RCP-Merging: Merging Long Chain-of-Thought Models with Domain-Specific Models by Considering Reasoning Capability as Prior

Junyao Yang[1], **Jianwei Wang**[1], **Huiping Zhuang**[1], **Cen Chen**[1], **Ziqian Zeng**[1] *

[1]South China University of Technology, China

tonyy10264611@gmail.com, wiwjwilliam@mail.scut.edu.cn, {hpzhuang, chencen, zqzeng}@scut.edu.cn

Abstract

Large Language Models (LLMs) with long chain-of-thought (CoT) capability, termed Reasoning Models, demonstrate superior intricate problem-solving abilities through multi-step long CoT reasoning. To create a dual-capability model with long CoT capability and domain-specific knowledge without substantial computational and data costs, model merging emerges as a highly resource-efficient method. However, significant challenges lie in merging domain-specific LLMs with long CoT ones since nowadays merging methods suffer from reasoning capability degradation, even gibberish output and output collapse. To overcome this, we introduce **RCP-Merging**: Merging Long Chain-of-Thought Models with Domain-Specific Models by Considering **R**easoning **C**apability as **P**rior, a novel merging framework designed to integrate domain-specific LLMs with long CoT capability, meanwhile maintaining model performance in the original domain. Treating reasoning model weights as foundational prior, our method utilizes a reasoning capability indicator to preserve core long CoT capability model weights while selectively merging essential domain-specific weights. We conducted extensive experiments on Qwen2.5-7B, Llama3.1-8B, and Qwen2.5-1.5B models in BioMedicine and Finance domains. Our results show that RCP-Merging successfully merges a reasoning model with domain-specific ones, improving domain task performance by 9.5% and 9.2% over state-of-the-art methods, without significantly harming the original long CoT reasoning capability.

Code — https://github.com/ZeroNLP/RCP-Merging

Datasets — https://github.com/ZeroNLP/RCP-Merging

Introduction

Large Language Models (LLMs) with long chain-of-thought (CoT) capability, termed Reasoning Models, have demonstrated exceptional performance on complex reasoning tasks (Jaech et al. 2024; OpenAI 2025; Guo et al. 2025; xAI 2025). Mostly trained on verifiable tasks like code generation and mathematical reasoning, the results in Table 1 show that the reasoning model demonstrates relatively weak performance compared with models that specifically fine-tune on a certain domain. However, long CoT's multi-step reasoning deduction is critical for complex problem-solving in specific domains like BioMedicine and Finance, extending beyond simple information retrieval (Cui et al. 2025; Tang et al. 2025). Moreover, the scarcity of models specifically trained for these fields remains a key challenge. This difficulty stems from current long CoT realization relying on additional training, which introduces challenges like catastrophic forgetting, inefficient resource allocation, not to mention the inherent difficulty in obtaining high-quality domain reasoning data (Dong et al. 2025; Zeng et al. 2025).

Fortunately, model merging (Li et al. 2023; Ilharco et al. 2023; Yang et al. 2024a) has recently emerged as a resource-efficient technique to create a single model with dual capabilities without requiring extra training data. However, a

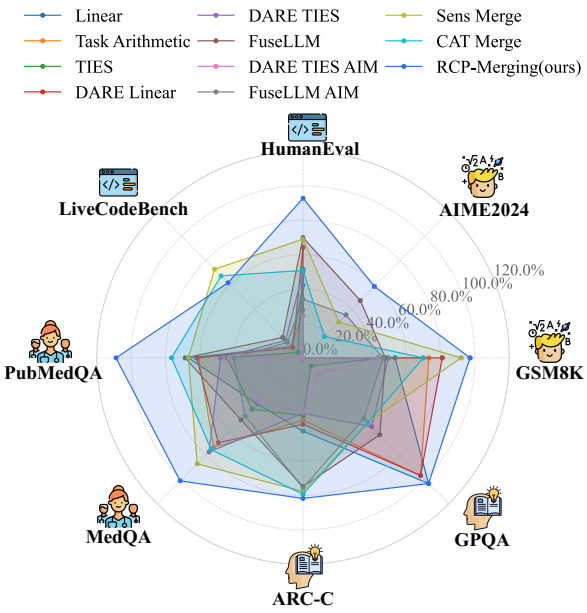


Figure 1: Performance comparison of RCP-Merging and other methods in merging Qwen2.5-7B, Meditron3-Qwen2.5-7B, and DeepSeek-R1-Distill-Qwen-7B on eight datasets in Math, Code, BioMedicine, and Knowledge areas.

