Locate-then-Merge: Neuron-Level Parameter Fusion for Mitigating Catastrophic Forgetting in Multimodal LLMs

Zeping Yu **Sophia Ananiadou**
Department of Computer Science, National Centre for Text Mining
The University of Manchester
{zeping.yu@postgrad. sophia.ananiadou@}manchester.ac.uk

Abstract

Although multimodal large language models (MLLMs) have achieved impressive performance, the multimodal instruction tuning stage often causes catastrophic forgetting of the base LLM's language ability, even in strong models like Llama3. To address this, we propose Locate-then-Merge, a training-free parameter fusion framework that first locates important parameters and then selectively merges them. We further introduce Neuron-Fusion, a neuron-level strategy that preserves the influence of neurons with large parameter shifts—neurons likely responsible for newly acquired visual capabilities—while attenuating the influence of neurons with smaller changes that likely encode general-purpose language skills. This design enables better retention of visual adaptation while mitigating language degradation. Experiments on 13 benchmarks across both language and visual tasks show that Neuron-Fusion consistently outperforms existing model merging methods. Further analysis reveals that our method effectively reduces context hallucination in generation.

1 Introduction

Multimodal large language models (MLLMs) (Liu et al., 2023; Anil et al., 2023; Chen et al., 2024; Wu et al., 2024; Hurst et al., 2024; Bai et al., 2025) have advanced rapidly by adapting pretrained large language models (LLMs) (Brown et al., 2020; Ouyang et al., 2022; Yang et al., 2024a; Grattafiori et al., 2024) through multimodal instruction tuning. Among various modalities, vision has received the most attention and become the primary focus for enhancing multimodal LLMs (Liang et al., 2024; Li et al., 2025). By introducing vision-language connectors and training with image-text pairs, MLLMs have demonstrated impressive performance on vision-language tasks such as visual question answering (Antol et al., 2015) and visual reasoning (Hudson and Manning, 2019).

However, recent studies (Ratzlaff et al., 2024; Zhang et al., 2024) find that although visual instruction tuning can obtain visual capabilities, it often severely degrades the original general language abilities of the base LLMs, a phenomenon known as catastrophic forgetting (Goodfellow et al., 2013; Kirkpatrick et al., 2017; Kemker et al., 2018). Particularly on complex language understanding and reasoning benchmarks such as ARC-Challenge (Clark et al., 2018) and GSM8K (Cobbe et al., 2021), finetuned MLLMs perform significantly worse than their original LLMs. Alarmingly, even strong open-source models like Llama3 suffer from this degradation, limiting the generalization and practical deployment of MLLMs across diverse and complicated language tasks.

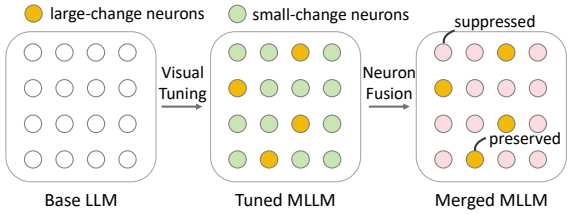

Figure 1: Neuron-Fusion in MLLMs. After visual tuning, some neurons exhibit larger changes than others. Neuron-Fusion selectively preserves neurons with significant parameter changes while suppressing those with smaller changes. This targeted fusion enables the model to retain newly acquired visual capabilities while minimally affecting its general language abilities.

Although catastrophic forgetting in MLLMs poses a significant challenge, systematic studies addressing this issue remain limited. Recent years have witnessed the emergence of several novel model merging techniques (Wortsman et al., 2022; Ilharco et al., 2022; Yadav et al., 2023; Yu et al., 2024), demonstrating the potential of parameter fusion to alleviate catastrophic forgetting. However, as an emerging research area, model merging still lacks a systematic framework to guide the design and evaluation of effective methods. Moreover, ex-

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 7065–7078
November 4-9, 2025

isting methods were primarily developed for single-modal LLMs, and their application to MLLMs remains largely unexplored. It is unclear whether these methods can effectively mitigate catastrophic forgetting in MLLMs. This highlights the urgent need for targeted solutions that can recover language abilities while preserving visual adaptation.

In this paper, we propose Locate-then-Merge, a general framework for training-free parameter fusion that decouples the process into two stages: locating important parameters and selectively merging them. Within this framework, we further develop Neuron-Fusion, a neuron-level fusion method designed to mitigate catastrophic forgetting, as illustrated in Figure 1. Our approach is motivated by an intuitive hypothesis: neurons with large changes during visual instruction tuning are likely to store newly acquired visual capabilities, whereas widespread but small changes across many neurons can cumulatively cause catastrophic forgetting in language ability. Therefore, Neuron-Fusion preserves the contributions of large-change neurons while suppressing the influence of small-change neurons. This design aims to retain the general language abilities of the base LLM while maintaining its acquired visual skills. Empirical results demonstrate that Neuron-Fusion surpasses five state-of-the-art model merging methods across 13 language and vision benchmarks, evaluated on two leading open-source MLLMs. Furthermore, generation analysis shows that Neuron-Fusion effectively reduces context hallucination, enhancing output quality and controllability.

Our contributions are summarized as follows:

a) We systematically validate that visual instruction tuning induces catastrophic forgetting even in powerful MLLMs, as evidenced by significant performance degradation on multiple language understanding and reasoning benchmarks, indicating that catastrophic forgetting remains a widespread and persistent challenge in MLLMs.

b) We propose Locate-then-Merge, a general parameter fusion framework that unifies existing merging strategies: locating important parameters and merging them. Within this framework, we introduce Neuron-Fusion, a neuron-level selection and fusion method that identifies a small subset of neurons with significant parameter changes as key paths for visual adaptation, which is useful for mitigating catastrophic forgetting in MLLMs.

c) We conduct extensive experiments across 13 benchmarks and two powerful open-source MLLMs to validate our approach. Our method is compared against several state-of-the-art model merging techniques, consistently demonstrating superior performance in mitigating catastrophic forgetting while preserving visual adaptation.

2 Related Work

2.1 Multimodal Large Language Models

A typical MLLM consists of three components: a modality encoder, a modality connector, and a pre-trained LLM. In this work, we focus on the vision modality. First, a pre-trained vision encoder such as CLIP (Radford et al., 2021) is employed to transform images into visual embeddings. Second, a vision-language connector, such as a lightweight two-layer MLP (Liu et al., 2023) or cross-attention layers (Alayrac et al., 2022), maps these embeddings into the feature space of the LLM. Finally, the pre-trained LLM, such as Llama3 (Grattafiori et al., 2024), generates output tokens conditioned on both the textual and visual inputs. During the multimodal instruction tuning phase, the parameters of both the connector and the LLM are jointly finetuned to adapt to vision-language tasks.

