To See a World in a Spark of Neuron: Disentangling Multi-task Interference for Training-free Model Merging

Zitao Fang[1] **Guodong Du**[2,*] **Shuyang Yu**[3,†] **Yifei Guo**[4,†]
Yiwei Zhang[1] **Yiyao Cao**[1] **Jing Li**[5] **Ho-Kin Tang**[5] **Sim Kuan Goh**[1,*]
[1]Xiamen University Malaysia [2]The Hong Kong Polytechnic University
[3]Columbia University [4]Duke University [5]Harbin Institute of Technology (Shenzhen)
ait2209071@xmu.edu.my, duguodong7@gmail.com, simkuangoh@gmail.com

Abstract

Fine-tuning pre-trained models on targeted datasets enhances task-specific performance but often comes at the expense of generalization. Model merging techniques, which integrate multiple fine-tuned models into a single multi-task model through task arithmetic, offer a promising solution. However, task interference remains a fundamental challenge, leading to performance degradation and suboptimal merged models. Existing approaches largely overlooked the fundamental roles of neurons, their connectivity, and activation, resulting in a merging process and a merged model that does not consider how neurons relay and process information. In this work, we present the first study that relies on neuronal mechanisms for model merging. Specifically, we decomposed task-specific representations into two complementary neuronal subspaces that regulate input sensitivity and task adaptability. Leveraging this decomposition, we introduced **NeuroMerging**, a novel merging framework developed to mitigate task interference within neuronal subspaces, enabling training-free model fusion across diverse tasks. Through extensive experiments, we demonstrated that NeuroMerging achieved superior performance compared to existing methods on multi-task benchmarks across both natural language and vision domains. Our findings highlighted the importance of aligning neuronal mechanisms in model merging, offering new insights into mitigating task interference and improving knowledge fusion. Our project is available at `https://ZzzitaoFang.github.io/projects/NeuroMerging/`.

1 Introduction

"To see a world in a grain of sand..."

–William Blake

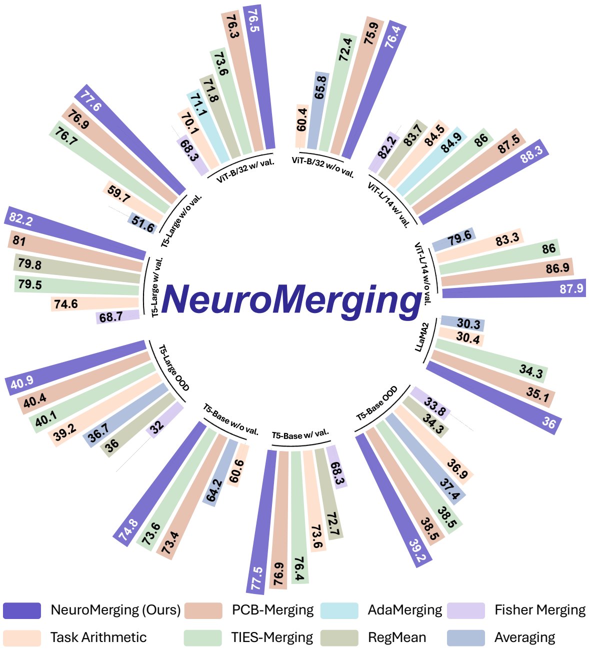

Figure 1: Comparative analysis of our method with baselines across models and domains.

To see a world in the spark of a neuron, the essence of deep learning lies in the complex dynamics of interconnected neurons, which has been empowering recent pre-trained models (PTMs), such as foundation models and large language models (LLMs) (Vaswani et al., 2017; Achiam et al., 2023; Touvron et al., 2023), to learn rich representations from large-scale datasets. These models demonstrate general capabilities while enabling effective fine-tuning for task-specific adaptation (Touvron et al., 2023). PTMs have driven significant advancements across core AI domains, including natural language processing (NLP), computer vision (CV), as well as applications in medicine, law, education (Bommasani et al., 2021; Moor et al., 2023; Ray, 2023). Building on this success, multi-task learning (MTL) has been a paradigm for integrating task-specific abilities into a model (Fifty et al., 2021), allowing generalization across multiple spe-

[*] Corresponding authors.
[†] Work conducted while at Xiamen University Malaysia.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 15720–15740
November 4-9, 2025 ©2025 Association for Computational Linguistics

Aspect	Task Vector	Neuronal Task Vector
Vector Space	Resides in the weight space	Resides in two complementary neuronal subspaces of neurons
Interpretation	Represents weight interpolation between pre-trained and fine-tuned models	Aligns with neuronal mechanisms, with input sensitivity and task adaptability
Activation	$y = \phi\left(\left(\mathbf{w}_0^k + \tau_t^k\right) \cdot \bar{\mathbf{x}}\right)$	$y = \phi\left(\mathbf{w}_0^k \cdot \bar{\mathbf{x}}_\parallel + \tau_{\parallel,t}^k \cdot \bar{\mathbf{x}}_\parallel + \tau_{\perp,t}^k \cdot \bar{\mathbf{x}}_\perp\right)$

Table 1: Differences between task vector and neuronal task vector. Please refer to Section 3 for details.

Method	Scale	Granularity Level
Fisher Merging[NeurIPS22]	Fisher Matrix	Parameter
RegMean[ICLR23]	Inner Product Matrix	Parameter
Task Arithmetic[ICLR23]	Uniformed	Task
Ties-Merging[NeurIPS23]	Uniformed	Parameter
DARE[ICML24]	$1/(1-p)$	Parameter
LoraHub[COLM24]	Evolver Searched	Task
AdaMerging[ICLR24]	Unsupervised Optimized	Layer
PCB-Merging[NeurIPS24]	Balancing Matrix	Parameter
NeuroMerging (Ours)	**L1-Norm**	**Neuron**

Table 2: Different merging scales and granularity levels.

cialized tasks. Nonetheless, MTL requires simultaneous training on all targeted datasets, which can be costly and pose privacy concerns. Model merging (Wortsman et al., 2022; Ilharco et al., 2023; Du et al., 2024) has recently emerged as an alternative paradigm to MTL for task adaptation, enabling the training-free integration of fine-tuned models, which are increasingly being shared publicly (e.g., on Hugging Face).

Model merging began with weight interpolation (Wortsman et al., 2022) to combine the strengths of different models in weight space by balancing competition and cooperation within shared representation (Du et al., 2024; Ilharco et al., 2023; Ortiz-Jimenez et al., 2023). In NLP, methods such as merging task-specific language models have been explored to build and update foundation models with multi-task capabilities (Wan et al., 2024a; Akiba et al., 2025; Wan et al., 2024b). Similarly, in CV, approaches like merging Vision Transformers (ViTs) trained on different tasks or domains have been investigated to create unified models capable of handling diverse visual tasks (Kim et al., 2021; Bao et al., 2022; Wang et al., 2024). In the multi-modal space, model merging has been applied to integrate models from different modalities, such as text and images, enhancing tasks like audio-visual question answering and image captioning (Sung et al., 2023; Sundar et al., 2024; Dziadzio et al., 2025). These advancements underscore model merging as a promising avenue for future research.

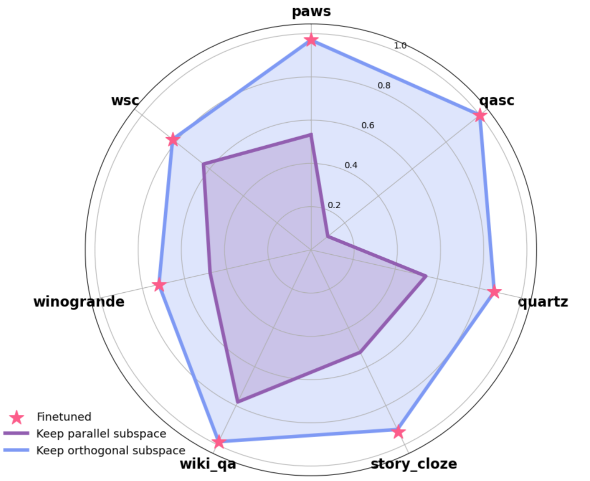

Figure 2: Impacts on neuronal subspaces decomposition of T5-Large. Retaining the orthogonal subspace while removing the parallel subspace preserves near-perfect performance across all tasks. In contrast, keeping the parallel subspace while removing the orthogonal subspace leads to a significant performance drop.

Existing methods for model merging primarily operate at three granularities: model-level, layer-level, or parameter-level (Ilharco et al., 2023; Yang et al., 2024b; Du et al., 2024), while overlooking the fundamental roles of neurons, their activation and connectivity (Suhaimi et al., 2022; Stelzer et al., 2021), which underpins the learning process all neural networks from Perception (Rosenblatt, 1958) to LLMs (Touvron et al., 2023). Table 2 provided a summary of existing methods across different granularities. In Figure 2 and Table 6 , we analyze on how changes in model weights along two complementary neuronal subspaces, leads to distinct impacts on task performance, by removing one and retaining the other. Notably, one subspace preserves most of the task-specific capabilities. This observation motivates us to explore model merging at the neuronal level, which could have important implications for mitigating task interference and could yield more robust merged models.

In this work, we present the first study to examine model merging at the neuronal level, illustrated in Figure 3. Specifically, we investigate the roles of neuronal mechanisms (i.e., neuron connectivity

