[Re] TIES-Merging: Resolving Interference When Merging Models

Anonymous authors
Paper under double-blind review

Abstract

This paper presents a detailed reproduction study of the TIES-MERGING model merging technique, as introduced by Yadav et al. (2023). Our objective is to replicate the primary findings of the original research, highlighting the efficacy of TIES-MERGING over baseline models across various scenarios, including different modalities and an increased number of tasks. Through our efforts, we aim to validate these findings, assess the optimal task quantity for peak single-task model performance, and evaluate the effectiveness of employing a top-k selection process while trimming. Utilizing the codebase provided by the original authors with necessary modifications, we incorporate additional scripts for data preparation and integrate code from Yu et al. (2023) for comparison against other merging algorithms. Despite encountering some challenges, such as missing components in the provided GitHub code and a lack of responsiveness from the original authors, our results largely corroborate the claims made by Yadav et al., showing a slight deviation in performance metrics. By conducting experiments under restricted settings, this paper demonstrates that TIES-MERGING operates effectively according to our reproduction efforts, affirming its potential in model merging applications.

1 Reproducibility Summary

Scope of Reproducibility - We examine the main result in Yadav et al. (2023). Through reproduction of the methods introduced in this paper, we verify the favorable performance of TIES-MERGING compared to the baseline models in different modalities, as well as its greater stability in terms of increased task number. We extend from the results in this original paper and validate the number of tasks best fit for the performance of single task models. Lastly, we assess the effectiveness of employing top-k selection.

Methodology - We leverage the existing code foundation with necessary modifications and add scripts for data preparation, integrating the code from Yu et al. (2023) for implementing other model merging algorithms. Minor adjustments are made due to library version discrepancies. Our code is available at *github repo*.

Results - We followed the methodology of the original paper and were able to achieve results that were, on average, about 1.5%p lower compared to the paper, with a standard deviation of 3.69%p, confirming that TIES-MERGING is coherent to the main claim.

What was easy - The original paper provided comprehensive details on the methodology, simplifying the reproduction process. Moreover, the author made much of the code available on GitHub and clearly listed the external libraries employed, which facilitated a smoother replication effort.

What was difficult - There were many missing parts in the GitHub code provided in the original paper. Among them, the T5 model did not come with the weight files, requiring us to fine-tune it ourselves. Additionally, merging methods such as Regmean and Fisher merging were not included in the code, so we had to source from other libraries or code these parts ourselves. There was no detailed explanation for some of the Hugging Face weight files used in the Bert experiments, making it impossible to know which files were used. Furthermore, among the models used in the experiments, the ViT required the implementation

of TIES MERGING codes in a different GitHub repository since it was not supported in the original code, and similar issues were encountered with Bert, making reproduction challenging.

Communication with original authors - We sent four emails to the authors, seeking clarification and further details on the methodologies and models employed in their experiments. Our inquiries were covering aspects that were either not fully explained in the publication or required additional insight to accurately replicate their work. Despite our efforts to establish a dialogue and better understand their research for accurate reproduction, we did not receive any responses from them.

2 Introduction

The emergence of pre-trained models (PTMs) has significantly revolutionized machine learning, offering flexible solutions for numerous tasks through the utilization of vast and diverse datasets (Zhuang et al., 2020). Building on the foundational work of pioneering researchers, the prevalent method involves fine-tuning these models for specific applications, thus improving performance with reduced dependence on extensive labeled data (Shnarch et al., 2022). However, despite their widespread success, the approach of fine-tuning individual models for distinct tasks encounters obstacles such as computational burdens and limited generalization capabilities.

In response to such impediment, recent research has increasingly explored model merging, an innovative technique that integrates two or more models into a single unified model (Li et al., 2023). This emerging and experimental approach provides a cost-effective way to develop new models, avoiding the over-exhaustion of graphics processing unit (GPU) resources. The application of model merging has shown significant impact on its effectiveness in the field.