2.2 Catastrophic Forgetting and Model Merging

Catastrophic forgetting (Goodfellow et al., 2013) refers to the phenomenon in which a machine learning model loses previously acquired knowledge when learning new capabilities. Luo et al. (2023) observe that catastrophic forgetting frequently occurs in LLMs during finetuning, and that larger models tend to forget even more information. Zhu et al. (2024) further investigate this phenomenon in the context of visual tasks. In addition, Zhang et al. (2024) and Ratzlaff et al. (2024) report that visual instruction tuning not only improves multimodal capabilities but also significantly impairs the general language abilities of the underlying LLMs. Feng et al. (2024a) introduce KIF, a framework that identifies and fuses task-specific and shared "skill units" via group-wise importance estimation and fine-grained fusion to both preserve prior knowledge and promote cross-task transfer during continual learning of language models. Feng et al. (2024b) mitigate catastrophic forgetting in continual dialog state tracking by localizing task-specific and shared parameters and consolidating them with fine-grained merging strategies. Ven et al. (2024) hypothesize that catastrophic forgetting arises be-

cause the parameters learned on new tasks deviate substantially from the optimal parameters for previous tasks.

Model merging has emerged as an effective technique for combining the parameters of different models to construct a universal model, without requiring access to the training data (Yang et al., 2024b). Alexandrov et al. (2024) demonstrate that model merging can be beneficial for mitigating catastrophic forgetting. Recent works (Wortsman et al., 2022; Ilharco et al., 2022; Yadav et al., 2023; Davari and Belilovsky, 2024; Yu et al., 2024; Deep et al., 2024) have shown that various merging strategies can significantly improve performance across different tasks and models.

2.3 Neuron-Level Ability Storage in LLMs

Geva et al. (2020) observe that in transformer models, the columns and rows of the two MLPs in Feed-Forward Network (FFN) layers can be interpreted as key and value memories, respectively. Similarly, Yu and Ananiadou (2023) find that the attention value-output matrices, which are implemented as two MLPs, can also be understood in terms of neuron representations. Dai et al. (2021) show that factual knowledge is primarily encoded in FFN neurons. Geva et al. (2022) and Lee et al. (2024) demonstrate that the generation of toxic language can be controlled by editing targeted neurons. Furthermore, Nikankin et al. (2024) and Yu and Ananiadou (2024) find that arithmetic abilities are localized to a small subset of neurons. Schwettmann et al. (2023) discover the "multimodal neurons" in pretrained text-only transformers. Yu et al. (2025) creates an interpretability tool, "logit flow", to analyze the neuron-level information flow in LLMs.

3 Background and Problem Formulation

3.1 Architectures of LLM and MLLM

The architectures of LLM and MLLM are illustrated in Figure 2. In this study, we focus primarily on the vision modality, although the methods can be similarly applied to other modalities (Liang et al., 2024). In the original LLM architecture, the base LLM processes textual inputs and generates corresponding textual outputs. In contrast, the MLLM architecture incorporates an additional vision encoder, which transforms images into visual embeddings. These embeddings are then mapped into the feature space of a tuned LLM via a vision-language connector. Finally, the tuned LLM takes both the textual inputs and the mapped visual embeddings as inputs, producing textual outputs. It is important to note that the tuned LLM's parameters are obtained by performing visual instruction tuning on the base LLM using image-text pairs.

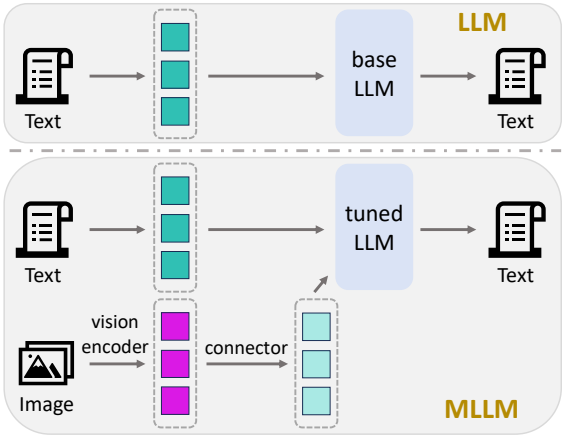

Figure 2: The structures of LLM and MLLM.

3.2 Problem Formulation

Formally, let LLM_{base} denote the base LLM with parameters θ_{base} and language ability L_{base}. Similarly, let LLM_{tuned} denote the tuned LLM in the MLLM, with parameters θ_{tuned}. We define the parameter delta as $\Delta = \theta_{\text{tuned}} - \theta_{\text{base}}$, representing the changes introduced during visual instruction tuning. After tuning, the MLLM acquires a visual ability V_{tuned}, but its language ability degrades to L_{tuned} (typically $L_{\text{tuned}} < L_{\text{base}}$) due to catastrophic forgetting. This degradation occurs because some parameters responsible for language capabilities are inadvertently modified during visual instruction tuning. A straightforward way to recover language ability is to replace LLM_{tuned} with LLM_{base} in the MLLM, while keeping the vision encoder and the vision-language connector unchanged—a process we refer to as "hard-merge". However, hard-merge often severely damages the visual ability, where V_{base} becomes significantly lower than V_{tuned}.

Our goal is to obtain a merged LLM, denoted as LLM_{merge}, with parameters θ_{merge} and language ability L_{merge}. By replacing LLM_{tuned} with LLM_{merge} in the MLLM while keeping the vision encoder and connector unchanged, we obtain a visual ability V_{merge}. Ideally, we aim for L_{merge} to be slightly smaller than, or even comparable to, L_{base}, and for V_{merge} to be slightly smaller than, or comparable to, V_{tuned}. In this desirable scenario, both the language and visual abilities are preserved.

4 Methodology

In order to solve the catastrophic forgetting problem, we aim to utilize model merging methods to obtain LLM_{merge} based on LLM_{base} and LLM_{tuned}. We first propose the Locate-then-Merge framework in Section 4.1. Then we introduce the neuron-level motivation and analysis in Section 4.2, and the Neuron-Fusion method in Section 4.3.