*Corresponding author.

significant gap exists that current model merging focuses on combining models for certain domains, such as merging a model specialized in General Knowledge with one for Chinese. As results show in the LiveCodeBench (Jain et al. 2024) and AIME (Veeraboina 2023) datasets in Figure 1, trying to merge a reasoning model with a domain-specific one often leads to a collapse of the output and a sharp performance decline. Therefore, it is highly valuable to find a method that can successfully integrate a domain-specific model with a reasoning model and subsequently boost the merged model's performance on its original domain-specific tasks.

To tackle this problem, existing merging methods often struggle to preserve long CoT capabilities when integrating reasoning models with domain-specific ones. For instance, some methods (Ilharco et al. 2023; Wan et al. 2024) operate under the assumption that larger weights are more important. By trimming the smaller weights (Yadav et al. 2023) or rescaling the larger weights (Yu et al. 2024), these methods create significant risks as the large-magnitude weights from a domain-specific model can easily overwrite the smaller, yet more critical weights for long CoT capability. Other works (Liu et al. 2025; Nobari et al. 2025) utilize the product of weight magnitude and its gradient on a certain domain to identify how crucial the model weight is. Some do this by identifying key neurons to preserve crucial knowledge (Ma et al. 2025) while others resolve knowledge conflicts before merging (Sun et al. 2025a,b). However, domain-specific gradient is not a suitable proxy for long CoT, as they often track performance adjustments on certain domains instead of the multi-step reasoning deduction that is crucial for long CoT capability. These superficial gradients make it challenging to identify and preserve the specific weights that are essential to long CoT capability (Thapa et al. 2025; Hao et al. 2025; Zeng et al. 2025). Consequently, merged models through these methods inadvertently compromise long CoT capability. Moreover, as shown in Figure 3, these models lead to the generation of non-sensical gibberish outputs, highlighting the primary challenge of improving performance in a specific domain without sacrificing long CoT capability.

Motivated by this objective, we propose our core method: Merging Long Chain-of-Thought Models with Domain-Specific Models by Considering **R**easoning **C**apability as **P**rior (**RCP-Merging**) RCP-Merging is a framework designed to equip a domain-specific model with long CoT capability by merging with a reasoning model and further boosting the merged model's performance on its original domain-specific tasks. The cornerstone of our method is the **Reasoning Preservation Indicator**. Instead of relying on conventional methods focusing on the gradient of loss on a certain domain and the magnitude of model weight, our method treats the model's long CoT capabilities as a guiding principle for the merge. It conceptually views reasoning model's parameters as a stable prior, constraining updates that would significantly deviate from this established reasoning foundation using the Fisher Information Matrix (Fisher 1925) gained from each calibration data. This ensures that as the model acquires new domain-specific knowledge, it is given an indicator for each model weight to ensure the merged weight does not greatly harm the long CoT capability, consistently preventing catastrophic forgetting, gibberish output, and long CoT capability degradation that emerged from previous methods. Our framework complements this with **Domain Knowledge Sensitivity** to identify and retain important domain-specific weights. Finally, **Reasoning-preserved Merging** step synthesizes these factors, utilizing both the reasoning preservation matrix and domain sensitivity as a comprehensive guide to select the most critical parameters for the final model, achieving a robust balance between domain-specific knowledge and long CoT capability.

We demonstrate that RCP-Merging, requiring only a small number of open-source calibration samples, can effectively integrate long CoT reasoning capabilities into a domain-specific model. Through extensive experiments across various tasks and model architectures like Qwen2.5 (Qwen 2024) and Llama3.1 (Grattafiori et al. 2024), our method consistently produces merged models that not only preserve domain-specific expertise but also exhibit surprisingly long CoT capabilities when addressing domain-specific questions, ultimately elevating their performance in certain domains. Notably, the average performance of the merged model on eight datasets improves by 9.5% and 9.2% compared with the state-of-the-art method on BioMedicine and Finance domains, respectively. Moreover, though model merging aims to find a comprehensive model that compromises the performance of original models, our method improved performance by 4.5% and 0.7% on PubMedQA and MedQA datasets (Jin et al. 2019, 2020), respectively, and improved performance by 0.5% on ConvFinQA dataset (Cheng, Huang, and Wei 2024) compared to the original domain-specific models. To sum up, our contributions include:

- We propose a novel model merging framework, RCP-Merging, which effectively integrates a domain-specific model with a long CoT reasoning model by treating reasoning ability as a prior.
- We conduct extensive experiments across multiple benchmarks, demonstrating that RCP-Merging surpasses existing methods by preserving both specialized knowledge and long-CoT reasoning capabilities.
- Results surprisingly demonstrate that models merged via RCP-Merging exhibit emergent long CoT reasoning capabilities within model outputs when handling domain-specific problems.

Related Work

Model merging (Goddard et al. 2024a; Yang et al. 2024a; Ruan et al. 2025; Li et al. 2023) aims to combine multiple specialized models into a single, powerful model without costly retraining (Ilharco et al. 2023; Yadav et al. 2023; Yang et al. 2024b). Existing approaches can be broadly categorized based on the information they use to determine how parameters are combined: magnitude-based methods that operate directly on parameter values, and activation-based methods that leverage model outputs or gradients on calibration data.

Magnitude-Based Methods

Magnitude-based methods merge models by performing arithmetic operations directly on their weight parameters or task vectors, often using parameter magnitude as a proxy for importance.

A foundational approach is simple Linear or weight averaging, which calculates the element-wise mean of the parameters of all models to be merged (Izmailov et al. 2018). Task Arithmetic (Ilharco et al. 2023) refines this by first computing task vectors, defined as the difference between fine-tuned and pre-trained weights ($\delta_{ft} = \theta_{ft} - \theta_{pre}$). These vectors, representing task-specific knowledge, are then combined through arithmetic operations like addition or negation before being applied to the base model.

To mitigate interference between task vectors, several methods have been proposed. TIES-Merging (Yadav et al. 2023) introduces a three-step process: it trims each task vector by retaining only a top-k of high-magnitude parameters and resetting the rest to zero, then elects a single, dominant sign for each parameter across all task vectors. DARE (Yu et al. 2024) and PCB-Merging (Du et al. 2024) adjust model weights to reduce task conflicts by randomly dropping a ratio of weights and rescaling the remaining ones. FuseLLM (Wan et al. 2024) operates by leveraging the generative probability distributions of diverse source LLMs to externalize their knowledge, which is then transferred to a single target model through a lightweight continual training phase.

A primary drawback of magnitude-based methods is their assumption that parameter magnitude equates to importance. This can lead to the retention of high-magnitude parameters that are harmful to other models, causing significant knowledge conflicts and degrading the performance of the merged model.

Activation-Based Methods

To address the limitations of magnitude-based approaches, activation-based methods leverage data-driven signals, such as model activations or gradients on a small calibration set, to obtain a more nuanced understanding of parameter importance (Springenberg et al. 2015; Michel, Levy, and Neubig 2019; Maini et al. 2023; Liu et al. 2024).

Sens-Merging (Liu et al. 2025) operates at two levels to perform task-specific analysis to identify the sensitivity of each layer and evaluate cross-task transferability between different models on a calibration dataset. CAT-Merging (Sun et al. 2025a) directly tackles knowledge conflict (Sun et al. 2025b) by identifying and trimming conflict-prone components from task vectors. Using a few unlabeled examples, it computes layer-specific projection operators for linear weights and masks for normalization parameters to resolve interference before merging.

Moreover, Fisher Merging (Matena and Raffel 2022) and RegMean (Jin et al. 2023) using Fisher Information Matrix to determine parameter importance or utilizing local regression for model merging; however, these approaches are characterized by high computational complexity. Other methods, such as Activation-Informed Merging (AIM) (Nobari et al. 2025) and LED-Merging (Ma et al. 2025) utilize activations to guide the merging process, offering ways to find neurons that are crucial to certain domains.

While these activation-based methods can more effectively mitigate the knowledge conflicts seen in magnitude-based approaches, they have their own limitations since the gradient-based evaluation is hard to capture the complex, sequential reasoning patterns within the model's weight.

