Disentangling Task Interference within Neurons: Model Merging in Alignment with Neuronal Mechanisms

Anonymous ACL submission

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 at various levels: model, layer, or parameter, offer a promising solution. However, task interference remains a fundamental challenge, leading to performance degradation and suboptimal merged models. Existing approaches largely overlook the fundamental role of individual neurons and their connectivity, resulting in a lack of interpretability in both the merging process and the merged models. In this work, we present the first study on the impact of neuronal alignment in model merging. We decompose task-specific representations into two complementary neuronal subspaces that regulate neuron sensitivity and input adaptability. Leveraging this decomposition, we introduce 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 demonstrate that NeuroMerging achieves superior performance compared to existing methods on multi-task benchmarks across both vision and natural language domains. Our findings highlight the importance of aligning neuronal mechanisms in model merging, offering new insights into mitigating task interference and improving knowledge fusion.

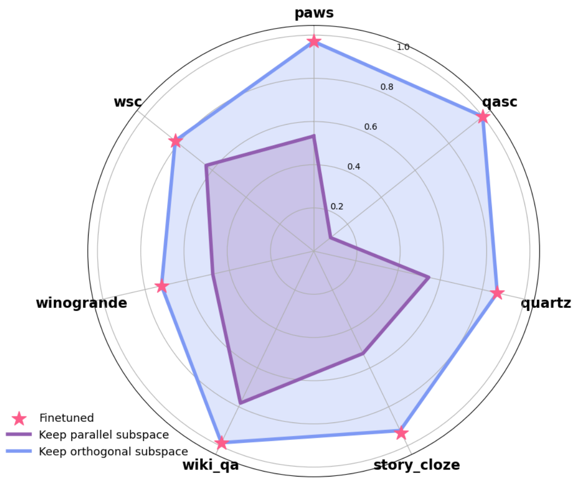

Figure 1: Impacts on neuronal subspaces decomposition of T5-Large. Retaining the orthogonal subspace while removing the parallel subspace preserves near-perfect performance across all tasks. In contrast, keeping the parallel subspace while removing the orthogonal subspace leads to a significant performance drop.

1 Introduction

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), have revolutionized AI by learning 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 specialized 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 interpo-

Table 1: Model merging with different scales and granularity levels.

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**

lation (Wortsman et al., 2022) to combine the strengths of different models and has since evolved into techniques that balance competition and cooperation within shared representation and enable task editing in weight space (Du et al., 2024; Ilharco et al., 2023; Ortiz-Jimenez et al., 2024). In NLP, methods such as merging task-specific language models have been explored to build and update foundation models with multi-task capabilities (Raia, 2021; Wan et al., 2024; Akiba et al., 2025; Wan et al., 2025). 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., 2024). These advancements underscore model merging as a promising avenue for future research.

Existing methods for model merging primarily operate at three granularities—model-level, layer-level, or parameter-level (Ilharco et al., 2023; Yang et al., 2023; Du et al., 2024)—while overlooking the fundamental role of individual neurons and their 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). In Figure 1, we illustrate that modifying model weights along two complementary neuronal subspaces—by removing one and retaining the other—leads to distinct impacts on task performance. 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 task interference at the neuronal level. Specifically, we investigate the role of neuronal alignment in model merging, illustrated in Figure 2. We begin by decomposing task-specific representations into two complementary neuronal subspaces that regulate neuron sensitivity and input adaptability. 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 vision and natural language domains, considering various settings, including in-domain and out-of-domain generalization. Empirically, our method outperforms existing approaches. The main contributions of our paper are as follows:

- We present the first exploration into the impact of neuronal alignment 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 propose NeuroMerging, a new framework developed to reduce task interference in alignment with neuronal mechanism.
- We show that NeuroMerging achieved superior performance compared to existing approaches on multi-task benchmarks in both vision and natural language processing.