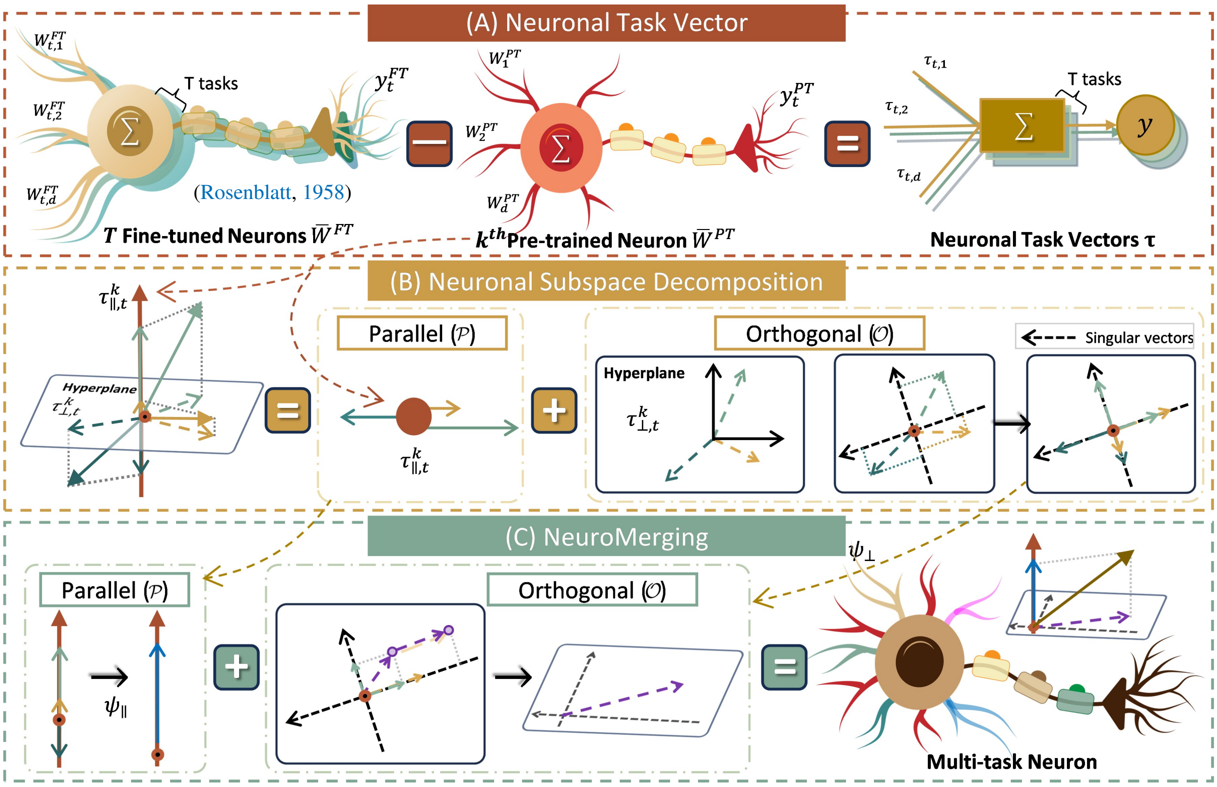

Figure 3: Illustration of our proposed framework. **(A)** Our approach explicitly considers neuronal activation mechanisms through *neuronal task vectors* $\tau^k_t \in R^d$, defined as the difference between fine-tuned and pre-trained neurons for task t. **(B)** These vectors are decomposed into parallel and orthogonal subspaces relative to pre-trained neurons, corresponding to input sensitivity and task adaptability. SVD constructs a coordinate system in the previously unstructured orthogonal subspace. **(C)** Our ***NeuroMerging*** method merges models efficiently within these low-dimensional complementary subspaces, rather than in the original high-dimensional weight space.

and activation) in model merging, both mathematically and empirically. We begin by decomposing task-specific representations from fine-tuned models into two complementary neuronal subspaces. These subspaces are mathematically characterized to show that they regulate neuron sensitivity and input adaptability. The key differences of our neuronal subspaces with existing task vectors summarized in Table 1. Leveraging the insights from the decomposition, we introduce **NeuroMerging**, a novel merging framework designed to mitigate task interference within neuronal subspaces, enabling training-free model fusion across diverse tasks. To evaluate our approach, we conduct experiments on multi-task benchmarks across both natural language and vision domains, considering various settings, including in-domain and out-of-domain generalization. Empirically, our method outperforms existing approaches as summarized in Figure 1. The main contributions of our paper are as follows:

- We presented the first exploration into the roles of neuronal mechanisms in the merging process, introducing a decomposition of task-specific representations into two complementary neuronal subspaces.
- Based on the insights from the neuronal subspaces, we proposed NeuroMerging, a new framework developed for model merging accounting input sensitivity and task specificity.
- We showed that NeuroMerging achieved superior performance compared to existing approaches on multi-task benchmarks in both natural language processing and vision domains.

2 Related Work

Multi-Task Learning (MTL) (Fifty et al., 2021) leverages transferable knowledge to handle multiple related tasks simultaneously. Existing MTL approaches primarily rely on architectural design or optimization strategies. Architectural-based methods, such as Mixture of Experts (MoE) (Shazeer et al., 2017), introduce specialized subnetworks that dynamically route inputs to task-specific experts, effectively reducing interference. However, these methods require modifying the pretrained model structure, increasing computational complexity, and limiting scalability (Liu et al., 2019; Shen et al., 2024; Lu et al., 2024). Optimization-based approaches, on the other hand, focus on balancing task gradients or loss functions to mitigate task conflicts during training (Bai et al., 2023; Cipolla et al., 2018). While these methods improve convergence, they still depend on task-specific training data, which may be impractical in real-world applications due to privacy concerns or data scarcity (Liang et al., 2022). In contrast, model merging offers an alternative paradigm by integrating knowledge from multiple fine-tuned models into a single unified model without requiring additional training data or architectural modifications (Wortsman et al., 2022; Ilharco et al., 2023). Notwithstanding the promising findings, a key challenge in model merging is task conflict (Yadav et al., 2023; Du et al., 2024), where different tasks compete for model capacity, potentially leading to suboptimal performance.

To resolve task conflicts, existing model merging methods can be categorized into three levels based on their granularity. Model-level merging combines entire model weights, typically through averaging or weighted aggregation, but often results in performance degradation due to the loss of task-specific knowledge, as seen in methods like Task Arithmetic (Ilharco et al., 2023) and LoRAHub (Huang et al., 2024a). Layer-level merging selectively integrates layers from different models under the assumption of shared representations; for instance, AdaMerging (Yang et al., 2024b) adapts layer selection to better preserve task-specific information. Parameter-level merging directly manipulates individual parameters to blend knowledge from multiple models, enhancing adaptability and robustness. Techniques such as Fisher Merging (Matena and Raffel, 2022), RegMean (Jin et al., 2023), TIES-Merging (Yadav et al., 2023), DARE (Yu et al., 2024a), and PCB-Merging (Du et al., 2024) exemplify this approach. However, existing methods largely overlook the fundamental role of neuron and neuron-level interactions in task specialization. In this work, we aim to bridge this gap.

Current neural network models, from the Perceptron invented by Rosenblatt (1958) to recent massive LLMs (Touvron et al., 2023), have grown significantly in scale and complexity. Nevertheless, the core principle remains unchanged: neurons and their connectivity still underpin the learning process (Stelzer et al., 2021; Suhaimi et al., 2022). During the pre-training and fine-tuning process, neurons are not merely passive components but active elements of the network, each contributing to learning and inference (Jiang et al., 2024; Islam et al., 2023). In this work, we attempt to conduct the first in-depth study on neuronal mechanisms in model merging.

3 Methodology

In this section, we first formalize the concept of neuronal task vector for model merging and then decompose neuronal task vectors into two complementary neuronal subspaces. Subsequently, we introduce our framework, **NeuroMerging**, shown in Algorithm 1 in Appendix A.4, which performs merging in the neuronal subspaces.

3.1 Preliminaries

Model merging considers the fusion of a set of T task specific models, $(\theta_1, \ldots, \theta_t, \ldots, \theta_T)$, fine-tuned from a pretrained model θ_0. With the task vector notation, each task is defined as $\tau_t = \theta_t - \theta_0$. The merged model is $\bar{\theta} = \theta_0 + \xi(\tau_1, ..., \tau_t, ..., \tau_T)$, where $\xi(\cdot)$ represents the transformation applied to each task vector τ_t and followed by merging.

Zooming into the neuronal level of τ_t from any neural network models (from perceptron to LLMs), the activation of the neuron can be computed with an input $\bar{x} \in \mathbb{R}^n$:

$$y = \phi(\mathbf{w}_t^k \cdot \bar{x}) \tag{1}$$

where $\mathbf{w}_t^k$ represents the synaptic connectivity to the k^{th} neuron after fine-tuning from the pre-trained weights $\mathbf{w}_0^k$. $\phi(\cdot)$ denotes non-linear activation function. Noting that although the input $\bar{x}$ is involved in the definition and derivation, it is not required during the merging process.

3.2 Neuronal Task Vector

Neuronal task vector is defined to be the difference between task-specific fine-tuning of a neuron with the pre-trained model: $\tau_t^k = \mathbf{w}_t^k - \mathbf{w}_0^k$. Based on the defined neuronal task vector, we can rewrite the activation of the k^{th} neuron as:

$$y = \phi((\mathbf{w}_0^k + \tau_t^k) \cdot \bar{x}) \tag{2}$$

3.3 Neuronal Subspace Decomposition

To examine how task-specific fine-tuning impact neurons, we decompose the neuronal task vectors into two complementary neuronal subspaces, visualized in Figure 3. Mathematically, this decomposition is formulated as:

$$\tau_t^k = \tau_{\parallel,t}^k + \tau_{\perp,t}^k, \tag{3}$$

where $\tau_{\parallel,t}^k = \mathbf{P}\tau_t^k$ projects the neuronal task vectors onto the pre-trained model's weight space, **Parallel Subspace** ($\mathcal{P}$). Here, $\mathbf{P}$ is the projection matrix onto the span of $\mathbf{W}_0^k$. $\tau_{\perp,t}^k = (\mathbf{I} - \mathbf{P})\tau_t^k$ captures the complementary orthogonal modifications, in the **Orthogonal Subspace** ($\mathcal{O}$). Noting that $\bar{x}$ is not required during decomposition.

Based on Neuronal Subspace Decomposition, we decompose both τ_t^k and $\bar{x}$ into two orthogonal subspaces ($\mathbf{P}$ and $\mathcal{O}$) of $\mathbf{w}_0^k$ and obtain:

$$y = \phi((\mathbf{w}_0^k + \tau_{\parallel,t}^k + \tau_{\perp,t}^k) \cdot (\bar{x}_{\parallel} + \bar{x}_{\perp})) \tag{4}$$

Due to the orthogonality between vectors in the parallel subspace and those in the orthogonal subspace, we can simplify the expression as:

$$y = \phi(\mathbf{w}_0^k \cdot \bar{x}_{\parallel} + \tau_{\parallel,t}^k \cdot \bar{x}_{\parallel} + \tau_{\perp,t}^k \cdot \bar{x}_{\perp}) \tag{5}$$

Since $\mathbf{w}_0^k$ and $\tau_{\parallel,t}^k$ are parallel, we can further express as:

$$y = \phi(s(\tau_{\parallel,t}^k) \cdot \bar{x}_{\parallel} + \tau_{\perp,t}^k \cdot \bar{x}_{\perp}) \tag{6}$$

Thus, we obtain a mechanistic view of the fine-tuned neuron: it adjusts its sensitivity s to input in the parallel subspace ($\mathcal{P}$) while adapting itself to capture $\bar{x}_{\perp}$ to handle task t in the orthogonal subspace ($\mathcal{O}$).

The role of each complementary subspace:

- **Parallel Subspace** ($\mathcal{P}$): this subspace captures transformations that preserve shared representations with the $\mathbf{w}_0^k$. It is also closely related to neuron sensitivity with larger magnitudes corresponding to higher sensitivity to changes in input activations.

- **Orthogonal Subspace** ($\mathcal{O}$): The orthogonal complementary subspace of $\mathcal{P}$ represents novel task-specific adaptations introduced during fine-tuning, capturing input adaptability to task specific representation.