In our target paper (Yadav et al., 2023),the authors introduce TIES-MERGING, a model-merging technique that implements TRIM, ELECT SIGN, and MERGE operations. This approach is specifically designed to mitigate the types of interference mentioned earlier before executing the merging process. Specifically, the merging method utilizes task vectors τ_t, which is $\theta_{t_{\text{ft}}} - \theta_{\text{init}}$, to encapsulate the directional adjustments required for each task t in the parameter space. The merging consists of three stages:

1. **Trimming** of the task vector τ_t to $\hat{\tau}_t$ by keeping the most significant k parameters.
2. **Electing** a dominant direction γ_m based on the aggregate movement of all tasks.
3. **Disjoint Merging** to average parameters θ_m that align with γ_m.

The final model parameters θ_m are given by $\theta_m = \theta_{init} + \lambda \cdot \tau_m$ where λ is a scaling factor. Notably, the TIES-MERGING technique has demonstrated superior performance over various established methods across a multitude of scenarios, encompassing different modalities, domains, number of tasks, model dimensions, architectures, and fine-tuning configurations. In our work, we reproduce the principal findings from the original work, and we expand upon this by conducting additional experiments to ascertain the practicality of the proposed methodology. Even with minor variations observed, our results substantiate the efficacy of TIES-MERGING introduced by the original authors.

3 Scope of Reproducibility

From this paper, we aim to verify the following claims:

Claim 1. TIES-MERGING demonstrates favorable performance compared to baseline model merging methods in different fine-tuning settings for several modalities including NLP and vision.

Claim 2. TIES-MERGING shows greater stability in overall performance as the number of merged tasks increases.

Claim 3. The optimized number of merged tasks varies among single task models, while generally depicting preference for smaller number of tasks.

Claim 4. Selection of top-k values during the trimming phase tends to outperform models resulted from random-k selection.

Claim 1 establishes the performance of TIES-MERGING through comparison with the baseline models from Yadav et al. (2023) in different T5 and ViT settings, with or without the validation set. For Claim 2, we specifically explore the drop in performance as the number of merged task increases in different merging models. Claim 3 is an extension from the original results of Yadav et al. (2023) and aims to discover whether there would be a specific number of tasks that would optimize the performance for different single-task models. Finally, we verify the efficacy of using the top-k selection for trimming task vectors in Claim 4.

These claims are the primary components among the results and analyses discussed in Yadav et al. (2023), for they **a.** indicate that the reproduced performance of the TIES-MERGING model on average ranks high in different settings and modalities, **b.** pinpoint essential components the current method they encompass, and **c.** substantiate the claim from the original paper regarding the increased significance of this model as the number of tasks grows.

The validation of the above hypotheses confirms the generalizability of TIES-MERGING between different modalities. Extending from Claim 1, we also identify the performance change in the model in terms of parameter size through performance evaluation on merging large language model (LLM) structures.

4 Methodology

We reproduce the results of Yadav et al. (2023) utilizing the source code made by the author. For merging techniques absent from the provided GitHub repository, we incorporate implementations from Yu et al. (2023). The experimental setup, encompassing both datasets and hyperparameters, is carefully arranged to reflect the specifications described in the initial study.

4.1 Model merging algorithms

We include all the merging techniques described in Yadav et al. (2023) and conduct experiments in two distinct scenarios: one with a validation set and one without. In the scenario without a validation set, the techniques include simple averaging (Wortsman et al., 2022), Task Arithmetic (Ilharco et al., 2022), and TIES-MERGING, with lambda values set at 0.4 for Task Arithmetic and 1.0 for TIES-MERGING, as specified in the original paper. In scenarios where a validation set is used, we include Fisher Merging (Matena & Raffel, 2022), RegMean (Jin et al., 2022), Task Arithmetic, and TIES-MERGING. Notably, for Task Arithmetic and TIES-MERGING, the approaches involve selecting the optimal lambda values based on their performance in the validation set, before assessing their performance on the test set.

