1: Eight-task average accuracy and worst per-task accuracy. All averages carry a ± 0.44 binomial band. DMS is highest on both columns and ties TSV-Merge and KnOTS within noise.

win over the SVD methods.

The comparison of cost, summarized in Figure 2, is honest about where the effort falls. Every method here, DMS included, is configured by a search over a validation split, so none runs truly data-free at these accuracies. DMS changes the operator that this search configures, replacing a per-tensor decomposition with a closed-form CPU rule. The price is paid in the search itself, which explores a per-module policy space instead of a handful of global coefficients, a larger one-time offline cost that does not recur at merge or deployment time. On the validation front of the search, the reallocation of Equation (3) moves the average from 64.94% to 66.38%, confirming that steering magnitude by c_m pushes accuracy in the right direction.

A conflict analysis explains why the LoRA merges fail. The interference is architectural rather than semantic. The magnitude imbalance that DMS corrects concentrates in the deep feed-forward projections, where the total update magnitude varies by more than a factor of two across tasks (`arc_challenge` near 54 against `boolq` near 24). The little directional conflict that remains sits at the opposite end of the network, in the early attention layers. Direction and magnitude thus occupy complementary sites, and on these adapters magnitude is the operative axis, read from weight and activation statistics rather than from any measure

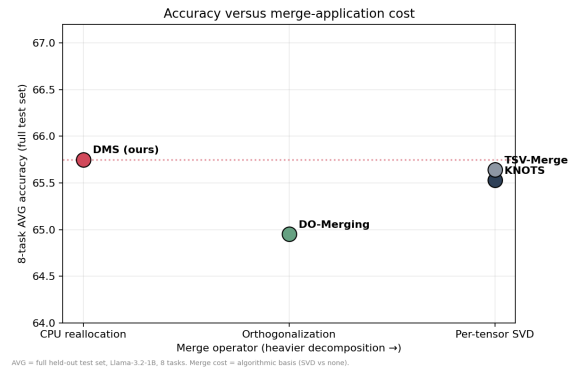


Figure 2: Accuracy versus merge-application cost. DMS matches the SVD methods without a per-tensor decomposition, at the cost of a heavier one-time offline search.

of task diversity. Acting on it per module lets a decomposition-free merge keep pace with the state of the art.

5. Conclusions

This work set out to make the merging of separately trained LoRA specialists reliable under a tight memory budget. The core difficulty is that a single global merge scale cannot serve tasks whose updates pull in opposite directions, so magnitude has to be controlled at the level of individual modules. DMS does this in three stages: an analytical profile of where the specialists cancel, an evolutionary search over per-module policies that optimizes the mean and the worst task together, and a parameter-free reallocation $\lambda_m = \lambda_{\text{base}}/c_m$ that restores magnitude where the average had removed it. The result is parity with the leading SVD methods, at 65.75%, reached without any per-tensor decomposition, together with an analysis showing that on LoRA adapters the interference lives in magnitude in the deep feed-forward layers rather than in the directional conflict that spectral methods are built to resolve.

The study has clear limits. All experiments use a single 1B-parameter backbone and a single training seed, so the reported band reflects test-set sampling rather than run-to-run variance, and it is untested whether the same conflict structure persists at larger scales or across many more tasks. The worst-task result is a guarantee against collapse and not a statistically measured advantage over the SVD methods. The per-module search is a larger offline

cost than tuning a few global coefficients, and `openbookqa` shows a small regression when the magnitude budget is spent elsewhere.

Two directions extend the framework. The first is scale: all results use a single 1B backbone, and re-profiling DMS on 3B–8B models and on larger, more heterogeneous task collections would establish whether the magnitude-driven account of interference is a property of small models or a general one. The second is composition. Because DMS controls only the per-module magnitude and leaves the directional combination to a simple operator, it composes naturally with methods that act on the other axis: the direction operator could be combined with the spectral cleaning of TSV-Merge [2] or KnOTS [6], while the sparsification of TIES-Merging [7] or DARE [8] could pre-trim the task vectors before the cancellation ratio is profiled. Because the reallocation is parameter-free and reads only the cached adapters, it can be layered on top of these methods at negligible cost.

References

- [1] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.
- [2] Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodolà. Task singular vectors: Reducing task interference in model merging. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18695–18705, 2025.
- [3] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- [4] Edward J. Hu, Phillip Wallis Yelong Shen, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021.
- [5] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.
- [6] 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*, 2025.
- [7] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, 2023.
- [8] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. *arXiv preprint arXiv:2311.03099*, 2024. DARE.
- [9] Fanhu Zeng, Haiyang Guo, Fei Zhu, Li Shen, and Hao Tang. Robust-merge: Parameter-efficient model merging for mllms with direction robustness. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.
- [10] Shenghe Zheng, Hongzhi Wang, Chenyu Huang, Xiaohui Wang, Tao Chen, Jiayuan Fan, Shuyue Hu, and Peng Ye. Decouple and orthogonalize: A data-free framework for LoRA merging. *arXiv preprint arXiv:2505.15875*, 2025.