4.1 Locate-then-Merge Framework

Model merging is an emerging research area that aims to combine the parameters of different models to achieve better performance. Early studies on model merging originate from model soups (Wortsman et al., 2022), which demonstrate that simply taking a weighted average of parameters from different models can improve overall performance:

$$\theta_{\text{merge}} = (1-\alpha)\theta_{\text{base}} + \alpha\theta_{\text{tuned}}$$

This approach can also be interpreted from the perspective of Task Arithmetic (Ilharco et al., 2022), where the merged model is expressed as:

$$\theta_{\text{merge}} = \theta_{\text{base}} + \alpha\Delta$$

with $\Delta = \theta_{\text{tuned}} - \theta_{\text{base}}$ representing the task vector that captures the parameter change. Building upon the task vector formulation, we propose the Locate-then-Merge framework for model merging:

$$\theta_{\text{merge}} = \theta_{\text{base}} + F(\text{Sub}(\Delta)) \quad (1)$$

where $\text{Sub}(\cdot)$ is a function that locates a subset of parameters from Δ, and $F(\cdot)$ is a function that transforms and merges the located parameters into the base model. The core intuition behind the Locate-then-Merge framework is inspired by the Lottery Ticket Hypothesis (Frankle and Carbin, 2018), suggesting that a sparse subnetwork within a dense model can perform comparably to the full model. Similarly, we hypothesize that a carefully selected subset of parameter changes can effectively preserve the newly acquired capabilities.

The state-of-the-art model merging approaches can be viewed as special cases within the Locate-then-Merge framework. As summarized in Table 1, each method corresponds to a specific instantiation of the $\text{Sub}(\cdot)$ and $\text{F}(\cdot)$ functions, depending on how parameters are located and merged. The primary differences among these methods lie in the locating stage. TIES (Yadav et al., 2023) trims the task vectors based on their magnitudes and resolves sign conflicts to elect the final sign for each parameter. Breadcrumbs (Davari and Belilovsky, 2024) applies a sparse mask by removing parameters from the extreme tails of the absolute magnitude distribution. DARE (Yu et al., 2024) randomly drops 99% of the parameters. DELLA (Deep et al., 2024) assigns higher dropout probabilities to parameters with lower magnitudes. At the merging stage, the main distinction is whether or not to rescale the selected parameters before combining them.

Method	Locate: Sub(x)	Merge: F(x)
Task Ari	x	αx
TIES	$\text{TRIMDrop}(x)$	αx
Bread	$\text{TailDrop}(x)$	αx
DARE	$\text{RandomDrop}(x)$	$\text{Rescale}(\alpha x)$
DELLA	$\text{MagDrop}(x)$	$\text{Rescale}(\alpha x)$

Table 1: Mapping existing model merging methods into the Locate-then-Merge framework, categorized by locating (Sub) and merging (F) strategies.

We note that a concurrent work, KIF (Feng et al., 2025), also demonstrates that identifying task-specific parameters can mitigate catastrophic forgetting. Unlike KIF, which is designed for continual learning and requires additional training on new tasks, our framework is training-free and directly merges existing models without any finetuning. Furthermore, our approach can be naturally integrated with recent advances in mechanistic interpretability: techniques for attributing knowledge to specific neurons, heads, or layers provide complementary tools for the "locate" stage, enabling more principled identification of parameters to preserve or merge and offering a concrete application scenario where interpretability analyses directly inform model improvement.

4.2 Neuron-Level Motivation and Analysis

Motivation for neuron-level fusion. The existing methods primarily develop strategies to locate important parameters based on their individual magnitudes. However, they do not explicitly consider the structural roles of neurons. Recent studies (Dai et al., 2021; Geva et al., 2022; Schwettmann et al., 2023; Nikankin et al., 2024) have shown that neurons serve as fundamental units that encode distinct capabilities. It is therefore highly plausible that certain neurons are disproportionately important during the fine-tuning process. Motivated by this insight, our work focuses on locating important neurons and designing neuron-level merging

strategies to better preserve acquired capabilities while mitigating catastrophic forgetting.

Neuron change in FFN and attention layers. We analyze the changes of FFN neurons and attention neurons between Llava-Next-Llama3 (tuned MLLM) (Liu et al., 2024a) and Llama3 (base LLM) (Grattafiori et al., 2024). In FFN layers, the k-th neuron corresponds to the k-th column in the down-projection matrix, as well as the k-th row in the up-projection and gate-projection matrices within the SwiGLU activation (Shazeer, 2020). In attention layers, the k-th neuron represents the k-th column in the output matrix, as well as the k-th row in the query, key, and value matrices. For each neuron, we quantify its change by summing the absolute differences of its parameters across all dimensions:

$$\mathrm{C}(i) = \sum_{j} |\Delta_{i,j}| \tag{2}$$

where j indexes all dimensions (weights) associated with neuron i. $\Delta_{i,j}$ represents the parameter change on dimension j in neuron i. The change of neurons in FFN and attention layers are shown in Figure 3 and 4. The x-axis is the neuron index, and the y-axis is the change score of the neurons.

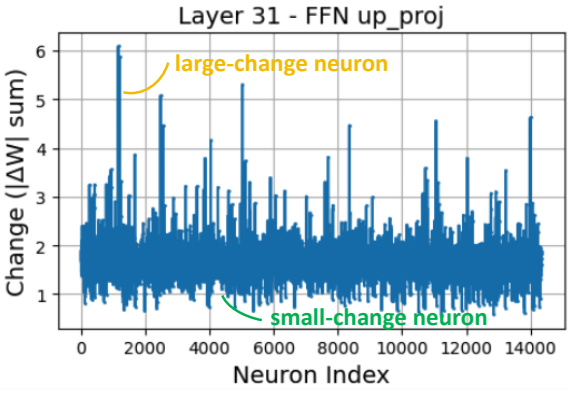

Figure 3: Change of neurons in FFN up matrix.

In Figure 3, we present the change of each neuron in the up-projection matrix in layer 31 as a representative example. Similarly, in Figure 4, we show the change of each neuron in the query matrix in layer 1. Similar trends are observed across other modules and layers. From the observation, we conclude that: **a small number of neurons exhibit significantly larger changes compared to the majority of neurons**.

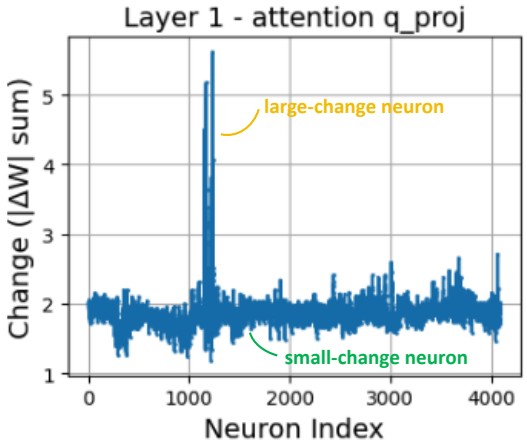

Figure 4: Change of neurons in attention query matrix.