3.4 NeuroMerging

Our proposed NeuroMerging merges neuronal task vectors along two complementary neuronal spaces:

$$\begin{aligned}\overline{\tau}^k = {} & \lambda_1 \psi_{\parallel}\left(\tau_{\parallel,1}^k, \ldots, \tau_{\parallel,t}^k, \ldots, \tau_{\parallel,T}^k\right) \\ & + \lambda_2 \psi_{\perp}\left(\tau_{\perp,1}^k, \ldots, \tau_{\perp,t}^k, \ldots, \tau_{\perp,T}^k\right)\end{aligned} \tag{7}$$

where $\psi_{\parallel}(\cdot)$ denotes the merging function for $\tau_{\parallel,t}^k \in \mathbb{R}$, which can be a commonly used weighted average, TIES's disjoint merge (Yadav et al., 2023), and others. $\psi_{\parallel}(\cdot)$ is also applied to non-neuronal parameters such as bias and pre-norm. For $\tau_{\perp,t}^k \in \mathbb{R}^T$, we first find dominant orthogonal subspaces within $\mathcal{O}$ using singular value decomposition (SVD) with rank being the number of tasks interact within the same neuron, and then project $\tau_{\perp,t}^k$ along each dimension of the SVD subspace, before applying $\psi_{\parallel}(\cdot)$ to them. Detailed discussion on SVD is provided in Appendix A.3.

λ_1 and λ_2 are scaling parameters. When validation data is available λ_1 and λ_2 could be tuned. However, when validation is unavailable, we propose setting $\lambda_1 = 0$ (as the corresponding subspace is observed to have little impact in Section 6.3) and estimate λ_2 based on the impact of top $r\%$ mentioned in Section 3.1. Specifically, λ_2 is estimated as $\lambda_2 = \frac{1}{1-\sigma}$, where $\sigma = max(\sigma_1, ..., \sigma_t, ..., \sigma_T)$ and $\sigma_t = \frac{\|\tau_t^{masked}\|_1}{\|\tau_t\|_1}$ is the ratio of the $L_1 - Norm$ of the zeroed-out elements in the task vector τ_t^{masked} to the $L_1 - Norm$ of the original task vector τ_t. Detailed discussion on the parameters is provided in Section 6.3.

Subsequently, we reconstruct the task vector $\overline{\tau}$ with the all the merged neuronal task vectors $\overline{\tau}_k$. Finally, we obtained the final merged model $\overline{\theta} = \theta_0 + \overline{\tau}_{rescaled}$. The complete procedure is presented in Algorithm 1, with additional discussion on the motivation provided in Appendix B.

4 Experimental Setup

Baseline Methods. Our baselines comprise two main categories: (1) non-model merging approaches, which include individually fine-tuned models and a multitask model trained jointly on the combined dataset serving as our theoretical upper bound, and (2) various advanced model merging techniques, including Simple Averaging (Wortsman et al., 2022), Fisher Merging (Matena and Raffel, 2022), RegMean (Jin et al., 2023), Task Arithmetic (Ilharco et al., 2023), TIES-Merging (Yadav et al., 2023), AdaMerging (Yang et al., 2024b),

Task(→)			**Test Set Performance**						
Method(↓)	**Validation**	**Average**	paws	qasc	quartz	story_cloze	wiki_qa	winogrande	wsc
Zeroshot	-	53.1	58.2	54.2	54.1	54.3	70.9	49.2	63.9
Finetuned	-	88.0	94.4	97.1	85.3	91.0	95.7	71.6	80.6
Multitask	-	88.1	94.2	98.5	89.3	92.0	95.4	73.5	73.6
Averaging[ICML22]	✗	51.6	59.2	26.3	69.6	53.8	67.3	49.1	36.1
Task Arithmetic[ICLR23]	✗	59.7	60.9	31.7	57.8	73.0	73.5	55.7	65.3
TIES-Merging[NeurIPS23]	✗	76.7	80.8	92.4	77.7	81.9	78.4	61.9	63.9
PCB-Merging[NeurIPS24]	✗	76.9	**82.9**	93.2	79.0	84.4	75.6	**63.5**	59.7
NeuroMerging (Ours)	✗	**77.6**	81.1	**94.3**	**81.6**	**84.7**	**81.2**	56.7	**83.9**
Fisher Merging[NeurIPS22]	✓	68.7	68.4	83.0	65.5	62.4	**94.1**	58.2	49.2
RegMean[ICLR23]	✓	79.8	83.9	**97.2**	73.2	82.6	**94.1**	63.2	64.4
Task Arithmetic[ICLR23]	✓	74.6	72.7	91.3	**76.4**	85.6	74.4	61.0	61.1
TIES-Merging[NeurIPS23]	✓	79.5	82.6	94.9	72.8	87.4	85.2	**66.6**	66.7
PCB-Merging[NeurIPS24]	✓	81.0	**87.0**	95.2	**76.4**	**88.1**	88.4	64.3	68.1
NeuroMerging (Ours)	✓	**82.2**	86.4	94.3	75.9	87.9	91.2	65.9	**73.6**

Table 3: Test set performance when merging T5-Large models on seven NLP tasks.

PCB-Merging (Du et al., 2024), NPS (Du et al., 2025), and our proposed NeuroMerging method. Notably, we used task-wise AdaMerging for fair comparison with existing baselines and our method. See Section 6.5 for further discussion and analysis. We reported average accuracy across all tasks' test sets as our primary evaluation metric.

Validation Set Availability. Previous works exhibited varying dependencies on a validation set. Fisher Merging inherently required a validation set to compute the Fisher matrix. Other approaches may optionally utilize validation data for hyperparameter tuning, while RegMean leverages training data to compute and store inner product matrices for model merging. However, since these matrices matched the dimensions of the original model, they introduced substantial storage and computational overhead, limiting scalability to larger models and more extensive merging tasks.

Task vector-based approaches such as Task Arithmetic, Ties-Merging, and PCB-Merging, along with our proposed NeuroMerging, are substantially more lightweight and efficient. These approaches are training-free and do not rely on a validation set, making them highly practical for real-world applications. To further evaluate this advantage, we conducted additional experiments comparing task vector-based methods in scenarios where validation sets were unavailable.

Hyperparameters. In the absence of an additional validation set, we set $\lambda = 1$ as the default value for all task-vector-based methods. For TIES-Merging and PCB-Merging, which required a masking ratio, we followed the settings of Yadav et al. (2023) and Du et al. (2024), applying $r = 0.2$ as the default value across all experiments. For NeuroMerging, we set a default masking ratio of $r = 0.15$, with λ_1 fixed at 0, while λ_2 was automatically adjusted according to the methodology described in Section 3.4.

When validation is allowed, we configure the non-diagonal multiplier α in RegMean to 0.9, except for the T5-base model, where it is set to 0.1. For Task Arithmetic, we performed a grid search over λ ranging from 0.2 to 1.5 with a step size of 0.1. For TIES-Merging, PCB-Merging, and NeuroMerging, we searched for the optimal masking ratio r in the range $[0.05, 0.2]$ with a step size of 0.05, and λ (λ_2 for NeuroMerging) from 0.8 to 5.0 with a step size of 0.1.

5 Results

5.1 Merging NLP Models

Following the experimental settings from Yadav et al. (2023), we used the T5-base and T5-large models (Raffel et al., 2020), which were encoder-decoder transformers (Vaswani et al., 2017) pre-trained via masked language modeling on a large text corpus, and fine-tuned them independently on seven tasks: Khot et al.'s (2020) QASC, Yang et al.'s (2015) WikiQA, and Tafjord et al.'s (2019) QuaRTz for Question Answering; Zhang et al.'s (2019) PAWS for Paraphrase Identification; Sharma et al.'s (2018) Story Cloze for Sentence Completion; and Sakaguchi et al.'s (2020) Winogrande together with Levesque et al.'s (2012) WSC for Coreference Resolution. Tables 3 and 13 demonstrated that our approach achieved superior performance over state-of-the-art methods, achieving improvements of 0.6% and 1.2% for T5-base and T5-large, respectively. Moreover, NeuroMerging without validation showed even more substantial gains, surpassing previous methods by 1.4% for T5-base and 0.7% for T5-large. For comprehensive results across all tasks and model variants, see Appendix C.1 Tables 3 and 13, with detailed error analysis provided in Appendix C.4.

5.2 Out-of-Domain Generalization

Building upon the experimental setup of Yadav et al. (2023), we also investigated how merged models between tasks enhance generalization in different domains. Following the approach used in prior NLP models, we merged models on seven in-domain datasets and evaluated their performance on six held-out datasets from the T0 mixture (Sanh et al., 2022) to assess cross-task generalization. These datasets encompass diverse tasks, covering three Question Answering datasets: Huang et al.'s (2019) Cosmos QA, Sap et al.'s (2019) Social IQA, and Rogers et al.'s (2020) QuAIL; one Word Sense Disambiguation dataset: Pilehvar and Camacho-Collados's (2019) WiC; and two Sentence Completion datasets: Gordon et al.'s (2012) COPA and Zellers et al.'s (2019) H-SWAG. As shown in Figure 4, NeuroMerging outperformed the strongest baseline by 0.7% and 0.5% for T5-base and T5-large models, respectively, showcasing enhanced out-of-domain generalization capabilities. For more comprehensive results, please refer to Appendix C.1 Tables 16 and 17, with detailed error analysis provided in Appendix C.4.

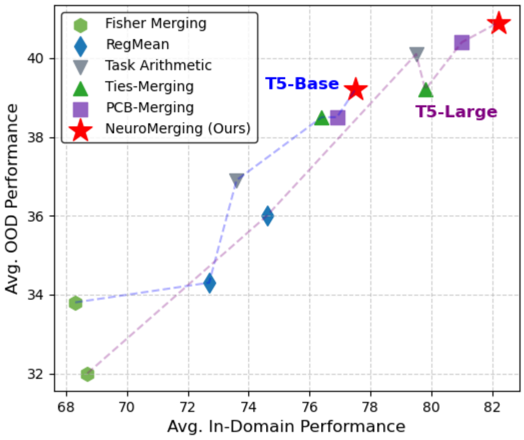

Figure 4: In-domain v.s. Out-domain performance.

5.3 Merging LLMs

We followed the experimental setup of Du et al. (2024) and extended our NeuroMerging to larger LLMs. Specifically, we merged three domain-specialized Llama-2-7b models (Touvron et al., 2023), and each was fine-tuned for distinct capabilities: Chinese language understanding, mathematical reasoning (Yu et al., 2024b), and code generation (Roziere et al., 2023). To rigorously evaluate the performance of each specialized model, we employed established benchmarks tailored to their respective domains: Li et al.'s (2024) CMMLU for assessing Chinese language proficiency, Cobbe et al.'s (2021) GSM8K for mathematical capabilities, and Chen et al.'s (2021) HumanEval for code generation competency. As demonstrated in Table 4, our approach exhibited substantial performance improvements, surpassing the strongest baseline by 0.9%. Notably, our method even outperformed the PCB-Merging utilizing Evolution Strategies (ES) optimization algorithm by 0.7%, underscoring the effectiveness of our proposed methodology. Appendix C.4 provides the error analysis.

