LLM Merging Competition Technical Report for NeurIPS 2024: Efficiently Building Large Language Models through Merging

Anonymous Author(s)
Affiliation
Address
email

Abstract

We present our solution for the LLM Merging Competition: Building LLMs Efficiently through Merging at NeurIPS 2024. We experimented with a range of base models and merging strategies, ultimately choosing *Llama3-8B-Instruct* and its variants as our foundation model, merged using the DARE-TIES strategy. To further improve inference-time performance, we incorporated few-shot enhancement and chain-of-thought prompting techniques. We secured 1st place on the released public dataset with a score of 0.83, and achieved a score of 0.41 in the Finals.

1 Introduction

Large Language Models (LLMs) have demonstrated significant success across a wide range of Natural Language Processing (NLP) downstream tasks [1, 2, 3, 4, 5], such as mathematical reasoning [6, 7, 8],instruction following[9, 10], code generation[11, 12] and multilingual processing[13, 14]. However, adapting LLMs to new tasks or expanding their multi-task capabilities, whether through instruction tuning or pretraining from scratch, imposes significant computational demands. To address these challenges, model merging [15] has emerged as a practical and efficient approach to enhance the multi-task performance of LLMs in resource-constrained or training-free scenarios. Considerable efforts have been devoted to developing techniques that seamlessly integrate fine-tuned models into a cohesive multitask merged model, effectively addressing issues like parameter alignment, weight interference, and task-specific optimization without incurring heavy computational overhead.

The LLM Merging Challenge emphasizes the significance of exploring model merging as a strategy for developing unified, adaptable multitask models that can operate efficiently and effectively under limited resource conditions. In this competition, we experimented with a variety of base models released prior to May 1, 2024, including *Mistral-7B-Instruct-v2, Llama3-8B-Instruct, Flan-T5-large, Gemma-7B-Instruct, and WizardLM-2-7B*. We also explored several merging strategies, such as Task Arithmetic [16], TIES-Merging [17], DARE [18], and Consensus [19]. After careful comparison, we selected Llama3-8B-Instruct and its variants as our foundation model, merging them using the DARE-TIES strategy. The merged model inherited the strengths of its sub-models and demonstrated stronger zero-shot capabilities. We further enhanced the merged model by incorporating Chain-of-Thought [20] and Few-Shot learning [21] techniques. The results demonstrate that the merged model retains and also benefits from in-context learning [22] capabilities. In terms of results, we secured 1st place on the public dataset with a score of 0.83 and achieved a score of 0.41 in the Finals.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

2 Method

We conducted experiments on multiple model merging methods to determine the most effective approach for combining selected models. We implemented and compared the following methods:Task Arithmetic[16], TIES-Merging[17] , DARE[18] and Consensus[19]. Below is a brief overview of each method.

Task Arithmetic creates a “task vector” for each fine-tuned model by subtracting a common base model, merging these task vectors linearly, and then adding them back to the base. This method retains the unique features of each model, especially when they share a common foundation, but may be limited in mitigating parameter interference.

TIES-Merging (Trim, Elect Sign & Merge) approach enhances Task Arithmetic method by applying magnitude sparsification to task vectors, then employs a sign consensus algorithm to reduce both interference of redundant parameter values and disagreement on the sign of a given parameter’s values across models.

DARE (Drop and Rescale) also reduces interference by sparsifying task vectors, but it differs with TIES by using random pruning with a rescaling technique. DARE can optionally incorporate the TIES sign consensus algorithm (dare_ties) or be applied linearly (dare_linear). This method has shown a strong capacity to maintain the strengths of the original models, even in complex merge scenarios.

Consensus method identifies task-specific paramaters in merged models and then removes "selfish" weights, which benefit only one task and interfere with others, and “catastrophic” weights, which are irrelevant to all tasks and degrade performance. By constructing “task masks” that identify which weights are important across multiple tasks, Consensus Merging ensures that only shared, beneficial parameters are retained. Like DARE, Consensus is also a plug-and-play module that can be applied to other merging method like Task Arithmetic(consensus_ta) and TIES-Merging(consensus_ties).

Another concurrent paper, EMR-Merging [23], proposes a similar concept but relies on separate masks for each downstream task instead of generating a single, unified model. Since this approach might conflict with competition rules, we chose not to adopt it.