4.3 Neuron-Fusion Method for Mitigating Catastrophic Forgetting

To mitigate catastrophic forgetting, we propose Neuron-Fusion, a targeted neuron-level merging strategy. This method is based on a key hypothesis: neurons with large parameter changes during visual instruction tuning encode newly acquired visual capabilities, whereas widespread small changes may disrupt previously learned language abilities. Accordingly, Neuron-Fusion selectively preserves large-change neurons while suppressing small-change ones, as illustrated in Figure 1. Our approach consists of three steps:

(1) **Neuron-Locate**: We compute the change score $C(i)$ of each neuron by aggregating the absolute differences of its associated parameters (Eq. 2). Then we select the top $M\%$ neurons with the highest scores as candidates for preservation.

(2) **Neuron-Suppress**: To attenuate the effect of small-change neurons, we apply a parameter-level sparsification technique that retains only $K\%$ of parameters within each module. These retained parameters are scattered across neurons, which collectively reduces the influence of widespread parameter changes. This suppression step is agnostic to the specific sparsification method and can be implemented using TIES, Breadcrumbs, or other magnitude-based techniques.

(3) **Neuron-Restore**: To restore the contributions of the previously identified large-change neurons, we introduce two restoration strategies:

(a) Neuron-Replace: Directly reinstates the parameters of the selected neurons from the tuned

model:

$$\Delta'_{i,j} = \begin{cases} \Delta_{i,j}, & \text{if } |\Delta_{i,j}| \text{ is in } K\% \\ \Delta_{i,j}, & \text{if neuron } i \text{ is in } M\% \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

(b) Neuron-Rescale: Adjusts the remaining parameters of each preserved neuron to match its original change score:

$$\Delta'_{i,j} = \begin{cases} 0, & \text{if } |\Delta_{i,j}| \text{ is not in } K\% \\ \Delta_{i,j} \times \frac{C(i)}{C'(i)}, & \text{if neuron } i \text{ is in } M\% \\ \Delta_{i,j}, & \text{otherwise} \end{cases} \quad (4)$$

where $C'(i)$ is the new change score after suppression. The merged model's parameters are calculated by:

$$\theta_{merge} = \theta_{base} + \Delta' \quad (5)$$

In the Locate-then-Merge framework (Eq. 1), Neuron-Suppress corresponds to the locating function Sub(·), while Neuron-Restore (either Replace or Rescale) serves as the merging function F(·).

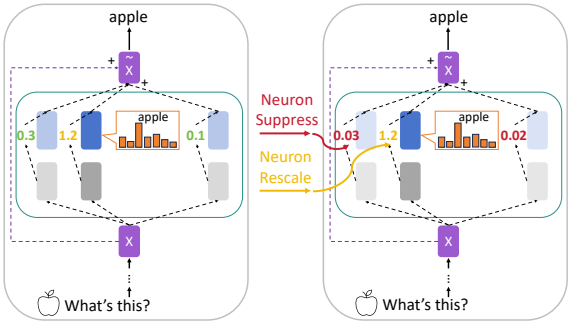

Figure 5: Change of coefficients after Neuron-Fusion.

Key insight: restoring important neurons' coefficients via Neuron-Restore. A key insight of Neuron-Restore comes from the lens of mechanistic interpretability, particularly drawing on the key-value memory view of FFN layers. Following Geva et al. (2020), the two MLPs in a transformer FFN layer correspond to a set of subkeys (first MLP rows) and subvalues (second MLP columns). The FFN output is a weighted sum of subvalues, where the weights—called coefficients—are determined by the inner product between the input vector and each subkey. Furthermore, Geva et al. (2022) show that the distribution over final prediction is influenced directly by these subvalues. As illustrated in Figure 5 (left), the second neuron contributes most strongly to predicting "apple". Its large coefficient (1.2) amplifies the corresponding subvalue, shifting the output distribution toward the correct label.

Now consider how Neuron-Fusion alters this mechanism, shown in Figure 5 (right). Let $x \in \mathbb{R}^D$ be the FFN input and $\Delta_i = [\Delta_{i,1}, \ldots, \Delta_{i,D}]$ be the parameter changes of the i-th subkey. The original coefficient c_i is computed as:

$$c_i = \sum_{j=1}^{D} x_j \Delta_{i,j}$$

When we apply the Neuron-Suppress stage, only $K\%$ dimensions of Δ_i are retained, reducing the coefficient to $\tilde{c}_i$:

$$\tilde{c}_i = \sum_{j \in S} x_j \Delta_{i,j}, \quad S \subset \{1, \ldots, D\}$$

This approximation weakens the contribution of important neurons and may reduce the probability of correct answers. To counteract this, the Neuron-Replace step can replace $\Delta_{i,j}$ into the important neurons and restore c_i. The Neuron-Rescale step can enlarge the surviving parameters in each important neuron to restore the total change score, where the new coefficient $\hat{c}_i$ becomes:

$$\hat{c}_i = \frac{\sum_{j=1}^{D} |\Delta_{i,j}|}{\sum_{j \in S} |\Delta_{i,j}|} \cdot \tilde{c}_i$$

If the input vector x has roughly uniform values, this rescaling approximately restores $\hat{c}_i \approx c_i$, thus preserving the neuron's influence.

5 Experiments

5.1 Experimental Settings

Language ability evaluation. We choose seven widely used benchmark datasets—RACE (Lai et al., 2017), CommonsenseQA (Talmor et al., 2018), PIQA (Bisk et al., 2020), OpenbookQA (Mihaylov et al., 2018), GSM8K (Cobbe et al., 2021), ARC-Easy, and ARC-Challenge (Clark et al., 2018)—to evaluate the general language ability of the model. These datasets cover a broad range of language understanding tasks, including reading comprehension, commonsense reasoning, mathematical problem solving, and general science question answering. Together, they provide a comprehensive assessment of a model's language capabilities. We use 8-shot chain-of-thought for GSM8K, and zero-shot for other datasets. We use the lm-evaluation-harness (Gao et al., 2024) library to calculate the exact-match accuracy for all the datasets.