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). Optimization-based approaches, on the other hand, focus on balancing task gradients or loss functions to mitigate task conflicts during

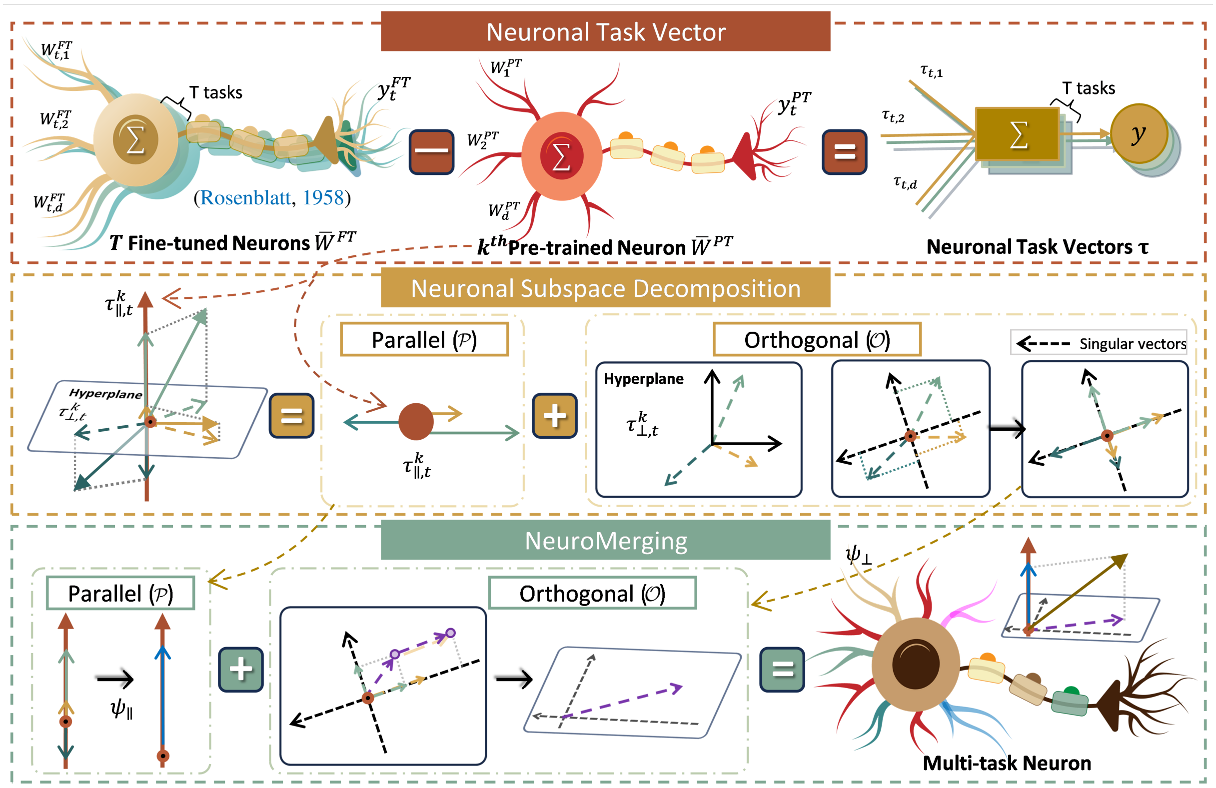

Figure 2: Illustration of our proposed framework. **Neuronal Task Vector:** Neuronal task vectors τ for the k^{th} neuron are defined as the difference between the fine-tuned and pre-trained neuron for each task. **Decomposition:** The pre-trained k^{th} neuron is decomposed into its parallel and orthogonal complementary subspaces, followed by the projection of neuronal task vectors onto these subspaces. **NeuroMerging:** Our proposed *NeuroMerging* operates within these complementary subspaces for neuronal model merging.

training (Bai et al., 2023; Kendall 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., 2020). 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., 2024; 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., 2023). Layer-level merging selectively integrates layers from different models under the assumption of shared representations; for instance, AdaMerging (Yang et al., 2023) 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., 2024), DARE (Le et al., 2024), 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: individual 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 conduct the first in-depth study on neuronal alignment in model merging, aiming to uncover its role in preserving task-specific knowledge and its potential to mitigate task conflict.