Model	CMMLU	GSM8K	Human-Eval	Average
Chinese	38.6	2.3	13.4	18.1
Math	31.2	65.6	0.0	32.3
Code	33.3	0.0	17.1	16.8
Averaging[ICML22]	35.6	48.5	6.7	30.3
Task Arithmetic[ICLR23]	35.4	46.1	9.8	30.4
TIES-Merging[NeurIPS23]	**36.5**	53.4	12.8	34.3
Consensus TA[ICML24]	-	-	-	33.5
Consensus TIES[ICML24]	-	-	-	34.4
PCB-Merging[NeurIPS24]	36.4	52.3	**16.5**	35.1
PCB-Merging+ES[NeurIPS24]	36.4	53.1	**16.5**	35.3
NPS[ACL25]	-	-	-	35.3
NeuroMerging (Ours)	36.1	**57.2**	14.6	**36.0**

Table 4: Performance comparison on LLaMA2.

5.4 Merging Vision Models

We also examined the modality of vision by adhering to the experimental setup outlined by Ilharco et al. (2022, 2023). Specifically, we adopted two variants of the CLIP model (Radford et al., 2021), ViT-B/32 and ViT-L/14, as visual encoders (Dosovitskiy et al., 2021). For a fair comparison, we

Method	Validation	ViT-B/32 Avg.	ViT-L/14 Avg.
Individual	-	90.5	94.2
Multi-task	-	88.9	93.5
Averaging[ICML22]	✗	65.8	79.6
Task Arithmetic[ICLR23]	✗	60.4	83.3
TIES-Merging[NeurIPS23]	✗	72.4	86.0
PCB-Merging[NeurIPS24]	✗	75.9	86.9
NeuroMerging (Ours)	✗	**76.4**	**87.9**
Fisher Merging[NeurIPS22]	✓	68.3	82.2
RegMean[ICLR23]	✓	71.8	83.7
Task Arithmetic[ICLR23]	✓	70.1	84.5
TIES-Merging[NeurIPS23]	✓	73.6	86.0
AdaMerging[ICLR2024]	✓	71.1	84.9
AdaMerging++[ICLR2024]	✓	73.7	87.3
PCB-Merging[NeurIPS24]	✓	76.3	87.5
NeuroMerging (Ours)	✓	**76.5**	**88.3**

Table 5: Performance comparison on ViT.

T5-large	In-domain	Out-domain	Total Average
Fine-tuned	88.0	53.8	58.7
Keep Orthogonal	**88.0**	**53.9**	**58.8**
Keep Parallel	52.7	51.8	51.9

Table 6: Role of orthogonal and parallel subspaces on T5-Large, details provided in Appendix Table 18.

obtained fine-tuned checkpoints from Ilharco et al. (2023), which were consistently utilized across all baseline methods. This comprehensive evaluation spans multiple classification domains, encompassing remote sensing, traffic analysis, and satellite imagery recognition, with evaluations conducted on standard benchmark datasets, including Cars (Krause et al., 2013), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2018), GTSRB (Stallkamp et al., 2011), MNIST (LeCun, 1998), RESISC45 (Cheng et al., 2017), SUN397 (Xiao et al., 2016), and SVHN (Netzer et al., 2011). Table 5 presented the results of NeuroMerging, demonstrating its competitive performance across different validation scenarios. When employing validation data, our method achieved performance improvements of 0.2% for ViT-B/32 and 0.8% for ViT-L/14 over state-of-the-art baselines. In the absence of additional validation, NeuroMerging further improved upon the strongest baseline by 0.5% and 1.0% for ViT-B/32 and ViT-L/14, respectively. These results substantiated the broad model compatibility of our approach. For comprehensive results across all tasks and model variants, see Appendix C.1 Tables 14 and 15, with detailed error analysis provided in Appendix C.4.

6 Additional Results and Analysis

6.1 Merging without Validation Sets

When validation data is unavailable, we examined the parameters λ_1 and λ_2 selected according to Section 3.4, where λ_1 was set to zero, as the corresponding subspace was observed to have little impact in Section 6.3. The value of λ_2 was computed based on the L1-norm of masked and unmasked task vectors. Figure 8 and Table 3 and Appendix Tables 3, 14, and 15 presented the evaluation of NeuroMerging on NLP and CV tasks across various model sizes, comparing it with existing methods. NeuroMerging achieves the highest average accuracy across all tasks. This demonstrated that our proposed method, with a simple rescaling, outperformed existing methods on average. Specifically, it achieved a 1.4% and 0.7% improvement over the strongest baseline for T5-Base and T5-Large, respectively. For vision models, it outperformed the strongest baseline by 0.5% and 1.0% for ViT-B/32 and ViT-L/14, respectively.

6.2 Role of Neuronal Subspaces

Figure 2, Table 6 and Appendix Table 18 illustrated the impacts of neuronal subspace decomposition on T5-Large. To examine the impact of each subspace, ablation was performed separately on each subspace by retaining one while removing the other. Retaining the orthogonal subspace while removing the parallel subspace preserved near-perfect performance of finetuned checkpoints or even improved them across most tasks for T5-Large, achieving 88.0% in-domain, 53.9% out-of-domain, and an average of 58.8%. In contrast, keeping the parallel subspace while removing the orthogonal subspace resulted in a significant performance drop.

6.3 Robustness of Hyperparameters

We systematically investigated the impact of hyperparameters on merging performance: λ_1 and λ_2, which control the parallel and orthogonal subspace contributions, respectively, and the mask ratio r. **Relationship Between λ_1 and λ_2.** To examine the effects of λ_1 and λ_2, we conducted a grid search with $\lambda_1 \in [0, 1.0]$ and $\lambda_2 \in [3.0, 4.0]$ at 0.1 intervals, fixing $r = 10\%$. As visualized in Figure 5, the performance exhibited column-wise uniformity in the heatmap, indicating insensitivity to variations in λ_1, which aligned with our earlier discussion on the role of the orthogonal subspace in Section 6.2. This highlights a practical strength of our approach: when computational resources are limited, users can simplify their merging tasks by primarily tuning λ_2, as adjusting λ_1 yields rel-

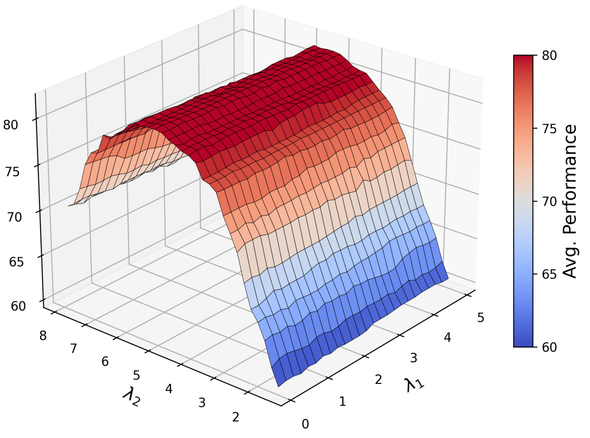

Figure 5: Impacts of λ_1 and λ_2 on T5-Large.

atively subtle effects. The optimal performance occurred at $\lambda_2 = 3.6$, attributed to the substantial proportion of masked variables.

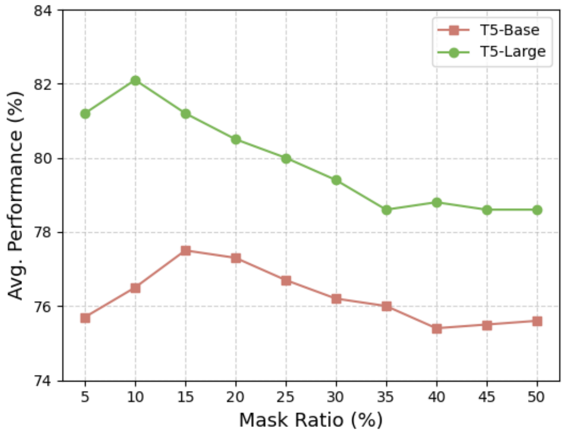

Figure 6: Impacts on mask ratio r.

Masking Ratio r. When validation is available, we analyzed the impact of the mask ratio on performance, showed in Figure 6. We observed robust performance across different ratios, with accuracy peaked at 15% and 10% for T5-Base and T5-Large, respectively. Performance variations remained bounded (within 2.5% for T5-Base and 4% for T5-Large) and stabilized beyond 35%. Maintaining r at 10%–20% improved performance, whereas exceeding this range often led to degradation. This suggested that an optimal masking ratio balances information retention and redundancy reduction.

6.4 Ablation Study on Merging Functions

We conducted ablation experiments on various merging functions $\psi(\cdot)$ to evaluate their effectiveness in combining numerics. As shown in Table 7, among all merging functions, the *elect+mean* approach from TIES-Merging achieves the highest performance at 82.2%. In comparison, using *elect+sum*, *averaging*, and *sum* methods resulted in performance decreases of 2.6%, 2.5%, and 2.4%, respectively.

Method	Avg. Acc
elect + mean	**82.2**
elect + sum	79.6
mean	79.7
sum	79.8

Table 7: Comparison of $\psi(\cdot)$ on average accuracy.

6.5 NeuroMerging + AdaMerging

We adopted task-wise AdaMerging (Yang et al., 2024b) for fair comparison, as the layer-wise variant incurs substantial computational cost due to tuning individual layer- and tasks-specific λ values via expensive backpropagation on a model larger than the pre-trained one. Further computational efficiency analysis between different methods is detailed in Appendix C.3.

As shown in Table 8, our method consistently improves the performance of both layer-wise AdaMerging and AdaMerging++ when applied to their publicly shared layer-wise λ directly. Notably, we observe a greater improvement when combining AdaMerging with NeuroMerging compared to AdaMerging++, suggesting that the TIES component within AdaMerging++ may inadvertently discard information beneficial to NeuroMerging.

Method	ViT-B/32	ViT-L/14
Layer-wise AdaMerging	80.1	90.8
+NeuroMerging	**82.3** (+2.2)	**91.3** (+0.5)
Layer-wise AdaMerging++	81.1	91.0
+NeuroMerging	**82.5** (+1.4)	**91.0** (+<0.1)

Table 8: Layer-wise enhancement on NeuroMerging.

7 Conclusion

In this paper, we revisited model merging from the core principle neuron connectivity and activation that underpin the learning process of recent deep neural networks and LLMs. Specifically, we presented the first exploration into the roles of neuronal mechanisms in the model merging process, by decomposing of task-specific representations into two complementary neuronal subspaces for characterization. It was mathematically shown that neuronal subspaces regulate input sensitivity and task adaptability. Based on these insights, we proposed NeuroMerging, a novel framework designed to reduce task interference within neurons. Our empirical evaluations demonstrated that NeuroMerging achieved superior performance compared to existing approaches on multi-task benchmarks across both vision and natural language processing tasks.

Future work could extend NeuroMerging to larger models or multimodal architectures with more tasks. While our study introduced a neuronal mechanistic perspective for in model merging, it focused only on neuron connectivity and activation, without considering higher-order neuronal dynamics (e.g., network level connectivity and dynamics). Further research is needed to investigate these factors and deepen our understanding of neuronal mechanisms in model merging, or even in multi-task learning.