Visual ability evaluation. We choose six common benchmarks—MME (Fu et al., 2023), MMMU (Yue et al., 2024), ScienceQA (Lu et al., 2022), GQA (Hudson and Manning, 2019), MMBench-CN, and MMBench-EN (Liu et al., 2024b)—to evaluate the visual ability of the model. These datasets cover a wide range of vision understanding skills, including fine-grained perception, subject-specific reasoning across science, math, and humanities, vision science question answering, and visual reasoning. We use the lmms-eval (Li et al., 2024) library to calculate the exact-match accuracy for these datasets.

Models. Model merging methods require access to the parameters of LLMs and MLLMs. So we choose two powerful open-source LLMs—Mistral-7B (Jiang, 2024) and Llama3-8B (Grattafiori et al., 2024)—as our base models. The corresponding MLLMs are obtained by performing visual instruction tuning (Liu et al., 2024a) on vision-language datasets. We use MergeKit (Goddard et al., 2024) library to merge the models.

Method	LA (%)	VA (%)	OA (%)
MLLM	57.14	64.76	60.95
LLM	62.91	48.63	55.77
Task Ari	61.39	63.90	62.65
TIES	62.43	62.19	62.31
Breadcrumbs	60.20	64.60	62.40
DARE	56.91	64.93	60.92
DELLA	55.60	64.12	59.86
Neu-P-TaskA	61.89	63.90	**62.90**
Neu-P-TIES	61.80	63.27	62.54
Neu-S-TIES	61.2	63.5	62.35
Neu-P-Bread	62.00	63.53	62.77
Neu-S-Bread	61.74	63.40	62.57

Table 2: Comparison of Neuron-Fusion and other model merging methods on Llama3. We report LA (language ability), VA (visual ability), and OA (overall ability).

5.2 Results of Neuron-Fusion Method

Comparative Performance. We evaluate Neuron-Fusion against several state-of-the-art model merging methods discussed in Section 4.1, including Task Arithmetic, TIES, Breadcrumbs, DARE, and DELLA. For Neuron-Fusion, we use Task Arithmetic, TIES, and Breadcrumbs as the Neuron-Suppress strategy and integrate them with either Neuron-Replace or Neuron-Rescale during the Neuron-Restore phase. These configurations are denoted as Neu-P-* and Neu-S-*, respectively.

Method	LA (%)	VA (%)	OA (%)
MLLM	55.40	63.07	59.23
LLM	56.89	32.53	44.71
Task Ari	56.86	63.16	60.01
TIES	57.60	62.57	60.08
Breadcrumbs	57.46	62.53	59.99
DARE	55.74	63.13	59.43
DELLA	53.63	62.27	57.95
Neu-P-TaskA	57.80	62.73	**60.27**
Neu-P-TIES	57.34	62.43	59.89
Neu-S-TIES	56.85	62.67	59.76
Neu-P-Bread	57.09	63.07	60.08
Neu-S-Bread	57.31	61.86	59.59

Table 3: Comparison of Neuron-Fusion and other model merging methods on Mistral. We report LA (language ability), VA (visual ability), and OA (overall ability).

Table 2 and Table 3 summarize the results on Llama3 and Mistral models, and the detailed results are shown in Appendix A. Neuron-Fusion consistently achieves the best overall ability (OA), demonstrating its effectiveness in simultaneously preserving language ability (LA) and visual ability (VA). Notably, Neu-P-TaskA and Neu-P-Bread achieve the highest scores on both models, showing that restoring high-impact neurons yields the most balanced performance. In contrast, DARE and DELLA exhibit inferior performance, with LA scores even lower than the base LLMs. This suggests that indiscriminate parameter dropout or excessive rescaling can severely disrupt learned representations.

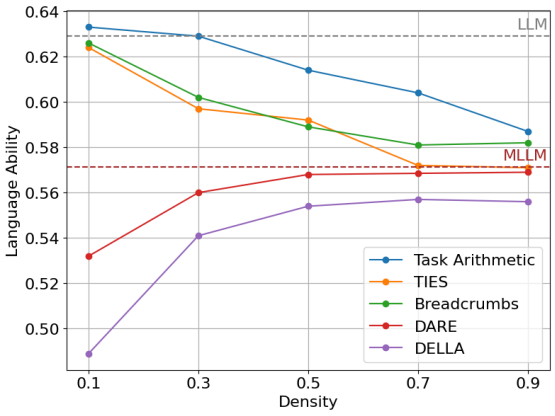

Figure 6: Language ability under different density K.

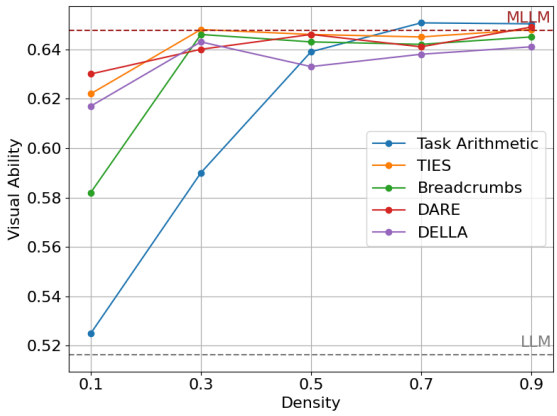

Figure 7: Visual ability under different density K.

Ablation on Neuron-Suppression density $K\%$. We conduct an ablation study to analyze the impact of the suppression density hyperparameter $K\%$, which controls the proportion of parameters retained during the Neuron-Suppress stage. As shown in Figure 6 and 7, we evaluate multiple merging strategies across different $K\%$ on Llama3. The curves of Mistral are shown in Appendix B, which have similar trends. On language ability, DARE and DELLA consistently perform worse than the MLLM across all densities, indicating they fail to recover even the degraded language skills. In contrast, Task Arithmetic, TIES, and Breadcrumbs show monotonically decreasing performance. On visual ability, all methods reach their lowest performance at K=0.1, then improve and decline again—showing a inflection point. Identifying this inflection point is crucial for balancing language retention and visual adaptation.

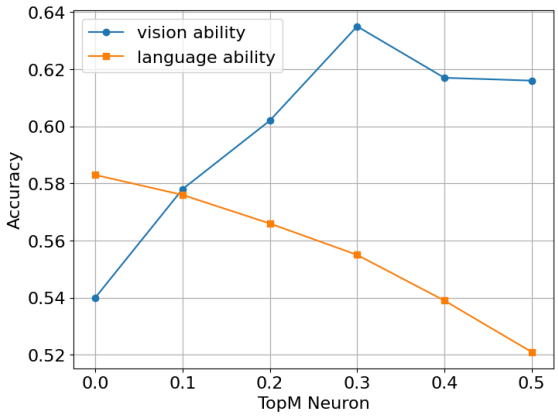

Figure 8: Results when restoring TopM (%) neurons.