4.2 Model descriptions

We conduct our experiments using the T5-base, T5-large (Raffel et al., 2020), ViT-B/32, ViT-L/14 (Dosovitskiy et al., 2020), and IA3 model (Liu et al., 2022) as outlined in the target paper. The T5 models transform NLP tasks into a unified text-to-text format, leveraging the Transformer architecture for a variety of tasks. The Vision Transformer (ViT) applies Transformer principles to computer vision, analyzing images as sequences of patches for advanced image classification. The IA3 model employs specialized techniques for optimizing models like T0-3B (Sanh et al., 2021) for specific tasks, showcasing the diverse applications and evolution of Transformer-based models in both the NLP and computer vision domains.

4.3 Datasets

We utilize the dataset as outlined in the original paper. Furthermore, when labels were absent in the test set, we repurposed a portion of the validation set as the test set. Specifically, for the IA3 model, we employ a total of 11 datasets. These include datasets from sentence completion (COPA (Roemmele et al., 2011), H-SWAG (Zellers et al., 2019), and Story Cloze (Sharma et al., 2018)), natural language inference (ANLI-1,2,3 (Nie et al., 2019), CB (De Marneffe et al., 2019), and RTE (Dagan et al., 2005)), coreference resolution (WSC

Method	Val	T5-base	T5-large	IA3	ViT-B/32	ViT-L/14
Zeroshot	-	47.4(-6.1)	51.4(-0.3)	53.1(-2.2)	-	-
Fine-Tuned		77.6(-5.2)	87.1(-1.7)	71.4(0.0)	91.7(1.2)	95.0(0.8)
Averaging	x	62.4(-3.5)	58.0(-1.6)	57.9(-0.1)	65.9(0.1)	79.7(0.1)
Task Arithmetic		68.3(-5.6)	62.2(-11.3)	59.1(-0.1)	60.6(0.2)	83.7(0.4)
TIES-MERGING		70.9(1.2)	66.0(-8.4)	64.8(-0.1)	72.8(0.4)	86.5(0.5)
Fisher	o	62.8(-6.1)	58.6(-6.0)	60.0(-2.2)	70.0(1.7)	78.58(-3.6)
RegMean		70.6(-0.6)	73.1(-0.1)	57.9(-0.1)	80.1(8.3)	88.5(4.8)
Task Arithmetic		68.3(-4.9)	64.1(-9.2)	64.0(0.1)	69.0(-1.1)	84.8(0.3)
TIES-MERGING		72.7(-1.2)	69.1(-7.8)	66.4(0.0)	73.7(0.1)	86.5(0.5)

Table 1: **[RE] Evaluating Various Model Merging Techniques** in a range of fine-tuning contexts and across different modalities (NLP and Vision), taking into account scenarios with and without the availability of a validation set. Here, Val denotes the utilization of a validation set. The numbers in parentheses indicate the difference from the results of the original paper.

(Levesque et al., 2012) and Winogrande (Sakaguchi et al., 2021)), and word sense disambiguation (WiC (Pilehvar & Camacho-Collados, 2018)). For the T5 models, a total of 7 datasets are used, incorporating Story Cloze, Winogrande, and WSC, along with additional datasets from paraphrase identification (PAWS (Zhang et al., 2019)), and question answering (QASC (Khot et al., 2020), WikiQA (Yang et al., 2015), and QuaRTz (Tafjord et al., 2019)). In the case of the ViT models, a total of 8 image classification datasets are included, featuring datasets from Cars (Krause et al., 2013),GTSRB (Stallkamp et al., 2011)), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), MNIST (LeCun et al., 2010), RESISC45 (Cheng et al., 2017), and SUN397 (Xiao et al., 2016).

4.4 Experimental setup

We conducted our experiments using A6000 and A100-SXM GPUs, and during the experimental process, for datasets without labels in the test set, we followed the Original paper's code and used the remainder of the validation set as the test set, excluding 32 samples.