Preliminary

Task Vector. We adopt the concept of task vectors from the field of model merging. A task vector, δ, represents the knowledge acquired by a model during fine-tuning for a specific task. It is computed as the difference between the weights of the fine-tuned model and base model, θ_t, where t represents the domain-specific task. The weights of the original pre-trained base model is represented by θ_{pre}:

$$\delta_t = \theta_t - \theta_{pre}, \text{ for } t \in \{1, ..., T\}. \tag{1}$$

In our framework, we define a task vector for each domain-specific model, $\delta_t = \theta_t - \theta_{pre}$, and a task vector for the reasoning model, $\delta_r = \theta_r - \theta_{pre}$, where θ_t and θ_r are the weights of the domain-specialized model and the long-chain reasoning model, respectively. Task vector-based merging combines these task vectors into a single, static model:

$$\theta_{merged} = \theta_{pre} + \sum_{t=1}^{T} \lambda \cdot \delta_t, \tag{2}$$

where the coefficient λ represents the importance of each merged task vector.

Fisher Information Matrix. The Fisher Information Matrix (FIM) is a fundamental concept in information geometry that quantifies the amount of information an observable random variable, x, carries about an unknown parameter, θ, of a statistical model. For a model with parameters θ, the FIM element $F(\theta)_{ij}$ is defined as the expected value of the outer product of the gradients of the log-likelihood function, the (i, j)-th element of the matrix can be denoted as:

$$F(\theta)_{ij} = E_{x\sim p(x|\theta)} \left[\left(\frac{\partial}{\partial \theta_i} \log p(x|\theta) \right) \left(\frac{\partial}{\partial \theta_j} \log p(x|\theta) \right) \right]. \tag{3}$$

This can also be expressed as the negative expected value of the Hessian of the log-likelihood:

$$F(\theta)_{ij} = -E_{x\sim p(x|\theta)} \left[\frac{\partial^2}{\partial \theta_i \partial \theta_j} \log p(x|\theta) \right]. \tag{4}$$

In the context of autoregressive tasks where the loss function $\mathcal{L}(\theta, x)$ is the negative log-likelihood, i.e., $\mathcal{L}(\theta, x) = -\log p(x|\theta)$, the diagonal elements F_i of the FIM can be approximated by the expected squared gradient of the loss function. For a single i-th diagonal parameter θ_i and a dataset D_r, this approximation is:

$$F(\theta)_i \approx E_{d\sim D_r} \left[\left(\frac{\partial \mathcal{L}(\theta, d)}{\partial \theta_i} \right)^2 \right] = E_{d\sim D_r} \left[(g_{i,d})^2 \right], \tag{5}$$

where $g^r_{i,d}$ is the gradient of the loss with respect to the parameter θ^i_r for a given data sample d. This approximation is pivotal for calculating our reasoning capability indicator.