Ablation on Neuron-Restore ratio $M\%$. We further study the impact of the neuron restoration ratio $M\%$, which controls the fraction of neurons whose changes are restored. Using TIES with K=0.2 and Neuron-Rescale, we vary M and plot the results in Figure 8. When $M = 0.0$, the vision ability is low due to the suppression of all neuron-level changes. As M increases from 0.0 to 0.5, the language ability decreases gradually, while the vision ability improves sharply up to $M = 0.3$ and then begins to decline. This trend indicates that a moderate value of M achieves a good balance between retaining visual capabilities and mitigating catastrophic forgetting. These results align with our hypothesis that large-change neurons are more critical for storing visual capabilities. Restoring the large-change neurons enhances vision ability, but extending restoring to more neurons introduces diminishing returns and may even harm performance.

Generation analysis. To better understand how Neuron-Fusion improves the accuracy in visual datasets, we analyze the generations at two stages of the Neuron-Fusion process. The most significant improvement occurs on the ScienceQA dataset. The accuracy after hard-merge is 0%. After Neuron-Suppress with $K = 0.2$, the accuracy increases to 7%. After Neuron-Restore with $M = 0.3$, the accuracy improves substantially to 52.6%. To investigate this improvement, we examine the transitions from false to correct generations between the stages of 0% to 7%, and 7% to 52.6%. We identify two major types of failure cases, as illustrated in Table 4. The first two examples fall under the "Not-Known" category, where the model refuses to answer. The last two examples are categorized as "Context-Hallucination," where the model produces content not grounded in the input. For instance, in the final example of Table 4, the false answer “flamboyant cuttlefish” does not correspond to any of the choices in the question.

We find that the Neuron-Suppress stage primarily addresses the "Not-Known" problem: 56.7% of the corrected cases at this stage fall into this category. In contrast, the Neuron-Restore stage is particularly effective at resolving "Context-Hallucination" errors, correcting 97.2% of such cases in this stage. These findings highlight that Neuron-Fusion effectively mitigates two critical issues in multimodal generation: uncertainty in answering and hallucination beyond provided context, which demonstrates that our method not only recovers quantitative performance but also qualitatively enhances output consistency and trustworthiness.

Question & Choices	False answer	Correct answer
Which continent is highlighted? A. Africa; B. North America; C. South America; D. Asia	I cannot directly see the image or the answer choices	D
Which closing is correct for a letter? A. see you soon, Rose; B. See you soon, Rose	I cannot directly answer from the choices	B
Which figure of speech is used in this text? Sing, O goddess, the anger of Achilles son of Peleus A. chiasmus; B. apostrophe	B. Some of the grass on the ground is burning.	B
Which animal's skin is better adapted as a warning sign? A. lichen katydid; B. opalescent nudibranch	B. flamboyant cuttlefish	B

Table 4: Examples of false to correct generations on the ScienceQA dataset after Neuron-Fusion: "Not-Known" (first 2) and "Context-Hallucination" (last 2) problems.

6 Conclusion

We propose Locate-then-Merge, a framework for mitigating catastrophic forgetting in MLLMs. Based on this framework, we develop Neuron-Fusion, a neuron-level parameter fusion method that selectively preserves neurons with large parameter shifts while suppressing harmful widespread changes. Through extensive experiments across language and vision benchmarks, Neuron-Fusion consistently outperforms existing model merging techniques, achieving better retention of both language and visual capabilities. Furthermore, generation analysis reveals that our method effectively reduces common failure modes such as Not-Known and Context-Hallucination, leading to more reliable and controllable model outputs. These results demonstrate the potential of neuron-level fusion strategies for advancing MLLMs' abilities.

7 Limitations

Our work focuses on the vision modality and the visual instruction tuning paradigm for MLLMs. While the proposed Neuron-Fusion method demonstrates strong performance on vision-language benchmarks, we have not investigated whether the same approach can be effectively extended to other modalities such as audio or video. Also, we do not examine its applicability to alternative vision-language model architectures such as CLIP. Additionally, our method is specifically developed for decoder-only LLMs, which currently represent the dominant architecture in high-performing language and multimodal models. Future work is needed to evaluate the generalizability of our approach across diverse multimodal frameworks and architectures.

The Locate-then-Merge framework and the Neuron-Fusion method are specifically designed for scenarios involving parameter merging between a base model and its fine-tuned counterpart, assuming both models share the same architecture and differ only in learned weights. As such, our method may not be directly applicable to settings involving architectural discrepancies or to merging independently trained models with differing objectives or tasks. Extending the framework to accommodate more diverse model merging scenarios remains an important avenue for future exploration.

8 Acknowledgements

This work is supported by the computational shared facility and the studentship from the Department of Computer Science at the University of Manchester.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, and 1 others. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Anton Alexandrov, Veselin Raychev, Mark Niklas Müller, Ce Zhang, Martin Vechev, and Kristina Toutanova. 2024. Mitigating catastrophic forgetting in language transfer via model merging. *arXiv preprint arXiv:2407.08699*.

Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, and 1 others. 2020. Piqa: Reasoning about physical

commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, and 1 others. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*.

MohammadReza Davari and Eugene Belilovsky. 2024. Model breadcrumbs: Scaling multi-task model merging with sparse masks. In *European Conference on Computer Vision*, pages 270–287. Springer.

Pala Tej Deep, Rishabh Bhardwaj, and Soujanya Poria. 2024. Della-merging: Reducing interference in model merging through magnitude-based sampling. *arXiv preprint arXiv:2406.11617*.

Yujie Feng, Xu Chu, Yongxin Xu, Zexin Lu, Bo Liu, Philip S Yu, and Xiao-Ming Wu. 2024a. Kif: Knowledge identification and fusion for language model continual learning. *arXiv preprint arXiv:2408.05200*.

Yujie Feng, Xu Chu, Yongxin Xu, Guangyuan Shi, Bo Liu, and Xiao-Ming Wu. 2024b. Tasl: Continual dialog state tracking via task skill localization and consolidation. *arXiv preprint arXiv:2408.09857*.

Yujie Feng, Xujia Wang, Zexin Lu, Shenghong Fu, Guangyuan Shi, Yongxin Xu, Yasha Wang, Philip S Yu, Xu Chu, and Xiao-Ming Wu. 2025. Recurrent knowledge identification and fusion for language model continual learning. *arXiv preprint arXiv:2502.17510*.