5 Results

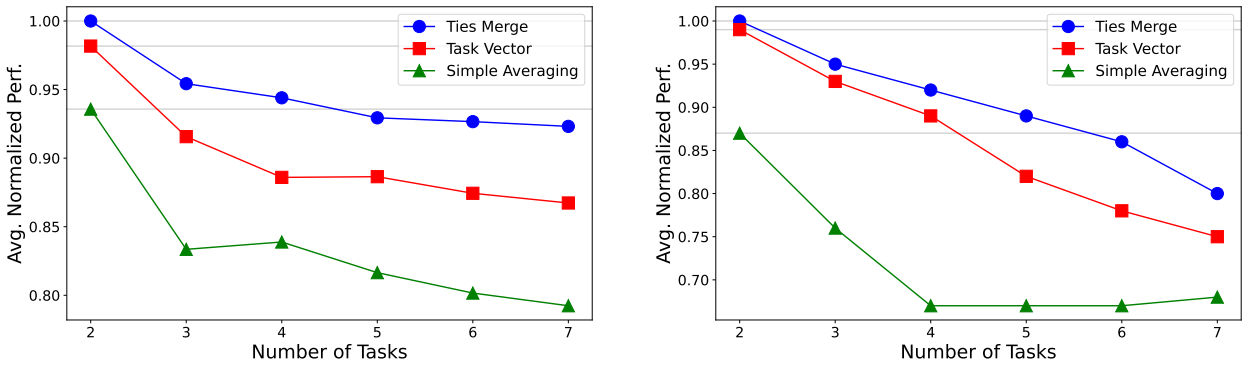
5.1 Results reproducing original paper

Result 1 - We replicate the primary experiment to evaluate how well TIES-MERGING, as described in Yadav et al. (2023), performs across different types, such as text and vision, and various fine-tuning methods. We conduct the experiments using the same settings as those presented in Table 1 of the original paper. For these experiments, we assess the performance of the model merging algorithms, as described in Section 4.1, employing T5-base, T5-large, ViT-B/32, ViT-L/14, and IA3. The datasets utilized in these experiments are derived from those mentioned in Section 4.3. Furthermore, in alignment with the methodology outlined in Yadav et al. (2023), experiments are performed for each method to evaluate their performance in scenarios both with and without the availability of a validation set.

In our reproduced experiments, TIES-MERGING consistently shows superior performance or generally favorable outcomes compared to other merging strategies across various evaluation settings. In comparing our experimental outcomes with those reported in Yadav et al. (2023). we observe an overall average difference of approximately -1.59%. Furthermore, the median difference is -0.1%, accompanied by a standard deviation of 3.69%.

Result 2 - Our reproducibility examines the scalability of TIES-MERGING in handling an increasing number of tasks and its ability to generalize across diverse task sets. Because of the resource constraints, we conducted experiments on T5-base and T5-large models.

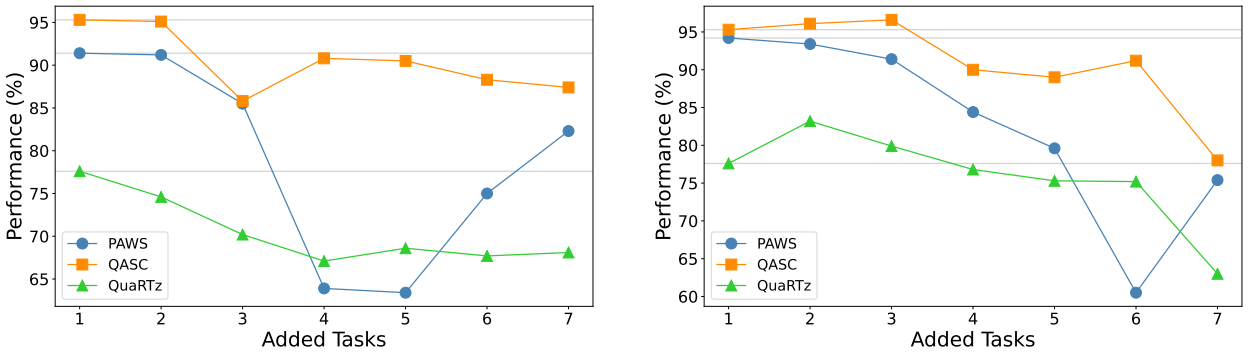
The results, as depicted in Figure 1, reveal a distinct superiority in performance of TIES-MERGING compared to other task integration techniques such as Task Vector and Simple Averaging when faced with a growing number of tasks. Figure 1(a) demonstrates the outcomes for the T5-base model, with TIES-MERGING maintaining relatively stable performance. In contrast, the Simple Averaging method suffers a more significant performance decline, highlighting its limitations in managing increasing task complexity. Figure 1(b) presents findings from the T5-large model, illustrating that TIES-MERGING surpasses alternative methods in performance as more tasks are integrated. Notably, the Simple Averaging approach lags significantly in performance. The empirical evidence, as argued in the original paper, suggests that TIES-MERGING is robust in the face of a growing number of tasks.