Our experiments demonstrated that DARE consistently outperformed other methods, retaining a higher degree of each model’s performance while reducing interference. Specifically, the dare_ties variant yielded the best results, combining the benefits of TIES’s sparsification and sign consensus algorithm with DARE’s adaptive pruning.

Based on these findings, we selected DARE as the final model merging method for this competition.

Algorithm 1 Model Merging Evaluation Process

Require: Pre-trained model θ_{PRE}, fine-tuned models $\{\theta^i_{\text{SFT}}\}_{i=1}^N$, hyperparameters, test dataset `test.csv`

Ensure: Merged model predictions `submission.csv`

Use the DARE-TIES to merge models: $\theta_{\text{MERGED}} = \text{dare\_ties}(\theta_{\text{PRE}}, \{\theta^i_{\text{SFT}}\}_{i=1}^N, \text{hyperparameters})$
for each multiple-choice task in `test.csv` **do**
    Compute the token-length normalized log probabilities[a] across options using θ_{MERGED}
    Select the option with the highest probability
    Apply self-consistency and chain-of-thought (CoT) strategies
end for
for each generative task in `test.csv` **do**
    Generate response directly using θ_{MERGED}
end for
Consolidate all generated responses into `submission.csv`

[a] `https://blog.eleuther.ai/multiple-choice-normalization/`

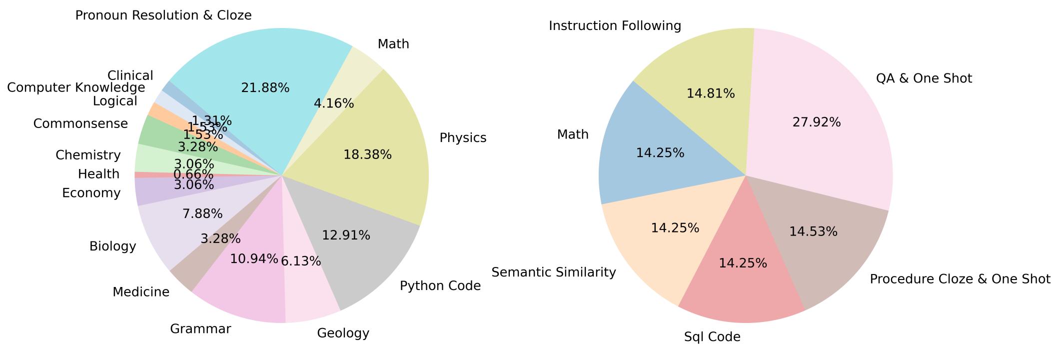

Figure 1: Statistical distribution of questions in the provided benchmark: (Left) distribution of multiple-choice questions, and (Right) distribution of generation-based questions.

3 Experiments

Benchmark statistics. The test set provided for benchmarking the performance of merged models consists of 807 questions, with 457 multiple-choice and 350 generation-based questions.

Figure 1 shows the statistical distribution of these questions across the two main categories. The multiple-choice questions span various domains, with physics knowledge (84 questions, 18.38%) and pronoun resolution & cloze tasks (100 questions, 21.88%) being the most prevalent, followed by coding-related, grammar-related, and other types of questions. The generation-based questions encompass a range of tasks, including question-answering & one-shot reasoning (98 questions, 27.92%), semantic similarity detection, and SQL generation. We conducted data analysis to extract publicly available datasets from MMLU[2], IFeval[5], RecipeNLG[24], TriviaQA[25], MedQA[26], and others. These datasets represent a comprehensive evaluation of multidisciplinary knowledge, instruction following, semantic understanding, code comprehension, and math reasoning.

Experiment setup. For multiple-choice questions, we calculate the probability of generating each option and select the option with the highest probability as the answer. For generation-based questions, we decode the answer based on the instruction and calculate the ROUGE-L score between the generated answer and the human-written ground truth. Decoding is performed with a maximum length of 1024 tokens and bf16 precision, with specific stop tokens set to eliminate irrelevant outputs. Multiple-choice and generation-based questions use separate chat templates for inference.

After thorough comparison, we select DARE [18] as our merging strategy, utilizing SGLang [27] for efficiency. To ensure consistent results during reasoning, the temperature for all LLMs was set to *zero*. All experiments were conducted on two RTX 4090 GPUs with a fixed seed.