Acknowledgements

This work was supported in part by the Ministry of Higher Education Malaysia through the Fundamental Research Grant Scheme (FRGS/1/2023/IC T02/XMU/02/1), National Science Foundation of China (12204130, 62476070), Shenzhen Science and Technology Program (JCYJ202412021235030 05, GXWD20231128103232001), Department of Science and Technology of Guangdong (2024A15 15011540), Shenzhen Start-Up Research Funds (H A11409065), and Xiamen University Malaysia Research Fund (XMUMRF/2024-C13/IECE/0049).

Limitations

While our work provided the first neuronal mechanistic perspective on multi-task interference when merging large models, (1) it remains a partial view of neuronal mechanisms as it does not yet explore higher order neuronal dynamics during merging and inference, which require further investigation. Moreover, this work shares similar limitations with current SOTA model merging methods, including (2) the effectiveness of task arithmetic in model merging relies on selecting fine-tuned checkpoints that are beneficial for specific domains, ensuring they originate from the same pretrained model, and addressing hyperparameter sensitivity; and (3) More effort is needed to develop a mathematical understanding of why and when task arithmetic in model merging works well, despite its simplicity and efficiency.

Ethical Considerations

Our research is based on publicly available datasets and models, all of which conform to their respective licenses and ethical guidelines. While the proposed NeuroMerging approach itself introduces no immediate ethical risks, caution should be exercised when generalizing results beyond the evaluated domains. Specifically, applying this technique to privacy-sensitive or high-stakes scenarios may require additional validation, as performance and reliability in untested contexts remain uncertain. We thus recommend thorough evaluation and responsible deployment practices in such applications.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Takuya Akiba, Makoto Shing, Yujin Tang, Qi Sun, and David Ha. 2025. Evolutionary optimization of model merging recipes. *Nature Machine Intelligence*, 7(2):195–204.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. 2022. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 32897–32912. Curran Associates, Inc.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

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

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3606–3613.

Roberto Cipolla, Yarin Gal, and Alex Kendall. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7482–7491.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias

Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*.

Guodong Du, Zitao Fang, Jing Li, Junlin Li, Runhua Jiang, Shuyang Yu, Yifei Guo, Yangneng Chen, Sim Kuan Goh, Ho-Kin Tang, Daojing He, Honghai Liu, and Min Zhang. 2025. Neural parameter search for slimmer fine-tuned models and better transfer. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 32668–32687, Vienna, Austria. Association for Computational Linguistics.

Guodong Du, Junlin Lee, Jing Li, Runhua Jiang, Yifei Guo, Shuyang Yu, Hanting Liu, Sim Kuan Goh, Ho-Kin Tang, Daojing He, and Min Zhang. 2024. Parameter competition balancing for model merging. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pages 84746–84776. Curran Associates, Inc.

Sebastian Dziadzio, Vishaal Udandarao, Karsten Roth, Ameya Prabhu, Zeynep Akata, Samuel Albanie, and Matthias Bethge. 2025. How to merge multimodal models over time? In *ICLR 2025 Workshop on Modularity for Collaborative, Decentralized, and Continual Deep Learning*.

Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. 2021. Efficiently identifying task groupings for multi-task learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 27503–27516. Curran Associates, Inc.

Andrew Gordon, Zornitsa Kozareva, and Melissa Roemmele. 2012. SemEval-2012 task 7: Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 394–398, Montréal, Canada. Association for Computational Linguistics.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 2018. Introducing eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 204–207.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. 2024a. Lorahub: Efficient cross-task generalization via dynamic loRA composition. In *First Conference on Language Modeling (COLM)*.

Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. 2024b. Emr-merging: Tuning-free high-performance model merging. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pages 122741–122769. Curran Associates, Inc.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos QA: Machine reading comprehension with contextual commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2391–2401, Hong Kong, China. Association for Computational Linguistics.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations (ICLR)*.

Gabriel Ilharco, Mitchell Wortsman, Samir Yitzhak Gadre, Shuran Song, Hannaneh Hajishirzi, Simon Kornblith, Ali Farhadi, and Ludwig Schmidt. 2022. Patching open-vocabulary models by interpolating weights. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 29262–29277. Curran Associates, Inc.

Md Tauhidul Islam, Zixia Zhou, Hongyi Ren, Masoud Badiei Khuzani, Daniel Kapp, James Zou, Lu Tian, Joseph C. Liao, and Lei Xing. 2023. Revealing hidden patterns in deep neural network feature space continuum via manifold learning. *Nature Communications*, 14(1):8506.

Chunheng Jiang, Zhenhan Huang, Tejaswini Pedapati, Pin-Yu Chen, Yizhou Sun, and Jianxi Gao. 2024. Network properties determine neural network performance. *Nature Communications*, 15(1):5718.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. 2023. Dataless knowledge fusion by merging weights of language models. In *The Eleventh International Conference on Learning Representations (ICLR)*.

Aecheon Jung, Seunghwan Lee, Dongyoon Han, and Sungeun Hong. 2024. Why train everything? tint a single layer for multi-task model merging. *arXiv preprint arXiv:2412.19098*.

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

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pages 5583–5594. PMLR.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *2013 IEEE International Conference on Computer Vision Workshops (ICCV)*, pages 554–561.

Yann LeCun. 1998. The mnist database of handwritten digits. `http://yann.lecun.com/exdb/mnist/`.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 552–561, Rome. AAAI Press.

Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2024. CMMLU: Measuring massive multitask language understanding in Chinese. In *Findings of the Association for Computational Linguistics (ACL)*, pages 11260–11285, Bangkok, Thailand. Association for Computational Linguistics.

Jian Liang, Ziqi Liu, Jiayu Zhou, Xiaoqian Jiang, Changshui Zhang, and Fei Wang. 2022. Model-protected multi-task learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 44(2):1002–1019.

Shikun Liu, Edward Johns, and Andrew J. Davison. 2019. End-to-end multi-task learning with attention. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1871–1880.

Zhenyi Lu, Chenghao Fan, Wei Wei, Xiaoye Qu, Dangyang Chen, and Yu Cheng. 2024. Twin-merging: Dynamic integration of modular expertise in model merging. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pages 78905–78935. Curran Associates, Inc.

Michael S Matena and Colin A Raffel. 2022. Merging models with fisher-weighted averaging. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 17703–17716. Curran Associates, Inc.

Michael Moor, Oishi Banerjee, Zahra Shakeri Hossein Abad, Harlan M Krumholz, Jure Leskovec, Eric J Topol, and Pranav Rajpurkar. 2023. Foundation models for generalist medical artificial intelligence. *Nature*, 616(7956):259–265.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. 2011. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. 2023. Task arithmetic in the tangent space: Improved editing of pre-trained models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 66727–66754. Curran Associates, Inc.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research (JMLR)*, 21(140):1–67.

Partha Pratim Ray. 2023. Chatgpt: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*, 3:121–154.

Anna Rogers, Olga Kovaleva, Matthew Downey, and Anna Rumshisky. 2020. Getting closer to ai complete question answering: A set of prerequisite real tasks. *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*, 34(05):8722–8731.

Frank Rosenblatt. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386.

Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, et al. 2023. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*, volume 34, pages 8732–8740.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan

Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. 2022. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations (ICLR)*.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China. Association for Computational Linguistics.

Rishi Sharma, James Allen, Omid Bakhshandeh, and Nasrin Mostafazadeh. 2018. Tackling the story ending biases in the story cloze test. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 752–757, Melbourne, Australia. Association for Computational Linguistics.

Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*.

Li Shen, Anke Tang, Enneng Yang, Guibing Guo, Yong Luo, Lefei Zhang, Xiaochun Cao, Bo Du, and Dacheng Tao. 2024. Efficient and effective weight-ensembling mixture of experts for multi-task model merging. *arXiv preprint arXiv:2410.21804*.

Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. 2011. The german traffic sign recognition benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks (IJCNN)*, pages 1453–1460.

Florian Stelzer, André Röhm, Raul Vicente, Ingo Fischer, and Serhiy Yanchuk. 2021. Deep neural networks using a single neuron: folded-in-time architecture using feedback-modulated delay loops. *Nature Communications*, 12(1):5164.

Ahmad Suhaimi, Amos W. H. Lim, Xin Wei Chia, Chunyue Li, and Hiroshi Makino. 2022. Representation learning in the artificial and biological neural networks underlying sensorimotor integration. *Science Advances*, 8(22):eabn0984.

Anirudh S. Sundar, Chao-Han Huck Yang, David M. Chan, Shalini Ghosh, Venkatesh Ravichandran, and Phani Sankar Nidadavolu. 2024. Multimodal attention merging for improved speech recognition and audio event classification. In *2024 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*, pages 655–659.

Yi-Lin Sung, Linjie Li, Kevin Lin, Zhe Gan, Mohit Bansal, and Lijuan Wang. 2023. An empirical study of multimodal model merging. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1563–1575, Singapore. Association for Computational Linguistics.

Oyvind Tafjord, Matt Gardner, Kevin Lin, and Peter Clark. 2019. QuaRTz: An open-domain dataset of qualitative relationship questions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5941–5946, Hong Kong, China. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30. Curran Associates, Inc.

Fanqi Wan, Xinting Huang, Deng Cai, Xiaojun Quan, Wei Bi, and Shuming Shi. 2024a. Knowledge fusion of large language models. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Fanqi Wan, Longguang Zhong, Ziyi Yang, Ruijun Chen, and Xiaojun Quan. 2024b. FuseChat: Knowledge fusion of chat models.

Haoxiang Wang, Pavan Kumar Anasosalu Vasu, Fartash Faghri, Raviteja Vemulapalli, Mehrdad Farajtabar, Sachin Mehta, Mohammad Rastegari, Oncel Tuzel, and Hadi Pouransari. 2024. Sam-clip: Merging vision foundation models towards semantic and spatial understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3635–3647.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pages 23965–23998. PMLR.

Jianxiong Xiao, Krista A. Ehinger, James Hays, Antonio Torralba, and Aude Oliva. 2016. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision (IJCV)*, 119(1):3–22.

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

(NeurIPS), volume 36, pages 7093–7115. Curran Associates, Inc.

Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xiaojun Chen, Xingwei Wang, and Dacheng Tao. 2024a. Representation surgery for multi-task model merging. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pages 56332–56356. PMLR.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. 2024b. Adamerging: Adaptive model merging for multi-task learning. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2013–2018, Lisbon, Portugal. Association for Computational Linguistics.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024a. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pages 57755–57775. PMLR.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng YU, Zhengying Liu, Yu Zhang, James Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2024b. Metamath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (NAACL-HLT)*, pages 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

A Additional Details

A.1 Checkpoints for LLaMA

We merged three domain-specialized Llama-2-7b models (Touvron et al., 2023), and each is fine-tuned for distinct capabilities: Chinese language understanding[1], mathematical reasoning[2] (Yu et al., 2024b), and code generation[3] (Roziere et al., 2023).