(a) T5-base (b) T5-large

Figure 1: **[RE] TIES-MERGING Scales Better**: This result measures average performance with the integration of varying numbers of tasks. The left figure presents the outcomes of experiments conducted with the T5-base model, whereas the right figure delineates the results from the T5-large model. As demonstrated in Yadav et al. (2023), it is observed that TIES-MERGING exhibits the best performance as the number of merged models increases.

5.2 Results beyond original paper



(a) T5-base (b) T5-large

Figure 2: **Comparison with single task models**: The presented findings denote a comparative analysis between single-task models of PAWS, QASC, and QuaRTz, and the corresponding merged models. The experimental outcomes for the T5-base model are depicted on the left, while the results for the T5-large model are illustrated on the right. Generally, there is an observable trend of performance degradation with a varying number of models engaged in the merging process.

Result 3 - In this section, we evaluate the performance disparity between models individually trained on specific tasks and those integrated via TIES-MERGING. Observations in section 5.1 reveal that TIES-

MERGING outperforms methodologies in model merging. Nonetheless, the extent of performance gap between merged models and those trained on each single task remains a pivotal consideration for their applicability in real-world scenarios. To this end, our analysis extends to a comparative assessment of merged models against single-task models, progressively increasing the number of tasks incorporated into the model merging.

Due to resource constraints, we conduct comparative analysis with single task models for PAWS, QASC, and QuaRTz, based on the T5-base and T5-large models. In each experiment for PAWS, QASC, and QuaRTz, the sequences for merging other models are as follows.

- PAWS, QASC, QuaRTz, WikiQA, Winogrande, Story Cloze, WSC
- QASC, PAWS, QuaRTz, WikiQA, Winogrande, Story Cloze, WSC
- QuaRTz, QASC, PAWS, WikiQA, Winogrande, Story Cloze, WSC

The experimental findings, as illustrated in Figure 2, show that there is a decrease in performance with the increase in the number of task models involved in the merging process. In the results of the T5-large model (Figure 2 (b)), we can observe that merging two models can achieve better performance for QASC and QuaRTz compared to single task models. This indicates the importance of carefully selecting which task models to integrate for each specific task. Furthermore, from the results of the T5-base and T5-large models, it can be observed that the performance of the merged models significantly declines after merging more than two or three models. These experimental results suggest that merging a smaller, appropriate number of models may be more advantageous for overall performance than attempting to cover multiple tasks.

Result 4 - In this analysis, we investigate the efficacy of extracting the top-k parameters based on their magnitude during the Trimming phase within the TIES-MERGING. The extraction of these top-k largest-magnitude parameters, the process advances to the Electing phase. Nonetheless, the influence of these selected top-k parameters on the overall approach lead us to raise questions about their actual efficacy. Therefore, to evaluate the significance of the top-k values, we executed the TIES-MERGING algorithm by randomly selecting k parameters during the trimming phase, enabling a comparative assessment of its impact. For the experimental setup, we refer to section B.2 of the original paper and conduct experiments using the T5-Base model, selecting k in the range of 10 to 100%.