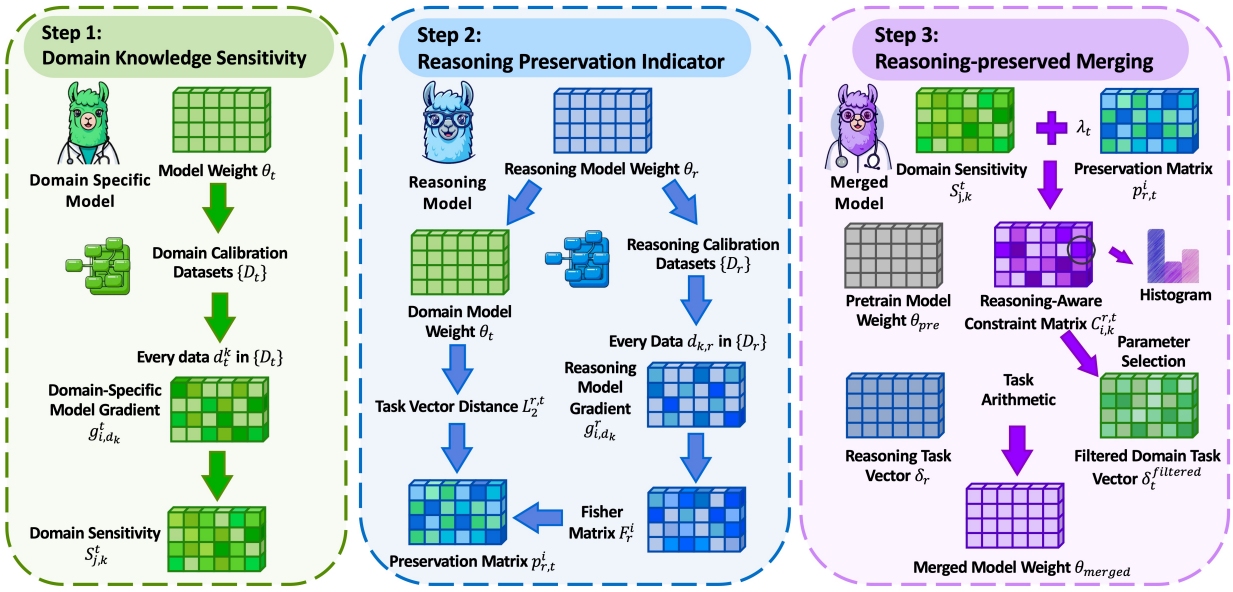


Figure 2: RCP-Merging consists of three stages. (1) **Domain Knowledge Sensitivity**. This step quantifies each weight's importance for a specific domain by measuring the change in model loss when that weight is removed. (2) **Reasoning Preservation Indicator**. To protect the model's core reasoning capabilities, this stage applies a preservation term to weights that are crucial for reasoning. (3) **Reasoning-preserved Merging**. The final stage balances domain sensitivity and the reasoning preserving matrix, merging only the weights that enhance domain knowledge without harming reasoning capabilities.

Methodology

Our methodology is designed to merge models by integrating domain-specific knowledge while preserving long CoT capability. This is achieved by first identifying parameters crucial for domain-specific tasks and then applying a preservation term derived from the Bayesian rule to mitigate the degradation of reasoning abilities. The final model is constructed by selectively merging domain-specific task vectors based on a reasoning-aware constraint matrix, as shown in Figure 2.

Domain Knowledge Sensitivity

To quantify the importance of each parameter on the domain-specific model for a task t, by setting the corresponding model as the domain-specific task model θ_t, we introduce the concept of Domain Knowledge Sensitivity, $S_{i,k}^t$. This metric measures the impact on the model's performance when a particular weight is nullified.

Given a domain-specific model with parameters $\theta_t = [\theta_1, \theta_2, \ldots, \theta_N]$ and a calibration dataset $\{D_t\}$, the sensitivity of the i-th parameter θ_t^i with respect to a data sample $d_t^k \in D_t$ is defined as the change in the loss function:

$$S_{i,k}^t = \left[\mathcal{L}(\theta_t) - \mathcal{L}(\theta_t - \theta_t^i)\right]_{d=d_t^k}, \tag{6}$$

where θ_t^i is a vector with only the i-th parameter being non-zero.

For computational efficiency, we approximate this value using a first-order Taylor expansion. This simplifies sensitivity to the product of the parameter and its corresponding gradient, $g_{i,d_k}^t = \frac{\partial \mathcal{L}(\theta_t)}{\partial \theta_t^i}$, as follows:

$$S_{i,k}^t \approx \left\| \frac{\partial \mathcal{L}(\theta_t)}{\partial \theta_t^i} \cdot \theta_t^i \right\|_{d=d_t^k} \approx \| g_{i,d_k}^t \cdot \theta_t^i \|_{d=d_t^k}. \tag{7}$$

A lower sensitivity score indicates that the parameter θ_t^i contributes positively to the model's performance in the specific domain, as its presence reduces the loss.

Reasoning Preservation Indicator

To prevent the primary drawback of output collapse emerging from previous methods when merging with reasoning models, we introduce a preserving function to indicate important weights in the model merging process. Inspired by Kirkpatrick et al. (2016), we adopt the Bayesian rule where the reasoning model's parameter distribution serves as a prior for the posterior distribution of the final merged model's parameters. Our goal is to find the parameters θ_t that maximize the posterior probability (MAP estimation), which is equivalent to minimizing the negative log-posterior:

$$\theta_{MAP} = \arg\min_{\theta_t} [-\log P(D_t|\theta_t) - \log P(\theta_t|D_r)]. \tag{8}$$

The term $-\log P(\theta_t|D_r)$ acts as a regularization term, discouraging the parameters from deviating significantly from the optimal weights learned on the reasoning task, which we denote as θ_r^*.