Jonathan Frankle and Michael Carbin. 2018. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*.

Chaoyou Fu, Peixian Chen, and Xunyang Shen. 2023. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2024. A framework for few-shot language model evaluation.

Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. *arXiv preprint arXiv:2203.14680*.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2020. Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2012.14913*.

Charles Goddard, Shamane Siriwardhana, Malikeh Ehghaghi, Luke Meyers, Vladimir Karpukhin, Brian Benedict, Mark McQuade, and Jacob Solawetz. 2024. Arcee's mergekit: A toolkit for merging large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 477–485.

Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. 2013. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2022. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*.

Fengqing Jiang. 2024. Identifying and mitigating vulnerabilities in llm-integrated applications. Master's thesis, University of Washington.

Ronald Kemker, Marc McClure, Angelina Abitino, Tyler Hayes, and Christopher Kanan. 2018. Measuring catastrophic forgetting in neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, and 1 others. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K Kummerfeld, and Rada Mihalcea. 2024. A mechanistic understanding of alignment algorithms: A case study on dpo and toxicity. *arXiv preprint arXiv:2401.01967*.

Bo Li, Peiyuan Zhang, Kaichen Zhang, Fanyi Pu, Xinrun Du, Yuhao Dong, Haotian Liu, Yuanhan Zhang, Ge Zhang, Chunyuan Li, and Ziwei Liu. 2024. Lmms-eval: Accelerating the development of large multimoal models.

Zongxia Li, Xiyang Wu, Hongyang Du, Huy Nghiem, and Guangyao Shi. 2025. Benchmark evaluations, applications, and challenges of large vision language models: A survey. *arXiv preprint arXiv:2501.02189*.

Zijing Liang, Yanjie Xu, Yifan Hong, Penghui Shang, Qi Wang, Qiang Fu, and Ke Liu. 2024. A survey of multimodel large language models. In *Proceedings of the 3rd International Conference on Computer, Artificial Intelligence and Control Engineering*, pages 405–409.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, and 1 others. 2024b. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pages 216–233. Springer.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2023. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*.

Yaniv Nikankin, Anja Reusch, Aaron Mueller, and Yonatan Belinkov. 2024. Arithmetic without algorithms: Language models solve math with a bag of heuristics. *arXiv preprint arXiv:2410.21272*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Neale Ratzlaff, Man Luo, Xin Su, Vasudev Lal, and Phillip Howard. 2024. Training-free mitigation of language reasoning degradation after multimodal instruction tuning. *arXiv preprint arXiv:2412.03467*.

Sarah Schwettmann, Neil Chowdhury, Samuel Klein, David Bau, and Antonio Torralba. 2023. Multimodal neurons in pretrained text-only transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2862–2867.

Noam Shazeer. 2020. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Gido M Ven, Nicholas Soures, and Dhireesha Kudithipudi. 2024. Continual learning and catastrophic forgetting. *arXiv preprint arXiv:2403.05175*.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and 1 others. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pages 23965–23998. PMLR.

Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang

Ma, Chengyue Wu, Bingxuan Wang, and 1 others. 2024. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36:7093–7115.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. 2024b. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666*.

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

Zeping Yu and Sophia Ananiadou. 2023. Neuron-level knowledge attribution in large language models. *arXiv preprint arXiv:2312.12141*.

Zeping Yu and Sophia Ananiadou. 2024. Interpreting arithmetic mechanism in large language models through comparative neuron analysis. *arXiv preprint arXiv:2409.14144*.

Zeping Yu, Yonatan Belinkov, and Sophia Ananiadou. 2025. Back attention: Understanding and enhancing multi-hop reasoning in large language models. *arXiv preprint arXiv:2502.10835*.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, and 1 others. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567.

Yi-Kai Zhang, Shiyin Lu, Yang Li, Yanqing Ma, Qing-guo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, De-Chuan Zhan, and Han-Jia Ye. 2024. Wings: Learning multimodal llms without text-only forgetting. *Advances in Neural Information Processing Systems*, 37:31828–31853.

Didi Zhu, Zhongyi Sun, Zexi Li, Tao Shen, Ke Yan, Shouhong Ding, Kun Kuang, and Chao Wu. 2024. Model tailor: Mitigating catastrophic forgetting in multi-modal large language models. *arXiv preprint arXiv:2402.12048*.

A Detailed Results on All Datasets

A.1 Catastrophic Forgetting in MLLMs

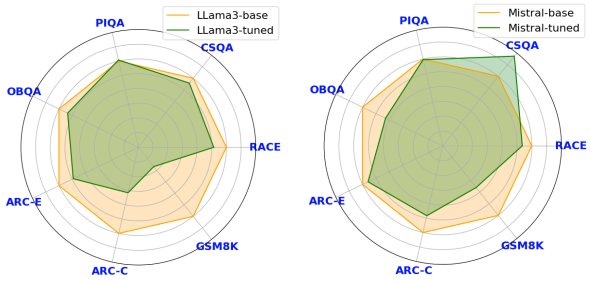

Figure 9: Accuracy of Llama3 (left) and Mistral (right) on language datasets after visual instruction tuning.

We present the accuracy of Llama3 and Mistral on the language datasets before and after visual instruction tuning in Figure 9. Except for Mistral's result on CommonsenseQA, the accuracy consistently decreases across most datasets, with particularly significant drops observed on GSM8K and ARC-Challenge. These results confirm the presence of catastrophic forgetting in the language capabilities after visual instruction tuning.

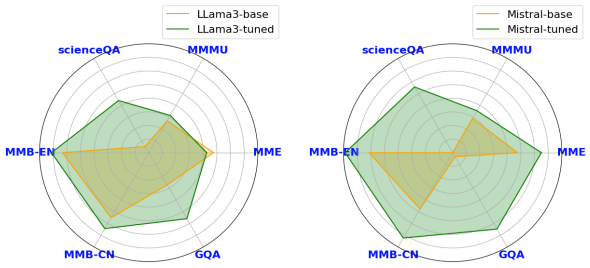

Figure 10: Accuracy of Llama3 (left) and Mistral (right) on vision datasets after hard-merge into tuned MLLMs.

As introduced in Section 3.2, a straightforward approach for recovering language ability is to directly replace the tuned MLLM's parameters with the base LLM. In Figure 10, we present the accuracy on visual datasets after hard-merging the original parameters of Llama3 and Mistral into their corresponding tuned MLLMs. Across both models and datasets, the accuracy drops significantly. Therefore, it is necessary to design a more effective method that can restore the language capabilities while preserving the visual capabilities.