From the Figure 3, The comparison between selecting the top-k and random-k parameters during the Trimming phase of TIES-MERGING reveals a noticeable trend. Generally, the top-k selection exhibits a superior performance over the random-k selection. This performance advantage is most apparent when k is smaller, suggesting that the top parameters by magnitude contribute more significantly to the model's effectiveness. As k increases, the performance gap narrows, indicating a diminishing return on the prioritization of parameters solely based on their magnitude.

6 Discussion

Our study highlights the effectiveness of the TIES-MERGING, initially introduced by Yadav et al. (2023). By replicating original experiments, we demonstrate its versatility and robustness not only in text and vision modalities but also with various fine-tuning strategies. Our further analysis reveals TIES-MERGING's exceptional scalability with T5-base and T5-large models, maintaining stable performance even as task complexity increases. This contrasts with other task integration techniques, underscoring its potential for integrating diverse task sets seamlessly.

We also explore the impact of merging strategies on performance, finding that while merging numerous tasks can reduce performance, selectively combining tasks can enhance it. This indicates the importance of strategic task combination and suggests a limit to the number of tasks that can be effectively merged. Additionally, we discover that selecting the top-k parameters significantly outperforms random selection, especially for smaller k values, highlighting the importance of key parameters in model effectiveness.

In conclusion, TIES-MERGING emerges as a potent model merging methodology for a range of tasks and settings, promising to advance research and application development in the evolving field of machine learning.

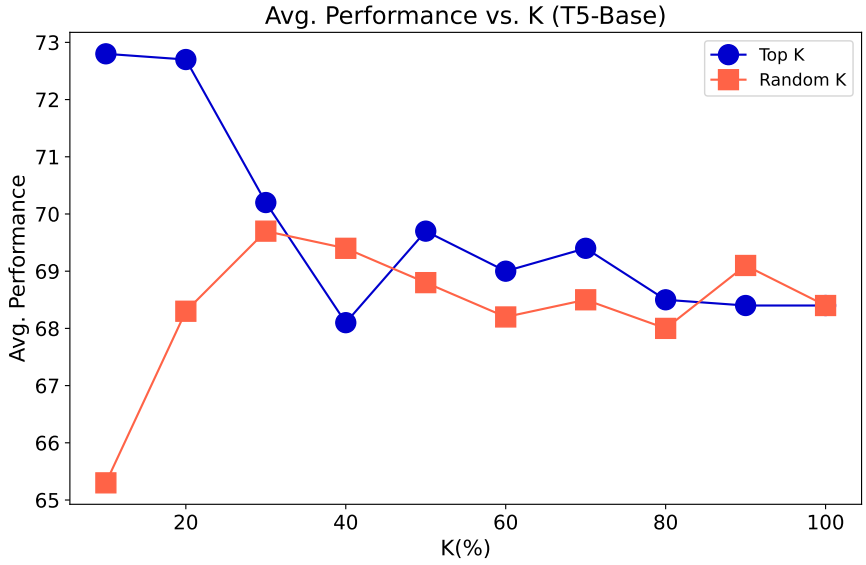


Figure 3: **Comparison between Top-k and Random-k**: The results when selecting values through either the Top-k or randomly choosing k values (Random-k) during the Trimming phase of TIES-MERGING. Generally, for the T5-base model, it is observed that Top-k tends to outperform Random-k in terms of performance.

References

Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3606–3613, 2014.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *Machine learning challenges workshop*, pp. 177–190. Springer, 2005.

Marie-Catherine De Marneffe, Mandy Simons, and Judith Tonhauser. The commitmentbank: Investigating projection in naturally occurring discourse. In *proceedings of Sinn und Bedeutung*, volume 23, pp. 107–124, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

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.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. In *The Eleventh International Conference on Learning Representations*, 2022.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. Qasc: A dataset for question answering via sentence composition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8082–8090, 2020.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 554–561, 2013.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

Hector Levesque, Ernest Davis, and Leora Morgenstern. The winograd schema challenge. In *Thirteenth international conference on the principles of knowledge representation and reasoning*, 2012.