For multiple-choice questions, performance is evaluated using accuracy, while for generation-based questions, ROUGE-L is employed as the metric. As ground truth answers are not available, we initially generate responses using GPT-4 in a zero-shot setting to establish an offline evaluation reference. These responses are subsequently reviewed and refined manually to create a high-quality answer set with minimal discrepancies, serving as a reliable benchmark for offline evaluation and optimization of the merging algorithm.

Baselines. As discussed in existing literature [28], a stronger base model tends to yield a more capable merged model. We first evaluate the performance of several training-free LLMs on both multiple-choice and generation-based tasks. The individual base models include *Mistral-7B-Instruct-v2*, *Llama3-8B-Instruct*, *Flan-T5-large*, *Gemma-7B-Instruct*, and *WizardLM-2-7B*, all evaluated in a zero-shot setting to identify the most suitable candidates for merging.

We further explore the potential of merged models to enhance performance through SOTA model merging strategies. Specifically, we merge *MaziyarPanahi/Llama-3-8B-Instruct-v0.8*[1] and *meta-llama/Meta-Llama-3-8B-Instruct*[2], experimenting with different merging strategies such as Task

[1] `https://huggingface.co/MaziyarPanahi/Llama-3-8B-Instruct-v0.8`

[2] `https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct`

Models or Methods	Multiple-choice					Generation-based	Online Bench
	Physics	Pronoun Res	Coding	Grammar	Overall	Rouge-L	Online Score
Unmerged base model							
Mistral-7B-Instruct-v0.2	50.0	71.0	81.4	92.0	58.9	38.0	43.0
Llama3-8B-Instruct	60.7	73.0	84.8	90.0	**62.8**	**46.6**	**53.0**
WizardLM-2-7B	39.3	60.0	74.6	78.0	52.1	42.5	-
Flan-T5-large	19.1	79.0	11.9	82.0	41.8	26.9	36.0
Gemma-7B-Instruct	47.6	55.0	81.4	92.0	52.5	16.4	-
Merged model							
Task Arithmetic [16]	59.2	71.0	76.8	85.0	61.7	43.0	49.0
TIES-Merging [17]	66.7	58.0	84.9	96.0	65.6	42.5	58.0
Consensus [18]	55.9	71.0	85.7	100	64.6	39.0	57.0
DARE-TIES [19]	61.9	74.0	86.44	94.0	68.2	43.2	60.0
+CoT	61.9	63.0	98.3	96.0	72.4	**45.0**	**65.0**
+Few-Shot	65.5	71.0	100.0	94.0	**74.2**	36.8	**65.0**

Table 1: Performance of base models (zero-shot) and merged models on key multiple-choice and generation-based tasks using different merging strategies, including Task Arithmetic, TIES-Merging, Consensus, and DARE, with CoT and Few-Shot enhancements for DARE.

Arithmetic, TIES-Merging, Consensus-Ties, and DARE-Ties. In our configuration, we set the *density* and *weight*[3] of *meta-llama/Meta-Llama-3-8B-Instruct* to 0.6 and 0.5, respectively, while configuring *MaziyarPanahi/Llama-3-8B-Instruct-v0.8* with a *density* of 0.55 and a *weight* of 0.5. Furthermore, we investigate whether the merged model could retain and leverage the In-context Learning [22] capabilities by integrating DARE merging with Chain-of-Thought [20] and Few-Shot [21] enhancements.

We also apply LoRA [29] to Llama-3-8B-Instruct for task-specific (e.g., MMLU, Semantic Similarity Detection), parameter-efficient fine-tuning prior to merging. However, due to overfitting to specific tasks, the merged model exhibits a loss of generalization on other types of tasks, often resulting in repeated outputs.

Main Results. The overall results are reported in Table 1 using 1. We analyze from the following perspectives.

Selecting an Appropriate Base Model by Performance Variability. We evaluate encoder-decoder models like *T5* and decoder-only LLMs such as *Llama3-8B-Instruct* and *Mistral-7B-Instruct-v0.2* on both offline and online benchmarks. *Llama3-8B-Instruct* achieves the highest online score of 53.0, followed by *Mistral-7B-Instruct-v0.2*, leading us to select *Llama3-8B-Instruct* as the base model.

Merged LLMs Outperform the Training-free Base Models. Overall, the merged models deliver significant performance gains over the unmerged base models, with improvements of 1–7%. Notably, DARE-TIES performs best, reaching an online score of 60.0, followed by TIES-Merging. However, these gains primarily result from improvements in multiple-choice questions, while in this specific case, performance on generation-based questions declines compared to base model.