A.2 Results on All Datasets

The detailed results on 6 visual datasets and 7 language datasets in Llama3 and Mistral are shown in Table 5 and Table 6, respectively.

B Different Density in Mistral

The language and visual capabilities of Task Arithmetic, TIES, Breadcrumbs, DARE, and DELLA under different density $K\%$ in Mistral are illustrated in Figures 11 and 12. These results exhibit trends similar to those observed with Llama3. A difference is that the Mistral-based methods—Task Arithmetic, TIES, and Breadcrumbs—achieve higher accuracy than the standalone LLM when the density is low. This is attributed to the MLLM achieving better performance on the CommonsenseQA dataset compared to the LLM (see Table 6). By integrating the MLLM with the LLM, the combined model benefits from the performance gains on this dataset.

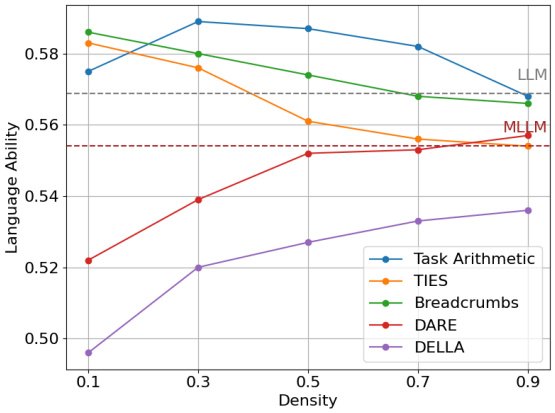

Figure 11: Language ability under different density K.

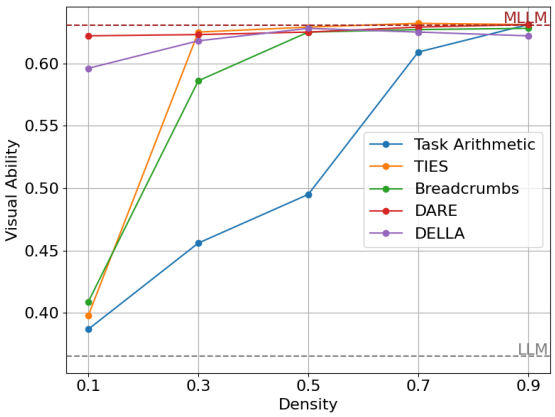

Figure 12: Visual ability under different density K.

Method	MME	MMMU	SciQA	MMB-E	MMB-C	GQA	RACE	CSQA	PIQA	OBQA	ARC-E	ARC-C	GSM8K
MLLM	71.2	41.6	54.2	81.2	74.4	66.0	42.2	73.0	79.2	32.8	74.8	42.8	55.2
LLM	64.0	37.0	15.2	73.2	65.2	37.2	44.6	75.0	78.8	34.2	80.2	52.0	75.6
Task Arithm	71.8	39.6	55.8	81.8	74.6	59.8	43.8	76.8	79.8	35.6	79.0	48.6	66.2
TIES	70.4	41.0	53.2	78.8	73.4	56.4	44.8	78.0	79.2	36.2	79.0	49.0	70.8
Breadcrumbs	71.0	40.8	55.6	82.0	74.0	64.2	43.8	76.6	79.6	34.8	75.8	46.6	64.2
DARE	71.8	41.6	54.4	81.0	74.4	66.4	42.4	72.8	79.2	32.8	74.8	42.2	54.2
DELLA	70.4	39.8	54.8	79.6	74.4	65.6	42.0	71.2	77.8	32.8	72.6	41.2	51.6
Neu-P-TaskA	71.0	41.4	57.6	81.0	74.2	58.2	44.0	77.8	79.8	35.2	79.4	49.4	67.6
Neu-P-TIES	70.4	40.6	54.4	80.4	75.2	58.6	43.6	77.8	78.8	36.0	78.2	48.6	69.6
Neu-S-TIES	70.6	40.2	54.4	80.4	75.4	60.0	43.8	78.0	79.4	35.0	77.4	47.0	67.8
Neu-P-Bread	70.0	40.8	56.2	80.0	75.0	59.2	44.0	78.2	78.6	36.0	78.0	48.4	70.8
Neu-S-Bread	70.0	40.0	57.0	80.2	75.0	58.2	44.4	77.0	78.6	35.0	78.8	47.8	70.6

Table 5: Results of different methods on 6 visual datasets and 7 language datasets in Llama3-7B.

Model	MME	MMMU	SQA	MMB-E	MMB-C	GQA	RACE	CSQA	PIQA	OBQA	ARC-E	ARC-C	GSM8K
MLLM	71.0	35.6	55.8	78.8	72.4	64.8	43.0	73.6	81.2	31.6	76.0	48.4	34.0
LLM	53.8	29.0	0.0	61.2	47.8	3.4	44.8	66.4	81.4	35.4	78.0	52.2	40.0
Task Arithm	71.4	36.0	56.8	79.4	72.0	63.4	43.6	74.2	82.2	32.8	77.0	49.4	38.8
TIES	72.8	37.0	53.8	78.0	73.0	60.8	44.0	74.0	83.0	32.6	78.4	51.0	40.2
Breadcrumbs	71.6	34.8	56.0	78.4	72.2	62.2	44.2	74.2	82.6	33.0	78.2	50.8	39.2
DARE	71.4	35.8	56.0	79.0	72.0	64.6	42.8	73.6	81.6	31.8	75.8	48.2	36.4
DELLA	70.2	33.2	55.8	78.6	70.6	65.2	41.8	72.6	80.6	30.2	74.2	46.4	29.6
Neu-P-TaskA	72.8	36.6	53.6	79.2	73.2	61.0	43.8	74.2	82.8	33.2	78.6	51.8	40.2
Neu-P-TIES	72.6	36.0	54.2	78.0	72.8	61.0	44.0	73.4	83.0	32.6	78.0	50.8	39.6
Neu-S-TIES	72.8	36.2	54.2	78.0	73.0	62.2	43.4	72.4	83.0	31.4	77.6	50.4	39.8
Neu-P-Bread	72.2	35.8	57.0	79.0	72.0	62.4	44.2	73.8	82.4	33.0	77.0	49.6	39.6
Neu-S-Bread	72.2	35.6	51.4	79.0	72.0	61.0	43.6	73.4	83.2	33.0	78.2	50.8	39.0

Table 6: Results of different methods on 6 visual datasets and 7 language datasets in Mistral-7B.