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**, showed in Algorithm 1, which performs merging in the neuronal subspaces.

3.1 Preliminaries

In this work, we consider the model merging 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 $\bar{\theta} = \theta_0 + \phi(\tau_1, ..., \tau_t, ..., \tau_T)$, where $\phi(\cdot)$ represents the transformation applied to each task vector τ_t and followed by merging. 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 are set to zero (Yadav et al., 2024). The masked task vector is defined as $\tau_t^{\text{masked}} = m_t \circ \tau_t$, where m_t is the mask that keeps the top $r\%$ of the elements of each task vector. In the following section, we use τ_t to represent the masked task vector for readability.

3.2 Neuronal Task Vector

Zooming in to the neuronal level of τ_t, we define $\mathbf{w}_t^k \in \mathbb{R}^d$ to represent the weight vector of k^{th} neuron in model θ_t. The task-specific adaptation of this neuron, relative to the pre-trained model, is given by the neuronal task vectors:

$$\tau_t^k = \mathbf{w}_t^k - \mathbf{w}_0^k, \tag{1}$$

where $\mathbf{w}_0^k$ corresponds to the weight of the k^{th} neuron in the pre-trained model θ_0. Figure 2 illustrates our proposed neuronal task vector.

3.3 Neuronal Subspace Decomposition

To examine how task-specific modifications impact individual neurons, we decompose the neuronal task vectors into two complementary neuronal subspaces, visualized in Figure 2. Mathematically, this decomposition is formulated as:

$$\tau_t^k = \tau_{\|,t}^k + \tau_{\perp,t}^k, \tag{2}$$

where $\tau_{\|,t}^k = \mathbf{P}\tau_t^k$ projects the neuronal task vectors onto the pre-trained model's weight space, **Parallel Subspace** ($\mathcal{P}$), capturing neuron sensitivity. 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** ($\mathcal{O}$), reflecting task-specific adaptability. 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.

Algorithm 1 NeuroMerging

Input: Task-specific models $\tau_1, \tau_2, \ldots, \tau_T$, pre-trained model θ_0, Mask ratio r
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.`
    $\tau_{\|,t}^k = \mathbf{P}\tau_t^k, \quad \tau_{\perp,t}^k = (\mathbf{I} - \mathbf{P})\tau_t^k$
  end
end
`▷ Merge neuronal task vectors.`
for $k \leftarrow 1$ **to** K **do**
  if *Validation data is available* **then**
    Tune λ_1 and λ_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
  $\bar{\tau}^k = \lambda_1 \psi_\|(\tau_{\|,1}^k, .., \tau_{\|,T}^k) + \lambda_2 \psi_\perp(\tau_{\perp,1}^k, .., \tau_{\perp,T}^k)$
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 $\bar{\theta}$

- **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.

The impact of weight editing in these complementary subspace on task adaption is discussed in Section 6.3.

3.4 NeuroMerging

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

$$\overline{\tau}^k = \lambda_1 \psi_{\parallel}\left(\tau^k_{\parallel,1}, \ldots, \tau^k_{\parallel,t}, \ldots, \tau^k_{\parallel,T}\right) + \lambda_2 \psi_{\perp}\left(\tau^k_{\perp,1}, \ldots, \tau^k_{\perp,t}, \ldots, \tau^k_{\perp,T}\right) \quad (3)$$

where $\psi_{\parallel}(\cdot)$ denotes the merging function for $\tau^k_{\parallel,t} \in \mathbb{R}$, which can be a commonly used weighted average, TIES's disjoint merge (Yadav et al., 2024), and others. $\psi_{\parallel}(\cdot)$ is also applied to non-neuronal parameters such as bias and pre-norm. For $\tau^k_{\perp,t} \in \mathbb{R}^T$, 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 $\tau^k_{\perp,t}$ along each dimension of the SVD subspace, before applying $\psi_{\parallel}(\cdot)$ to them.