Merged LLMs also Retain and Benefit from In-context Learning Abilities. We evaluate the DARE-TIES merging strategy with CoT and Few-Shot enhancements, and results show that the merged model retains and also benefits from in-context learning capabilities. It achieves accuracies of 72.4% and 74.2% on multiple-choice questions, respectively. However, as few-shot examples are challenging to obtain in online evaluations, we retain only the CoT technique.

4 Conclusion

We examine various model merging strategies to enhance large language models across multiple-choice and generation-based tasks. Thanks to effective model merging techniques and in-context learning capabilities, DARE-TIES with Chain-of-Thought (CoT) achieves notable performance gains, particularly in multiple-choice accuracy. Experimental results highlight model merging as an efficient way to build adaptable, high-performance multitask LLMs in resource-limited environments.

[3] *Weight* refers to the relative weighting of the task vector, while *Density* represents the fraction of the task vector's weights retained after sparsification.

References

[1] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *CoRR*, abs/2406.01574, 2024.

[2] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[3] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021.

[4] Sumithra Bhakthavatsalam, Daniel Khashabi, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, and Peter Clark. Think you have solved direct-answer question answering? try arc-da, the direct-answer AI2 reasoning challenge. *CoRR*, abs/2102.03315, 2021.

[5] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *CoRR*, abs/2311.07911, 2023.

[6] Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *CoRR*, abs/2308.09583, 2023.

[7] Zhibin Gou, Zhihong Shao, Yeyun Gong, yelong shen, Yujiu Yang, Minlie Huang, Nan Duan, and Weizhu Chen. ToRA: A tool-integrated reasoning agent for mathematical problem solving. In *The Twelfth International Conference on Learning Representations*, 2024.

[8] Zhibin Gou, Zhihong Shao, Yeyun Gong, yelong shen, Yujiu Yang, Nan Duan, and Weizhu Chen. CRITIC: Large language models can self-correct with tool-interactive critiquing. In *The Twelfth International Conference on Learning Representations*, 2024.

[9] Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. KTO: model alignment as prospect theoretic optimization. *CoRR*, abs/2402.01306, 2024.

[10] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[11] Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. Wizardcoder: Empowering code large language models with evol-instruct. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

[12] Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton-Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. Code llama: Open foundation models for code. *CoRR*, abs/2308.12950, 2023.

[13] Ahmet Üstün, Viraat Aryabumi, Zheng Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. Aya model: An instruction finetuned open-access multilingual language model. In

Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 15894–15939. Association for Computational Linguistics, 2024.

[14] Linzheng Chai, Jian Yang, Tao Sun, Hongcheng Guo, Jiaheng Liu, Bing Wang, Xinnian Liang, Jiaqi Bai, Tongliang Li, Qiyao Peng, and Zhoujun Li. xcot: Cross-lingual instruction tuning for cross-lingual chain-of-thought reasoning. *CoRR*, abs/2401.07037, 2024.

[15] Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *CoRR*, abs/2408.07666, 2024.

[16] 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.

[17] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-Merging: Resolving Interference When Merging Models.

[18] 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.

[19] Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jimenez, François Fleuret, and Pascal Frossard. Localizing Task Information for Improved Model Merging and Compression.

[20] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[21] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[22] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

[23] Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. Emr-merging: Tuning-free high-performance model merging. *arXiv preprint arXiv:2405.17461*, 2024.

[24] Michal Bien, Michal Gilski, Martyna Maciejewska, Wojciech Taisner, Dawid Wisniewski, and Agnieszka Lawrynowicz. Recipenlg: A cooking recipes dataset for semi-structured text generation. In Brian Davis, Yvette Graham, John D. Kelleher, and Yaji Sripada, editors, *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pages 22–28. Association for Computational Linguistics, 2020.

[25] Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1601–1611. Association for Computational Linguistics, 2017.

[26] Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? A large-scale open domain question answering dataset from medical exams. *CoRR*, abs/2009.13081, 2020.

[27] Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. Sglang: Efficient execution of structured language model programs, 2024.

[28] Anonymous. What matters for model merging at scale? In *Submitted to The Thirteenth International Conference on Learning Representations*, 2024. under review.

[29] Edward J Hu, yelong shen, Phillip Wallis, 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*, 2022.