However, directly computing the true posterior $P(\theta_t|D_r)$ is intractable for complex neural networks. To address this, we employ the Laplace approximation, which approximates

the posterior with a Gaussian distribution centered at the mode θ_r^*: $P(\theta_t|D_r) \approx \mathcal{N}(\theta_t|\theta_r^*, F_r^{-1})$. The precision matrix of this Gaussian is the Fisher Information Matrix (FIM), F_r, which measures the curvature of the log-likelihood landscape. The probability density function is:

$$P(\theta_t|D_r) \approx \frac{|F_r|^{1/2}}{(2\pi)^{k/2}} \exp\left(-\frac{1}{2}(\theta_t - \theta_r^*)^T F_r(\theta_t - \theta_r^*)\right). \tag{9}$$

By taking the natural logarithm and discarding terms that are constant with respect to θ_t, we simplify the expression for optimization purposes. This yields a tractable form for the log-posterior preservation matrix:

$$\log P(\theta_t|D_r) \approx -\frac{1}{2}(\theta_t - \theta_r^*)^T F_r(\theta_t - \theta_r^*). \tag{10}$$

This quadratic term measures how much the updated parameters θ_t have diverged from the reasoning-optimal parameters θ_r^*, weighted by the FIM F_r. A higher value in F_r for a certain parameter indicates its importance for the reasoning task, and thus incurs a larger preservation for any deviation.

To make this computation more feasible, we assume a diagonal FIM. As shown in Equation 5, this simplifies $p_{r,t}^i$ into a sum of per-parameter contributions, where for each parameter θ_t^i, the penalty is $\log P(\theta_t^i|D_r) \approx -\frac{1}{2}F_{r,ii}(\theta_t^i - \theta_{r,i}^*)^2$. The i-th diagonal elements of the FIM, $F_{r,ii}$, can be approximated by the average of the squared gradients over the calibration reasoning dataset $D_r = \{d_k\}_{k=1}^{N_r}$. Combining these steps, we define the final reasoning preservation indicating matrix $p_{r,t}^i$ for each parameter θ^i as:

$$p_{r,t}^i = \left\| -\frac{1}{2N_r} \sum_{k=1}^{N_r} (g_{i,d_k}^r)^2 (\theta_t^i - \theta_r^i)^2 \right\|. \tag{11}$$

Here, g_{i,d_k}^r is the gradient of the loss for sample d_k with respect to parameter θ_r^i. This metrics quantifies how much the new parameter θ_t^i impairs the model's reasoning ability.

Reasoning-preserved Merging

To integrate domain knowledge while preserving core reasoning skills, we propose reasoning-aware merging strategy. We implement this by defining a Constraint metric $C_{i,k}^{r,t}$ for each parameter θ_i to quantify the importance of long CoT capability, combining its Domain Knowledge Sensitivity ($S_{i,k}^t$) and Reasoning Capability Indicator ($p_{r,t}^i$):

$$C_{i,k}^{r,t} = S_{i,k}^t + \lambda_r \cdot p_{r,t}^i. \tag{12}$$

Here, the hyperparameter λ_r balances the trade-off between domain performance and long CoT capability preservation.

Next, we filter parameter updates using a majority vote criterion. An update for parameter θ_t^i is accepted if the number N of data samples in the domain dataset D_t yielding a negative conflict score is greater than the number N of those yielding a non-negative one:

$$\text{Update } \theta_t^i \quad \text{if} \quad N(C_{i,k}^{r,t} < 0) > N(C_{i,k}^{r,t} \geq 0). \tag{13}$$

This condition generates a binary mask $M \in \{0,1\}^N$, where $M_i = 1$ signifies an accepted update for the corresponding parameter.

Finally, we use this mask to create a filtered domain-specific task vector, $\delta_t^{filtered}$, via an element-wise product with the original task vector $\delta_t = \theta_t - \theta_{pre}$. The final model weights, θ_{merged}, are then obtained by adding the complete reasoning vector δ_r and the weighted sum of these filtered task vectors to the pre-trained weights θ_{pre}:

$$\delta_t^{filtered} = M \odot \delta_t, \tag{14}$$

$$\theta_{merged} = \theta_{pre} + \delta_r + \sum_{t=1}^{T} \lambda_t \cdot \delta_t^{filtered}, \tag{15}$$

where T is the number of domain-specific tasks and λ_t are scaling coefficients. This approach ensures the model benefits from domain-specific knowledge while robustly maintaining its reasoning abilities.