Weishi Li, Yong Peng, Miao Zhang, Liang Ding, Han Hu, and Li Shen. Deep model fusion: A survey. *arXiv preprint arXiv:2309.15698*, 2023.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35:1950–1965, 2022.

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

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*, 2019.

Mohammad Taher Pilehvar and Jose Camacho-Collados. Wic: the word-in-context dataset for evaluating context-sensitive meaning representations. *arXiv preprint arXiv:1808.09121*, 2018.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*, 2011.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021.

Rishi Sharma, James Allen, Omid Bakhshandeh, and Nasrin Mostafazadeh. Tackling the story ending biases in the story cloze test. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 752–757, 2018.

Eyal Shnarch, Alon Halfon, Ariel Gera, Marina Danilevsky, Yannis Katsis, Leshem Choshen, Martin Santillan Cooper, Dina Epelboim, Zheng Zhang, Dakuo Wang, et al. Label sleuth: From unlabeled text to a classifier in a few hours. *arXiv preprint arXiv:2208.01483*, 2022.

Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: a multi-class classification competition. In *The 2011 international joint conference on neural networks*, pp. 1453–1460. IEEE, 2011.

Oyvind Tafjord, Matt Gardner, Kevin Lin, and Peter Clark. Quartz: An open-domain dataset of qualitative relationship questions. *arXiv preprint arXiv:1909.03553*, 2019.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, pp. 23965–23998. PMLR, 2022.

Jianxiong Xiao, Krista A Ehinger, James Hays, Antonio Torralba, and Aude Oliva. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision*, 119:3–22, 2016.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Yi Yang, Wen-tau Yih, and Christopher Meek. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 2013–2018, 2015.

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*, 2023.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

Yuan Zhang, Jason Baldridge, and Luheng He. Paws: Paraphrase adversaries from word scrambling. *arXiv preprint arXiv:1904.01130*, 2019.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020.

A More Results

Method	Val	Avg	PAWS	QASC	QUARTZ	STORY CLOZE	WIKI-QA	WINOGRANDE	WSC
Zeroshot	-	47.4	50.2	37.8	51.9	47.5	34.6	50.1	59.7
Fine-Tuned		77.6	91.4	95.3	77.6	81.3	95.2	50.7	51.4
Averaging	x	62.4	55.9	80.9	57.5	48.7	91.5	50.9	51.4
Task Arithmetic		68.3	60.6	85.6	63.5	71.7	93.6	50.9	52.1
TIES-MERGING		70.9	74.6	90.6	64.3	69.1	91.0	51.6	54.9
Fisher	o	62.8	57.6	91.7	57.7	56.0	73.7	50.9	52.1
RegMean		70.6	74.1	87.6	67.1	70.7	94.7	51.9	47.9
Task Arithmetic		68.3	60.6	85.6	63.5	71.7	93.6	50.9	52.1
TIES-MERGING		72.7	82.3	87.4	68.1	75.0	91.5	52.1	52.8

Table 2: Performance of the T5-base model tested on 8 NLP datasets

Method	Val	Avg	PAWS	QASC	QUARTZ	STORY CLOZE	WIKI-QA	WINOGRANDE	WSC
Zeroshot	-	51.4	49.6	21.5	51.1	54.0	70.8	49.2	63.9
Fine-Tuned		87.1	94.2	95.4	85.0	90.3	95.7	70.0	79.2
Averaging	x	58.0	54.3	59.1	67.6	53.8	60.8	50.5	59.7
Task Arithmetic		62.2	62.5	88.3	73.4	78.5	24.1	57.2	51.4
TIES-MERGING		66.0	66.7	87.4	73.9	79.2	46.5	60.2	47.9
Fisher	o	58.6	48.2	41.6	58.6	87.9	67.5	49.7	56.9
RegMean		73.1	87.0	62.1	80.3	80.3	84.8	55.3	61.8
Task Arithmetic		64.1	67.4	87.7	71.8	79.9	27.2	59.3	55.5
TIES-MERGING		69.1	75.4	78.0	63.0	84.5	60.2	64.2	58.3