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 6.4) 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 σ_t is the ratio of the $L_1 - Norm$ of the zeroed-out elements in the task vector in Section 3.1 to the $L_1 - Norm$ of the original task vector: $\sigma_t = \frac{\|\tau_t^{masked}\|_1}{\|\tau_t\|_1}$. Detailed discussion on the parameters is provided in Section 6.4.

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 $\overline{\theta} = \theta_0 + \overline{\tau}_{rescaled}$.

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-Merging (Yadav et al., 2024), PCB-Merging (Du et al., 2024), and our proposed NeuroMerging method. We report average accuracy across all tasks' test sets as our primary evaluation metric.

Validation Set Availability. Previous works exhibit varying dependencies on a validation set. Fisher Merging inherently requires 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 match the dimensions of the original model, they introduce substantial storage and computational overhead, limiting scalability to larger models and more extensive merging tasks.

Notably, 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 require a masking ratio, we follow the settings of Yadav et al. (2024) 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 is 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 perform 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 search 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.

Table 2: Test set performance when merging T5-Large models on seven NLP tasks.

Task(→) Method(↓)	Validation	Average	Test Set Performance 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**

5 Results

5.1 Merging NLP Models

Following the experimental settings from Yadav et al. (2024), we use the T5-base and T5-large models (Raffel et al., 2020), which are encoder-decoder transformers (Vaswani et al., 2017) pretrained via masked language modeling on a large text corpus, and fine-tune them independently on seven tasks: Khot et al.'s (2019) 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 (2021) Winogrande together with Levesque et al.'s (2012) WSC for Coreference Resolution. Table 2 and 7 demonstrated that our approach has 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 B.2.

5.2 Out-of-Domain Generalization

Building upon the experimental setup of Yadav et al. (2024), 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 3, 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 B.2 Table 10 and 11.

5.3 Merging LLMs

We follow the experimental setup of Du et al. (2024) and extend our NeuroMerging to larger LLMs. Specifically, we merged three domain-

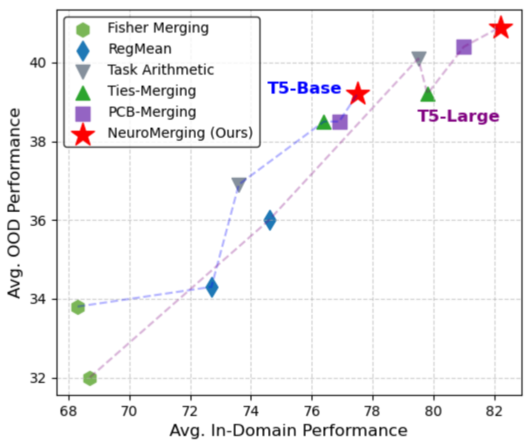

Figure 3: In-domain v.s. Out-domain performance.

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., 2024), and code generation[3] (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 3, our NeuroMerging 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.

Table 3: Performance comparison on LLaMA2.

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
PCB-Merging[NeurIPS24]	36.4	52.3	16.5	35.1
PCB-Merging+ES [NeurIPS24]	36.4	53.1	16.5	35.3
NeuroMerging (Ours)	36.1	**57.2**	14.6	**36.0**

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). During the fine-tuning process, we optimized the visual encoder across eight distinct tasks while maintaining a fixed text encoder configuration. 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., 2019), 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 4 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.

[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

Table 4: Performance comparison on ViT.

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]	✗	<u>75.9</u>	<u>86.9</u>
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
PCB-Merging[NeurIPS24]	✓	<u>76.3</u>	<u>87.5</u>
NeuroMerging (Ours)	✓	**76.5**	**88.3**

6 Additional Results and Analysis

6.1 Merging without Validation Sets

When validation data is unavailable, we examine the parameters λ_1 and λ_2 selected according to Section 3.4, where λ_1 is set to zero, as the corresponding subspace is observed to have little impact in Section 6.4. The value of λ_2 is computed based on the L1-norm of masked and unmasked task vectors. Figure 7 and Table 2 and Appendix Tables 2, 8, and 9 present 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 demonstrates that our proposed method, with a simple rescaling, outperforms 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 outperforms the strongest baseline by 0.5% and 1.0% for ViT-B/32 and ViT-L/14, respectively.