A.2 Masking Ratio

As entries of the task vector with larger magnitudes are more relevant to task-specific adaptation, only the top $r\%$ of τ_t with the largest magnitudes are kept, while the others were set to zero (Yadav et al., 2023). The masked task vector is defined as t, where m_t is the mask that keeps the top $r\%$ of the elements of each task vector. We used τ_t to represent the masked task vector for readability.

A.3 Singular Value Decomposition

We employed singular value decomposition (SVD) to construct a meaningful coordinate system within the orthogonal subspace $\mathcal{O}$, defined as the null space of the pre-trained neuron weights. Specifically, since $\mathcal{O}$ inherently lacks an explicit coordinate system, we first vertically stacked the task-specific neuron weight vectors τ_t^k into a matrix D, and then performed SVD on D as follows:

$$D = Q_k \Sigma_k P_k^\top \tag{8}$$

Here, k denotes the rank, which was set to be the number of tasks, and P_k contains the top right singular vectors, which represent the dominant, i.e., principal, directions of task variability in the orthogonal subspace $\mathcal{O}$. These directions serve as the coordinate axes for the orthogonal subspace.

Next, we projected the task weights onto these axes using DP_k, and apply the merging function $\psi_{\parallel}(\cdot)$ column-wise to aggregate the projected components into a merged task representation ζ^k in the low-dimensional space. The final reconstruction of the merged weights was given by $\zeta^k P_k^\top$.

A.4 Algorithm

Algorithm 1 is the pseudo-code for NeuroMerging.

```
Algorithm 1 NeuroMerging
Input: Task-specific models $\tau_1, \tau_2, \ldots, \tau_T$, pre-trained model $\theta_0$, mask $m_t$, mask ratio $r$, proj matrix $\mathbf{P}$
Output: Merged model $\bar{\theta}$
$\tau_t = m_t \circ \tau_t$ // Mask task vector based on $r$
for $k \leftarrow 1$ to $K$ do
    for $t \leftarrow 1$ to $T$ do
        ▷ Create neuronal task vector.
        $\tau_t^k = \mathbf{w}_t^k - \mathbf{w}_0^k$
        ▷ Decompose neuronal subspaces.
        $\tau_{\parallel,t}^k = \mathbf{P}\tau_t^k, \quad \tau_{\perp,t}^k = (\mathbf{I} - \mathbf{P})\tau_t^k$
    end
end
▷ Merge neuronal task vectors.
for $k \leftarrow 1$ to $K$ do
    if Validation data is available then
        Tune $\lambda_1$ and $\lambda_2$ using the validation dataset
    else
        $\sigma_t = \frac{\|\tau_t^{masked}\|_1}{\|\tau_t\|_1}, \quad \sigma = \max(\sigma_1, .., \sigma_T)$
        $\lambda_1 = 0, \quad \lambda_2 = \frac{1}{1-\sigma}$
    end
    $\bar{\tau}^k = \lambda_1 \psi_{\parallel}(\tau_{\parallel,1}^k, .., \tau_{\parallel,T}^k) + \lambda_2 \psi_{\perp}(\tau_{\perp,1}^k, .., \tau_{\perp,T}^k)$
end
▷ Reconstruct the merged task vector $\bar{\tau}$ by
  combining the $\bar{\tau}^k$ for each neuron.
$\bar{\theta} = \theta_0 + \bar{\tau}$ // final merged model
return $\bar{\theta}$
```

B Why Merging at the Neuronal Level?

From the statistical analysis of neuronal versus non-neuronal parameters, it was observed that neuronal parameters dominate the T5-Base and T5-Large, showed in Figure. 7. As a result, they play a major role in shaping the model's learning dynamics, task adaptability, and overall performance. This dominance suggests that understanding and optimizing neuronal parameter interactions is crucial for improving model merging, reducing task interference, and enhancing generalization across diverse tasks.

Moreover, we demonstrated in the main manuscript that merging in the orthogonal subspace ($\mathcal{O}$) is more effective. To explain why, we mathematically show that $\bar{x}_\perp$ does not influence the activation of the pre-trained neuron. For comparison, consider the activation of the pre-trained neuron:

$$y = \phi(\mathbf{w}_0^k \cdot \bar{x}) = \phi(\mathbf{w}_0^k \cdot (\bar{x}_\parallel + \bar{x}_\perp)) \tag{9}$$

Since $\bar{x}_\perp$ belongs to the complementary orthog-

[1] https://huggingface.co/LinkSoul/Chinese-Llama-2-7b

[2] https://huggingface.co/meta-math/MetaMath-7B-V1.0

[3] https://huggingface.co/qualis2006/llama-2-7b-int4-python-code-18k

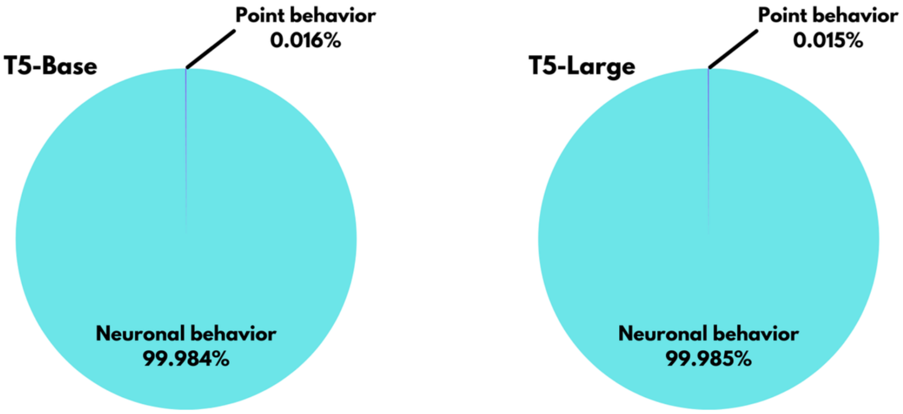

Figure 7: Statistical analysis of neuronal versus non-neuronal parameters.

onal null space of $\mathbf{w}_0^k$, we have:

$$\mathbf{w}_0^k \cdot \bar{x}_\perp = 0 \quad (10)$$

Due to vector product of orthogonal subspaces, we simplify the activation as:

$$y = \phi(\mathbf{w}_0^k \cdot \bar{x}_\parallel) \quad (11)$$

Hence, the orthogonal subspace ($\mathcal{O}$) encapsulates more task-specific adaptations, making it a more effective space for merging. In Table 2, we summarized the differences between our work with existing methods.

C Additional Results

C.1 Comprehensive Task-Level Results

We provided all task-level results in T5-Base, T5-Large (Raffel et al., 2020), LLaMA2 (Touvron et al., 2023), ViT-B/32, and ViT-L/14 (Dosovitskiy et al., 2021), respectively. The task-level results of the in-domain experiments for all models can be found in Tables 13, 14, 15. The task-level results of the out-domain experiments for T5-Base and T5-Large can be found in Tables 16 and 17. Lastly, Table 18 showed the task-level results from Section 6.2 when only one of the neuronal subspace was retained.

C.2 Existing Model Merging Methods in Neuronal Subspaces

To empirically validate the neuronal subspaces, we extended our experiments to various existing model merging methods. As shown in Tables 9 and 10, experimental results demonstrated that applying these merging methods within our proposed orthogonal neuronal subspaces consistently enhanced their performance. Notably, this improvement occurred despite these existing methods do not explicitly incorporating neuronal mechanisms into their designs. This observation further underscored the significance and motivation of our proposed approach.

C.3 Scalability Evaluation

We further evaluated the scalability of NeuroMerging from the perspective of computational efficiency. As summarized in table 11 and 12 (runtime in seconds, memory usage in GB; lower is better), NeuroMerging consistently achieved superior efficiency compared to existing methods across representative NLP and CV benchmarks. Specifically, it demonstrated comparable or reduced memory usage and improved accuracy relative to TIES-Merging (Yadav et al., 2023) (+2.7% on T5-Large, +2.3% on ViT-L/14, +1.7% on LLaMA2), significantly lower runtime and memory overhead compared to PCB-Merging (Du et al., 2024), and vastly reduced computational requirements relative to AdaMerging (Yang et al., 2024b), which incured up to 100 times longer runtime and 10 times higher memory usage on ViT-L/14 due to additional training steps. In general, unlike prior approaches (Yadav et al., 2023; Du et al., 2024; Yang et al., 2024b; Jung et al., 2024; Yang et al., 2024a; Huang et al., 2024b; Lu et al., 2024; Matena and Raffel, 2022), which either required loading all parameters simultaneously or introduce costly auxiliary components, NeuroMerging leveraged neuronal task vectors aligned with neuronal mechanisms, ensuring scalable, lightweight, and efficient merging.

C.4 Error Analysis

This section analyzes individual sub-tasks where NeuroMerging's performance is more than 3% lower compared to the best-performing baseline. The detailed breakdown covers NLP-series, Vision-series, and LLM-series tasks across various settings, including with and without validation, different model sizes, as well as in-domain and out-of-distribution (OOD) evaluations. For the quantitative analysis, we report both the performance differences and the single-task rankings. For the qualitative analysis, we offer possible explanations to interpret these observed differences.

NLP series. Performance details are summarized in Tables 3, 13, 16, and 17. On Winogrande, T5-Large without validation is behind by 6.8% (ranking 3/5), and T5-Base with validation by 6.6% (ranking 6/6); on Quartz, T5-Base with and without validation lag by 6.3% (ranking 4/6) and 3.3% (ranking 3/5), respectively; on Story Cloze, T5-Base without validation trails by 5.5% (ranking 2/5); for Cosmos QA in the out-of-distribution (OOD) setting, T5-Large is lower by 9.7% (ranking 6/8); and on WiC, T5-Large and T5-Base under OOD conditions show performance gaps of 6.1% (ranking 3/8) and 13.6% (ranking 3/8), respectively.

These tasks typically require deeper semantic reasoning or more fine-grained contextual alignment compared to other tasks. Winogrande (Sakaguchi et al., 2020) and WiC (Pilehvar and Camacho-Collados, 2019) both rely heavily on nuanced lexical disambiguation or pronoun resolution. In these scenarios, the multi-task merging process, especially under limited model capacity or validation scenarios, may inadvertently ignore task-specific weights, diluting fine-grained semantic signals. Quartz (Tafjord et al., 2019) and Cosmos QA (Huang et al., 2019) demand coherent integration of multi-sentence causal chains and background knowledge. However, parameter interference from shorter-text tasks occasionally compresses the reasoning chain, thus restricting deep semantic extraction. Story Cloze (Sharma et al., 2018) emphasizes narrative coherence modeling, where different optimal configuration conflicts among merged tasks can weaken sensitivity to long-range dependencies. Overall, the performance gaps identified mainly arise from representational shifts induced by cross-task distributional differences, coupled with mismatches between model capacity and task complexity. These findings indicate promising directions for future research on model merging.