Table 3: Performance of the T5-large model tested on 8 NLP datasets

Method	Val	Avg	RTE	CB	WINOGRANDE	WIC	WSC	COPA	H-SWAG	STORY CLOZE	ANLI-R1	ANLI-R2	ANLI-R3
Zeroshot	-	53.1	58.4	54.2	50.9	51.9	63.9	75.0	39.0	86.5	35.8	34.3	34.3
Fine-Tuned		71.42	82.4	95.8	75.4	71.6	66.0	85.3	44.4	95.0	70.3	46.5	53.0
Averaging	x	57.9	81.4	58.3	53.9	55.1	52.8	80.9	40.1	92.4	43.1	39.3	40.1
Task Arithmetic		59.1	76.5	79.2	57.6	52.0	49.3	67.6	31.4	81.3	60.1	47.4	47.9
TIES-MERGING		64.8	81.4	87.5	61.1	59.5	57.6	80.2	42.6	91.1	58.0	46.5	47.2
Fisher	o	60.0	81.0	75.0	53.7	52.3	62.5	80.9	40.1	92.9	43.7	38.9	39.4
RegMean		57.9	81.4	58.3	53.9	55.1	52.8	80.9	40.1	92.4	43.1	39.3	40.1
Task Arithmetic		64.0	74.1	83.3	62.9	49.0	49.3	88.2	41.5	95.3	60.8	49.3	50.2
TIES-MERGING		66.4	78.0	83.3	67.9	57.6	59.0	81.7	42.7	90.2	66.9	51.4	51.2

Table 4: Performance of the IA3 T0-3B model tested on 11 NLP datasets

Method	Val	Avg	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
Individual	-	91.65	75.29	77.68	96.11	99.74	97.46	98.73	99.69	88.48
Averaging	x	65.94	65.25	63.34	71.46	71.19	64.16	52.79	87.46	51.86
Task Arithmetic		60.62	36.71	41.04	53.79	64.70	80.62	66.05	98.12	43.91
TIES-MERGING		72.83	59.75	58.61	70.67	79.78	86.20	72.11	98.30	57.21
Fisher	o	70.01	65.06	68.86	77.16	70.41	72.48	60.31	97.48	48.32
RegMean		80.10	67.86	66.70	82.71	93.15	86.42	82.03	97.26	64.73
Task Arithmetic		69.01	55.26	54.94	66.68	76.11	80.21	69.65	97.34	51.89
TIES-MERGING		73.69	62.79	61.22	72.90	79.22	84.88	72.17	97.95	58.35

Table 5: Performance of the ViT-B/32 model tested on 8 image classification datasets

Method	Val	Avg	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD
Individual	-	95.03	82.32	92.39	97.37	99.89	98.11	99.24	99.69	91.20
Averaging	x	79.69	72.14	81.56	82.60	91.30	78.23	70.65	97.01	64.02
Task Arithmetic		83.70	72.51	79.21	84.54	90.11	89.24	86.52	99.10	68.35
TIES-MERGING		86.54	76.51	84.95	89.35	94.56	90.34	83.33	99.03	74.23
Fisher	o	78.53	70.10	82.13	79.56	98.85	82.59	56.44	99.15	59.41
RegMean		88.47	74.43	87.09	90.48	97.96	92.67	91.27	99.23	74.63
Task Arithmetic		84.80	74.13	82.13	86.65	92.48	87.91	86.78	98.94	69.41
TIES-MERGING		86.51	76.04	84.44	89.00	94.59	90.90	83.76	99.09	74.26

Table 6: Performance of the ViT-L/14 model tested on 8 image classification datasets