6.2 Ablation Study on Merging Functions

We conducted ablation experiments on various merging functions $\psi(\cdot)$ to evaluate their effective-

Table 5: Comparison of $\psi(\cdot)$ on Avg. Accuracy.

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

ness in combining numerics. As shown in Table 5, 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.

6.3 Role of Neuronal Subspaces

Figure 1 and Appendix Table 12 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.

Table 6: In/Out-domain Performance of T5-large.

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

6.4 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 4, the performance exhibited column-wise uniformity in the heatmap, indicating insensitivity to variations in λ_1, which aligns with our earlier discussion on the role of the orthogonal subspace in Section 6.3. The optimal performance occurred at $\lambda_2 = 3.6$, attributed to the substantial proportion of masked variables.

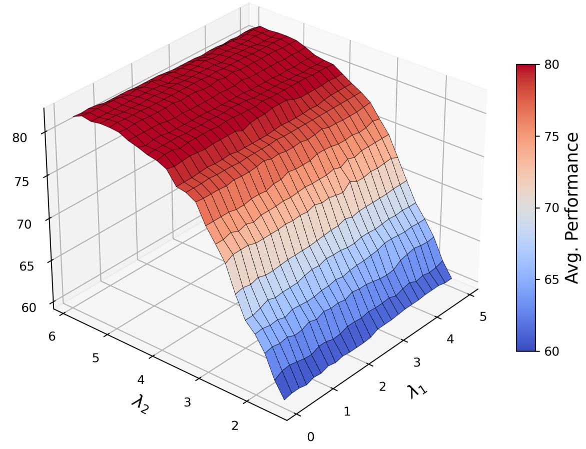

Figure 4: Impacts of λ_1 and λ_2 on T5-Large.

Masking Ratio r. We observed robust performance across different ratios, with accuracy peaks at 15% and 10% for T5-Base and T5-Large, respectively. Performance variations remain bounded (within 2.5% for T5-Base and 4% for T5-Large) and stabilize beyond 35%.

7 Conclusion

In this paper, we revisited model merging from the core principle neuron and its connectivity that underpin the learning process of recent deep neural networks and LLMs. Specifically, we presented the first exploration into the impact of neuronal alignment in the model merging process, by decomposing of task-specific representations into two complementary neuronal subspaces and examining their roles. Based on these insights, we proposed NeuroMerging, a novel framework designed to reduce task interference within neurons. Our empirical evaluations demonstrate that NeuroMerging achieves 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 introduces a neuronal mechanistic perspective on task interference in model merging, it captures only one aspect of neuronal mechanisms, as it does not yet explore interactions between synchronized neuron groups or neuronal dynamics. Further research is needed to investigate these factors and deepen our understanding of neuronal mechanisms in model merging or even multi-task learning.

8 Limitations

While our work provides the first neuronal mechanistic perspective on task interference when merging language models, (1) it remains a partial view of neuronal mechanisms as it does not yet explore interactions between synchronized neuron groups or 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.

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*, pages 1–10.

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. *Advances in Neural Information Processing Systems*, 35:32897–32912.

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*, pages 3606–3613.

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)*. https://openreview.net/forum?id=YicbFdNTTy.

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 *The Thirty-eighth Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Sebastian Dziadzio, Vishaal Udandarao, Karsten Roth, Ameya Prabhu, Zeynep Akata, Samuel Albanie, and Matthias Bethge. 2024. How to merge your multimodal models over time? *arXiv preprint arXiv:2412.06712*.

Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. 2021. Efficiently identifying task groupings for multi-task learning. *Advances in Neural Information Processing Systems*, 34:27503–27516.

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. 2019. 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.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. 2023. Lorahub: Efficient cross-task generalization via dynamic lora composition. *arXiv preprint arXiv:2307.13269*.

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*.

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. *Advances in Neural Information Processing Systems*, 35:29262–29277.

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*.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Alexander Jansen, and Ashish Sabharwal. 2019. QASC: A dataset for question answering via sentence composition. In *AAAI*.

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *International conference on machine learning*, 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*, pages 554–561.

Yu Le, Yu Bowen, Yu Haiyang, Huang Fei, and Li Yongbin. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*.

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

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

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 2024*, pages 11260–11285, Bangkok, Thailand. Association for Computational Linguistics.

Jian Liang, Ziqi Liu, Jiayu Zhou, Xiaoqian Jiang, Changshui Zhang, and Fei Wang. 2020. Model-protected multi-task learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(2):1002–1019.

Shikun Liu, Edward Johns, and Andrew J Davison. 2019. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1871–1880.

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

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 2011*.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. 2024. Task arithmetic in the tangent space: Improved editing of pre-trained models. *Advances in Neural Information Processing Systems*, 36.

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 *International Conference on Machine Learning (ICML)*. https://arxiv.org/abs/2103.00020.

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*, 21(140):1–67.

Jonathan Raia. 2021. Model merge and its methods. *Medium*. Accessed: 2025-02-11.

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 AAAI Conference on Artificial Intelligence*, 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. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

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*.

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.

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 (Volume 2: Short Papers)*, 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. *arXiv preprint arXiv:1701.06538*.

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*, 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 WH 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. IEEE.

Yi-Lin Sung, Linjie Li, Kevin Lin, Zhe Gan, Mohit Bansal, and Lijuan Wang. 2023. An empirical study of multimodal model merging. *arXiv preprint arXiv:2304.14933*.

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*, volume 30. Curran Associates, Inc.

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

Fanqi Wan, Longguang Zhong, Ziyi Yang, Ruijun Chen, and Xiaojun Quan. 2025. 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 Pour Ansari. 2024. Merge vision foundation models via multi-task distillation.

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*, 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*, 119(1):3–22.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. 2024. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. 2023. Adamerging: Adaptive model merging for multi-task learning. *arXiv preprint arXiv:2310.02575*.

Yi Yang, Scott Wen-tau Yih, and Chris Meek. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. ACL - Association for Computational Linguistics.

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

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*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase Adversaries from Word Scrambling. In *Proc. of NAACL*.

A Statistical Analysis of Parameters

From the statistical analysis of neuronal versus non-neuronal parameters, it is observed that neuronal parameters dominate the T5-Base and T5-Large, showed in Figure. 5. 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.

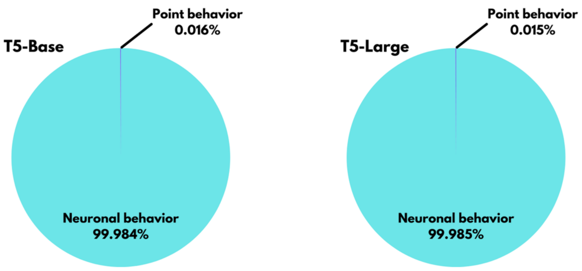

Figure 5: Statistical analysis of neuronal versus non-neuronal parameters.

B Additional Results

B.1 Mask Ratio vs. Performance

When validation is available, we analyze the impact of the mask ratio on performance, showed in Figure 6. Maintaining r at 10–20% improves performance, whereas exceeding this range often leads to degradation. This suggests that an optimal masking ratio balances information retention and redundancy reduction.