Vision series. Performance details are summarized in Tables 5, 14 and 15. On SUN397 (Xiao et al., 2016), ViT-B/32 with validation lag by 3.3% (ranking 3/8); and on Cars (Krause et al., 2013), ViT-B/32 with validation is behind by 3.5% (ranking 2/8), with results similar to other merging methods except for Fisher Merging.

Fisher Merging demonstrated strong performance on these datasets, likely because it leverages the Fisher Information Matrix to selectively emphasize task-critical parameters on these two datasets (e.g., SUN397 requires panoramic scene composition, while Cars hinges on sub-class visual micro-signatures) through computing the Fisher Information Matrix using second-order derivatives or approximation through backpropagation. However, the high computational cost of the Fisher Information Matrix and its dependence on data make it less practical. As a result, recent model merging methods often avoid using the Fisher Information Matrix despite its effectiveness on these two datasets.

LLM series. Performance details are summarized in Table 4. NeuroMerging does not exhibit any sub-task with a performance drop exceeding 3% in this scenario.

D Implementation Details

We executed NLP and CV experiments on Nvidia GeForce 4090 GPUs with 24GB RAM, and LLM experiments on Nvidia A6000 GPUs with 48GB RAM, respectively. The merging experiments demonstrated highly computational efficiency, with evaluation times under 2 minutes for T5-Base, T5-Large, ViT-B/32, and ViT-L/14 models. For large language model, specifically LLaMA2, the validation process across three datasets required approximately 40 minutes per complete evaluation cycle.

E Dataset Details

We utilized several datasets, and each of them comes with specific licenses. The following datasets are available under the Creative Commons License: WiC (Pilehvar and Camacho-Collados, 2019), WSC (Levesque et al., 2012), Story Cloze (Sharma et al., 2018), QuaRTz (Tafjord et al., 2019), Cars (Krause et al., 2013), and GTSRB (Stallkamp et al., 2011). Winogrande (Sakaguchi et al., 2020) and

Task(→)
Method(↓)
	T5-Base
	Acc. (%)
		
		Orth. ($\mathcal{O}$)
			
			Par. ($\mathcal{P}$)
				T5-Large
				Acc. (%)
					
					Orth. ($\mathcal{O}$)
						
						Par. ($\mathcal{P}$)

TIES-Merging[NeurIPS2023]	**76.4**	76.2	55.5	79.5	**79.9**	52.5
PCB-Merging[NeurIPS2024]	76.9	**77.1**	54.7	81.0	**81.8**	52.0
NeuroMerging (Ours)	**77.5**	**77.5**	54.6	**82.2**	**82.2**	52.2

Table 9: NLP tasks' accuracy under orthogonal and parallel projections across T5-Large and T5-Base models.

Task(→)
Method(↓)
	ViT-B/32
	Acc. (%)
		
		Orth. ($\mathcal{O}$)
			
			Par. ($\mathcal{P}$)
				ViT-L/14
				Acc. (%)
					
					Orth. ($\mathcal{O}$)
						
						Par. ($\mathcal{P}$)

Task-wise AdaMerging[ICLR2024]	71.1	**71.8**	48.9	84.9	**85.0**	65.1
Task-wise AdaMerging++[ICLR2024]	73.7	**74.5**	49.2	87.3	**87.5**	65.6
Layer-wise AdaMerging[ICLR2024]	80.1	**80.4**	50.4	90.8	**91.0**	66.1
Layer-wise AdaMerging++[ICLR2024]	**81.1**	**81.1**	50.3	91.0	**91.2**	65.9
PCB-Merging[NeurIPS2024]	76.3	**76.4**	48.7	87.5	**87.8**	65.0
NeuroMerging (Ours)	76.5	**76.6**	48.7	**88.3**	**88.3**	65.1

Table 10: Vision tasks' accuracy under orthogonal and parallel projections across ViT-B/32 and ViT-L/14 models.

Task(→)
Method(↓)
	T5-Base
	Time (s,↓)
		
		Mem (GB,↓)
			
			Acc. (%,↑)
				T5-Large
				Time (s,↓)
					
					Mem (GB,↓)
						
						Acc. (%,↑)

TIES-Merging[NeurIPS2023]	**17**	61.4	76.4	**50**	190.1	79.5
PCB-Merging[NeurIPS2024]	98	105.5	76.9	323	312.1	81.0
NeuroMerging (Ours)	67	**18.7**	**77.5**	161	**69.1**	**82.2**

Table 11: Comparison on computational efficiency and accuracy for NLP tasks.

Task(→)
Method(↓)
	ViT-B/32
	Time (s,↓)
		
		Mem (GB,↓)
			
			Acc. (%,↑)
				ViT-L/14
				Time (s,↓)
					
					Mem (GB,↓)
						
						Acc. (%,↑)

TIES-Merging[NeurIPS2023]	**12**	28.4	73.6	**36**	84.7	86.0
PCB-Merging[NeurIPS2024]	51	45.4	76.3	149	137.1	87.5
AdaMerging[ICLR2024]	6793	80.1	71.1	18871	265.4	84.9
AdaMerging++[ICLR2024]	6384	95.7	73.1	18184	335.7	87.3
NeuroMerging (Ours)	67	**8.8**	**76.5**	141	**21.7**	**88.3**

Table 12: Comparison on computational efficiency and accuracy for vision tasks.

QASC (Khot et al., 2020) are distributed under the Apache License, while COPA (Gordon et al., 2012) is covered by the BSD-2 Clause License. WikiQA (Yang et al., 2015) is governed by the Microsoft Research Data License Agreement. Cosmos QA (Huang et al., 2019) is licensed under the CC BY 4.0. QuAIL (Rogers et al., 2020) and CMMLU (Li et al., 2024) are licensed under the CC BY-NC-SA 4.0. H-SWAG (Zellers et al., 2019), GSM8K (Cobbe et al., 2021), HumanEval (Chen et al., 2021), and EuroSAT (Helber et al., 2018) fall under the MIT License, and MNIST (LeCun, 1998) is licensed under the GNU General Public License.

For the datasets DTD (Cimpoi et al., 2014), RESISC45 (Cheng et al., 2017), SUN397 (Xiao et al., 2016), SVHN (Netzer et al., 2011), Social IQA (Sap et al., 2019), and PAWS (Zhang et al., 2019), we were unable to determine specific licenses. However, they are publicly shared for research and education purposes.

Task(→)
Method(↓)
	Validation	Average	Test Set Performance
			paws
				
				qasc
					
					quartz
						
						story_cloze
							
							wiki_qa
								
								winogrande
									
									wsc

Zeroshot	-	53.5	49.9	35.8	53.3	48.1	76.2	50.0	61.1
Finetuned	-	79.6	93.9	98.0	81.4	82.5	95.4	51.9	54.2
Multitask	-	83.6	94.0	97.9	82.5	86.7	95.0	64.1	65.3
Averaging[ICML22]	✗	64.2	65.1	81.1	59.9	48.8	94.7	50.9	48.6
Task Arithmetic[ICLR23]	✗	60.6	78.5	30.0	56.8	65.6	95.0	49.6	48.6
TIES-Merging[NeurIPS23]	✗	73.6	82.4	94.1	71.9	66.0	91.2	**51.5**	58.3
PCB-Merging[NeurIPS24]	✗	73.4	**83.1**	92.6	**72.6**	**73.4**	88.0	**51.5**	52.8
NeuroMerging (Ours)	✗	**74.8**	81.8	**96.5**	69.3	67.9	**94.8**	51.0	**62.5**
Fisher Merging[NeurIPS22]	✓	68.3	66.7	85.6	63.5	57.1	90.1	54.2	60.8
RegMean[ICLR23]	✓	72.7	77.2	93.8	63.6	64.6	90.4	**58.4**	60.7
Task Arithmetic[ICLR23]	✓	73.6	83.2	89.9	69.3	72.9	**95.2**	52.1	52.8
TIES-Merging[NeurIPS23]	✓	76.4	**88.6**	94.1	**74.5**	75.6	92.1	53.2	56.9
PCB-Merging[NeurIPS24]	✓	76.9	88.2	95.2	71.0	**77.3**	95.1	51.9	59.7
NeuroMerging (Ours)	✓	**77.5**	87.5	**95.7**	68.2	76.8	94.5	51.8	**68.1**

Table 13: Test set performance when merging T5-Base models on seven NLP tasks. Please refer to Section 5.1 for experimental details and Section C.4 for error analysis.

Task(→)
Method(↓)
	Validation	Average	Test Set Performance
			SUN397
				
				Cars
					
					RESISC45
						
						EuroSAT
							
							SVHN
								
								GTSRB
									
									MNIST
										
										DTD

Individual	-	90.5	75.3	77.7	96.1	99.7	97.5	98.7	99.7	79.4
Multitask	-	88.9	74.4	77.9	98.2	98.9	99.5	93.9	72.9	95.8
Averaging[ICML22]	✗	65.8	65.3	63.4	71.4	71.7	64.2	52.8	87.5	50.1
Task Arithmetic[ICLR23]	✗	60.4	36.7	41.0	53.8	64.4	80.6	66.0	98.1	42.5
TIES-Merging[NeurIPS23]	✗	72.4	59.8	58.6	70.7	79.7	**86.2**	72.1	98.3	54.2
PCB-Merging[NeurIPS24]	✗	75.9	**65.8**	**64.4**	**78.1**	81.1	84.9	77.1	98.0	58.4
NeuroMerging (Ours)	✗	**76.4**	64.7	64.2	77.0	**83.9**	**86.2**	**78.0**	**98.5**	**58.7**
Fisher Merging[NeurIPS22]	✓	68.3	**68.6**	**69.2**	70.7	66.4	72.9	51.1	87.9	**59.9**
RegMean[ICLR23]	✓	71.8	65.3	63.5	75.6	78.6	78.1	67.4	93.7	52.0
Task Arithmetic[ICLR23]	✓	70.1	63.8	62.1	72.0	77.6	74.4	65.1	94.0	52.2
TIES-Merging[NeurIPS23]	✓	73.6	64.8	62.9	74.3	78.9	83.1	71.4	97.6	56.2
AdaMerging[ICLR2024]	✓	71.1	58.0	53.2	68.8	**85.7**	81.1	**84.4**	92.4	44.8
AdaMerging++[ICLR2024]	✓	73.7	60.8	56.9	73.1	83.4	**87.3**	82.4	95.7	50.1
PCB-Merging[NeurIPS24]	✓	76.3	66.7	65.5	**78.5**	79.3	86.4	77.1	98.2	59.1
NeuroMerging (Ours)	✓	**76.5**	65.3	65.7	77.1	84.8	84.5	77.9	**98.3**	58.5

Table 14: Test set performance when merging ViT-B/32 models on eight vision tasks. Please refer to Section 5.4 for experimental details and Section C.4 for error analysis.