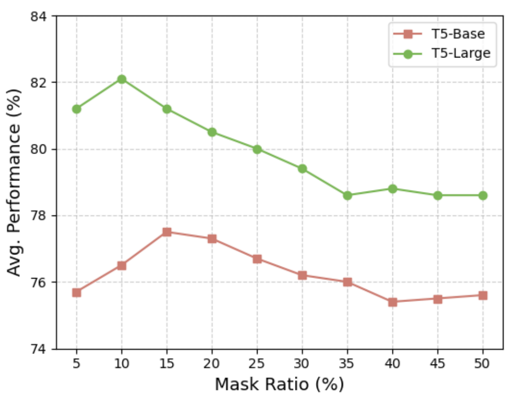

Figure 6: Impacts on mask ratio r.

B.2 Comprehensive Task-Level Results

We provide 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 7, 8, 9. The task-level results of the out-domain experiments for T5-Base and T5-Large can be found in Tables 10 and 11. Lastly, Table 12 shows the task-level results from Section 6.3 when only one of the neuronal subspace is retained.

C Implementation Details

We executed all our experiments on Nvidia A6000 GPUs equipped with 48GB RAM. 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.

D 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., 2021) and QASC (Khot et al., 2019) 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., 2019) 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.

Table 7: Test set performance when merging T5-Base models on seven NLP tasks. Please refer to Section 5.1 for experimental details.

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]	✗	<u>73.6</u>	<u>82.4</u>	<u>94.1</u>	<u>71.9</u>	66.0	<u>91.2</u>	**51.5**	<u>58.3</u>
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	<u>67.9</u>	**94.8**	<u>51.0</u>	**62.5**
Fisher Merging[NeurIPS22]	✓	68.3	66.7	85.6	63.5	57.1	90.1	<u>54.2</u>	<u>60.8</u>
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]	✓	<u>76.9</u>	<u>88.2</u>	<u>95.2</u>	<u>71.0</u>	**77.3**	<u>95.1</u>	51.9	59.7
NeuroMerging (Ours)	✓	**77.5**	87.5	**95.7**	68.2	<u>76.8</u>	94.5	51.8	**68.1**

Table 8: Test set performance when merging ViT-B/32 models on eight vision tasks. Please refer to Section 5.4 for experimental details.

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	<u>65.3</u>	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	<u>98.3</u>	54.2
PCB-Merging[NeurIPS24]	✗	<u>75.9</u>	**65.8**	**64.4**	**78.1**	<u>81.1</u>	<u>84.9</u>	<u>77.1</u>	98.0	<u>58.4</u>
NeuroMerging (Ours)	✗	**76.4**	64.7	<u>64.2</u>	<u>77.0</u>	**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
PCB-Merging[NeurIPS24]	✓	<u>76.3</u>	<u>66.7</u>	65.5	**78.5**	<u>79.3</u>	**86.4**	<u>77.1</u>	<u>98.2</u>	<u>59.1</u>
NeuroMerging (Ours)	✓	**76.5**	65.3	<u>65.7</u>	<u>77.1</u>	**84.8**	<u>84.5</u>	**77.9**	**98.3**	58.5

Table 9: Test set performance when merging ViT-L/14 models on eight vision tasks. Please refer to Section 5.4 for experimental details.

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	<u>99.1</u>	64.3
TIES-Merging[NeurIPS23]	✗	86.0	<u>76.5</u>	85.0	<u>89.3</u>	95.7	**90.3**	83.3	99.0	68.8
PCB-Merging[NeurIPS24]	✗	<u>86.9</u>	75.8	<u>86.0</u>	89.2	<u>96.0</u>	88.0	<u>90.9</u>	<u>99.1</u>	<u>70.0</u>
NeuroMerging (Ours)	✗	**87.9**	**77.0**	**86.9**	**90.3**	**96.3**	<u>89.9</u>	**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	<u>70.0</u>
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	<u>89.4</u>	95.9	<u>90.3</u>	83.3	<u>99.0</u>	68.8
PCB-Merging[NeurIPS24]	✓	<u>87.5</u>	<u>76.8</u>	86.2	<u>89.4</u>	**96.5**	88.3	<u>91.0</u>	98.6	**73.6**
NeuroMerging (Ours)	✓	**88.3**	**77.3**	<u>87.1</u>	**90.1**	<u>96.1</u>	**91.0**	**92.2**	**99.4**	<u>73.0</u>

Table 10: Out-of-Distribution performance of T5-Base models checkpoints on six tasks. Please refer to Section 5.2 for experimental details.

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	<u>29.5</u>	<u>54.0</u>	55.9	**27.7**
RegMean[ICLR23]	34.3	<u>23.1</u>	28.1	24.9	48.4	62.5	18.8
TIES-Merging[NeurIPS23]	<u>38.5</u>	21.9	<u>37.4</u>	29.3	52.0	<u>64.7</u>	<u>25.5</u>
PCB-Merging[NeurIPS24]	<u>38.5</u>	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 11: Out-of-Distribution performance of T5-Large models checkpoints on six tasks. Please refer to Section 5.2 for experimental details.

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**	<u>58.6</u>	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	<u>67.6</u>	27.6
PCB-Merging[NeurIPS24]	<u>40.4</u>	<u>25.6</u>	<u>40.7</u>	25.7	55.1	66.2	**29.3**
NeuroMerging (Ours)	**40.9**	24.7	<u>40.7</u>	26.6	56.4	**69.1**	<u>27.9</u>

Table 12: 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 6.3 for experimental details.

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**			

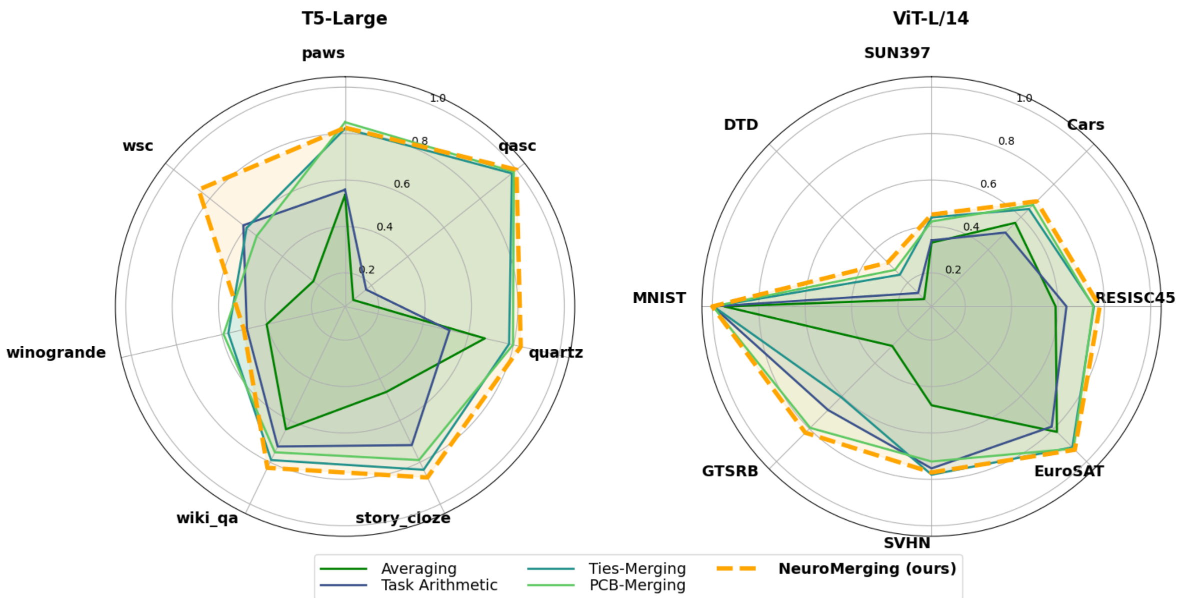

Figure 7: Comparison of merging methods on NLP with T5-Large **(Left)** and CV with ViT-L/14 **(Right)** without validation datasets. NeuroMerging outperforms existing methods in most tasks. Please refer to Section 6.1 for more discussion.