Task(→)
Method(↓)
	Validation	Average	Test Set Performance
			SUN397
				
				Cars
					
					RESISC45
						
						EuroSAT
							
							SVHN
								
								GTSRB
									
									MNIST
										
										DTD

Individual	-	94.2	82.3	92.4	97.4	100	98.1	99.2	99.7	84.1
Multitask	-	93.5	90.6	84.4	99.2	99.1	99.6	96.3	80.8	97.6
Averaging[ICML22]	✗	79.6	72.1	81.6	82.6	91.9	78.2	70.7	97.1	62.8
Task Arithmetic[ICLR23]	✗	83.3	72.5	79.2	84.5	90.6	89.2	86.5	99.1	64.3
TIES-Merging[NeurIPS23]	✗	86.0	76.5	85.0	89.3	95.7	**90.3**	83.3	99.0	68.8
PCB-Merging[NeurIPS24]	✗	86.9	75.8	86.0	89.2	96.0	88.0	90.9	99.1	70.0
NeuroMerging (Ours)	✗	**87.9**	**77.0**	**86.9**	**90.3**	**96.3**	89.9	**92.1**	**99.2**	**71.8**
Fisher Merging[NeurIPS22]	✓	82.2	69.2	**88.6**	87.5	93.5	80.6	74.8	93.3	70.0
RegMean[ICLR23]	✓	83.7	73.3	81.8	86.1	97.0	88.0	84.2	98.5	60.8
Task Arithmetic[ICLR23]	✓	84.5	74.1	82.1	86.7	93.8	87.9	86.8	98.9	65.6
TIES-Merging[NeurIPS23]	✓	86.0	76.5	85.0	89.4	95.9	90.3	83.3	99.0	68.8
AdaMerging[ICLR2024]	✓	84.9	75.6	83.4	82.6	89.9	85.1	**96.0**	97.7	69.0
AdaMerging++[ICLR2024]	✓	87.3	77.2	86.3	88.7	94.9	88.5	93.2	98.8	71.3
PCB-Merging[NeurIPS24]	✓	87.5	76.8	86.2	89.4	**96.5**	88.3	91.0	98.6	**73.6**
NeuroMerging (Ours)	✓	**88.3**	**77.3**	87.1	**90.1**	96.1	**91.0**	92.2	**99.4**	73.0

Table 15: Test set performance when merging ViT-L/14 models on eight vision tasks. Please refer to Section 5.4 for experimental details and Section C.4 for error analysis.

Method (↓)	**Average**	cosmos_qa	social_iqa	quail	wic	copa	h-swag
paws	37.2	25.0	37.0	29.9	49.5	57.4	24.5
qasc	36.5	21.2	37.4	29.5	49.8	54.4	26.7
quartz	36.9	24.7	36.9	28.8	48.5	57.4	25.0
story_cloze	36.8	21.9	36.4	25.7	53.6	57.4	26.1
wiki_qa	36.2	25.9	36.3	29.9	51.2	48.5	25.2
winogrande	36.8	23.9	37.9	24.1	51.8	58.8	24.4
wsc	39.5	26.9	38.1	29.5	55.4	61.8	25.2
Pretrained	36.8	22.9	36.4	29.9	50.8	55.9	24.8
Averaging[ICML22]	37.4	**23.7**	36.8	29.3	51.3	58.8	24.8
Fisher Merging[NeurIPS22]	33.8	15.6	21.9	24.9	**65.6**	53.1	21.9
Task Arithmetic[ICLR23]	36.9	19.0	35.6	29.5	54.0	55.9	**27.7**
RegMean[ICLR23]	34.3	23.1	28.1	24.9	48.4	62.5	18.8
TIES-Merging[NeurIPS23]	38.5	21.9	37.4	29.3	52.0	64.7	25.5
PCB-Merging[NeurIPS24]	38.5	22.8	**37.5**	29.1	51.3	63.2	27.0
NeuroMerging (Ours)	**39.2**	21.2	37.3	**29.9**	52.0	**69.1**	25.4

Table 16: Out-of-Distribution performance of T5-Base models checkpoints on six tasks. Please refer to Section 5.2 for experimental details and Section C.4 for error analysis.

Method (↓)	**Average**	cosmos_qa	social_iqa	quail	wic	copa	h-swag
paws	38.2	28.4	37.6	25.4	60.9	51.5	25.2
qasc	37.9	23.1	37.0	25.5	49.0	64.7	28.1
quartz	36.2	26.1	38.0	25.7	51.3	50.0	26.2
story_cloze	37.9	22.9	37.5	24.5	51.2	64.7	26.6
wiki_qa	35.0	23.2	37.4	26.1	51.2	47.1	25.1
winogrande	36.1	25.2	39.6	24.1	51.3	50.0	26.4
wsc	37.2	26.2	38.8	28.8	55.4	48.5	25.8
Pretrained	36.3	23.7	37.8	28.1	51.2	51.5	25.5
Averaging[ICML22]	36.7	25.3	37.0	23.4	51.5	57.4	25.9
Fisher Merging[NeurIPS22]	32.0	**34.4**	25.0	26.1	40.6	56.2	9.4
Task Arithmetic[ICLR23]	39.2	24.6	38.0	**27.3**	58.6	58.8	28.1
RegMean[ICLR23]	36.0	**34.4**	28.1	25.3	**62.5**	50.0	15.6
TIES-Merging[NeurIPS23]	40.1	25.1	**40.8**	23.0	56.3	67.6	27.6
PCB-Merging[NeurIPS24]	40.4	25.6	40.7	25.7	55.1	66.2	**29.3**
NeuroMerging (Ours)	**40.9**	24.7	40.7	26.6	56.4	**69.1**	27.9

Table 17: Out-of-Distribution performance of T5-Large models checkpoints on six tasks. Please refer to Section 5.2 for experimental details and Section C.4 for error analysis.

Task(→)
Method(↓)
	Dataset (↓)	Test Set Performance							

		paws	qasc	quartz	story_cloze	wiki_qa	winogrande	wsc	**Average**
Fine-tuned	paws	94.4	18.0	53.3	53.1	87.9	49.8	54.2	58.7
	qasc	54.3	97.1	55.7	64.7	66.3	49.8	41.7	61.4
	quartz	59.9	65.1	85.3	51.2	72.5	48.9	62.5	63.6
	story_cloze	53.4	31.9	54.2	91.0	57.4	49.1	56.9	56.3
	wiki_qa	55.8	16.3	50.9	53.7	95.7	48.7	63.9	55.0
	winogrande	55.7	50.2	62.4	55.3	78.4	71.6	56.9	61.5
	wsc	55.8	16.3	57.7	48.5	73.3	47.4	80.6	54.2
	Total Avg.	**In-Domain: 88.0**		**Out-Domain: 53.8**		**All: 58.7**			
Orthogonal	paws	94.4	18.1	53.3	53.3	88.1	49.6	56.9	59.1
	qasc	54.4	97.1	55.6	64.8	66.3	50.0	43.1	61.6
	quartz	60.0	65.0	85.3	51.5	72.5	48.5	63.9	63.8
	story_cloze	53.3	31.8	53.8	90.9	57.7	49.0	56.9	56.2
	wiki_qa	55.8	16.3	51.1	53.5	95.7	48.2	63.9	54.9
	winogrande	55.7	49.8	62.2	55.8	78.4	71.7	56.9	61.5
	wsc	55.8	16.6	57.4	48.2	73.4	47.4	80.6	54.2
	Total Avg.	**In-Domain: 88.0**		**Out-Domain: 53.9**		**All: 58.8**			
Parallel	paws	54.4	14.5	53.4	54.2	71.8	49.5	63.9	51.7
	qasc	55.3	14.9	54.0	54.3	71.0	48.6	63.9	51.7
	quartz	55.3	14.3	55.6	54.2	71.7	49.4	63.9	52.1
	story_cloze	55.3	14.3	54.1	53.8	71.1	49.3	63.9	51.7
	wiki_qa	55.6	14.7	54.5	54.3	77.1	49.1	63.9	52.7
	winogrande	55.3	14.8	54.2	53.6	71.8	49.5	63.9	51.9
	wsc	55.3	14.7	53.7	53.7	72.5	49.5	63.9	51.9
	Total Avg.	**In-Domain: 52.7**		**Out-Domain: 51.8**		**All: 51.9**			

Table 18: Test set performance comparison of T5-Large models under different keeping strategies (naive finetuned, keep orthogonal, and keep parallel) across seven NLP tasks. Please refer to Section 1 and 6.2 for experimental details.

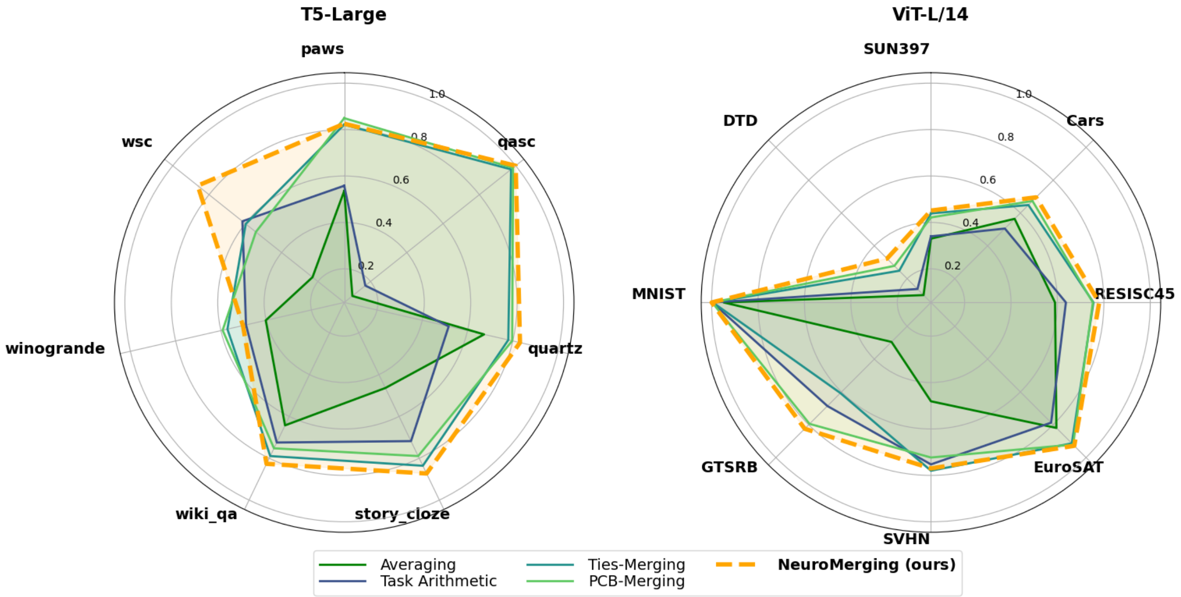

Figure 8: Comparison of merging methods on NLP with T5-Large **(Left)** and CV with ViT-L/14 **(Right)** without validation datasets. NeuroMerging outperformed existing methods in most tasks. Please refer to Section 6.1 for more discussion.