Experiment

Experimental Setup

Baselines. We compare RCP-Merging with multiple merging baselines: **Average** (Izmailov et al. 2018), **Task Arithmetic** (Ilharco et al. 2023), **TIES-Merging** (Yadav et al. 2023), **DARE-Merging**, **DARE-Merging** with TIES (Yu et al. 2024), **FuseLLM** (Wan et al. 2024), **FuseLLM with AIM**, **DARE TIES with AIM** (Nobari et al. 2025), **Sens-Merging** (Liu et al. 2025), and **CAT-Merging** (Sun et al. 2025a). We utilize mergekit (Goddard et al. 2024b) as merging tools for baseline methods.

Datasets&Metrics. We assess merged model performance through four pillars: (1) Mathematical reasoning (Math) via GSM8k (Cobbe et al. 2021) and AIME2024 (Veeraboina 2023) (Accuracy↑ with CoT); (2) Code generation (Code) evaluated by HumanEval (Chen et al. 2021) and LiveCodeBench (Jain et al. 2024) (Pass@1↑); (3) Medical question answering (BioMedicine) through PubMedQA (Jin et al. 2019) and MedQA (Jin et al. 2020) (Accuracy↑); (4) General knowledge question answering with ARC-C (Clark et al. 2018) and GPQA (Rein et al. 2023) (Accuracy↑).

Models. The experiment involves a set of models built upon the Qwen2.5-7B (Qwen 2024) Base model architecture. The domain-specific model is Meditron3-Qwen2.5-7B (Chen et al. 2023) for BioMedicine, and the Reasoning model is DeepSeek-R1-Distill-Qwen-7B (Guo et al. 2025).

RCP-Merging's Superior Performance

RCP-Merging achieves a SOTA average performance of 49.4 on the BioMedicine domain, surpassing all baselines by superiorly balancing domain expertise and reasoning capabilities, as shown in Table 1. It obtains top scores of 55.5 on PubMedQA and 54.1 on MedQA, effectively integrating domain knowledge. Simultaneously, it enhances reasoning, achieving state-of-the-art performance among merged models in Math, with scores of 84.3 on GSM8K and 33.3 on AIME2024, and in Code, scoring 71.3 on HumanEval. This highlights RCP-Merging's unique effectiveness, as baseline methods typically sacrifice reasoning for domain performance.

Method/Task	Math		Code		BioMedicine		Knowledge		Average
	GSM8K	AIME2024	HumanEval	LiveCodeBench	PubMedQA	MedQA	ARC-C	GPQA	
Base	69.4	0.0	50.6	12.4	32.5	22.9	60.9	7.6	32.0
BioMedicine	81.5	0.0	54.3	2.2	51.0	53.5	74.9	9.6	40.9
Reasoning	86.7	56.7	76.6	29.8	38.0	30.2	76.5	15.2	51.2
Linear	46.4	0.0	32.3	2.8	34.0	20.2	32.6	**15.7**	23.0
Task Arithmetic	63.5	0.0	21.3	2.8	31.0	39.3	27.7	14.7	25.0
TIES-Merging	40.6	0.0	39.6	1.3	22.5	22.5	25.8	1.0	19.2
DARE Linear	70.2	0.0	49.4	2.5	31.5	37.3	29.5	14.7	29.4
DARE TIES	43.1	0.0	38.4	4.3	24.5	41.4	24.6	8.6	23.1
FuseLLM	41.5	26.7	53.7	5.0	35.0	27.3	57.2	9.6	32.0
DARE TIES AIM	37.8	0.0	18.9	0.4	22.0	19.3	24.2	2.0	15.6
FuseLLM AIM	40.3	20.0	26.2	3.8	20.5	25.5	57.8	7.6	25.2
Sens-Merging	79.8	16.7	53.0	**21.7**	34.0	46.6	59.4	8.1	39.9
CAT-Merging	60.7	10.0	39.0	20.1	39.0	40.4	60.7	8.1	34.8
RCP-Merging	**84.3**	**33.3**	**71.3**	18.4	**55.5**	**54.1**	**82.5**	**15.7**	**49.4**

Table 1: Performance comparison of merging Qwen2.5-7B (Base), Meditron3-Qwen2.5-7B (BioMedicine) and DeepSeek-R1-Distill-Qwen-7B (Reasoning) on all datasets. Best performance of all merging methods on each dataset is highlighted in **bold**.

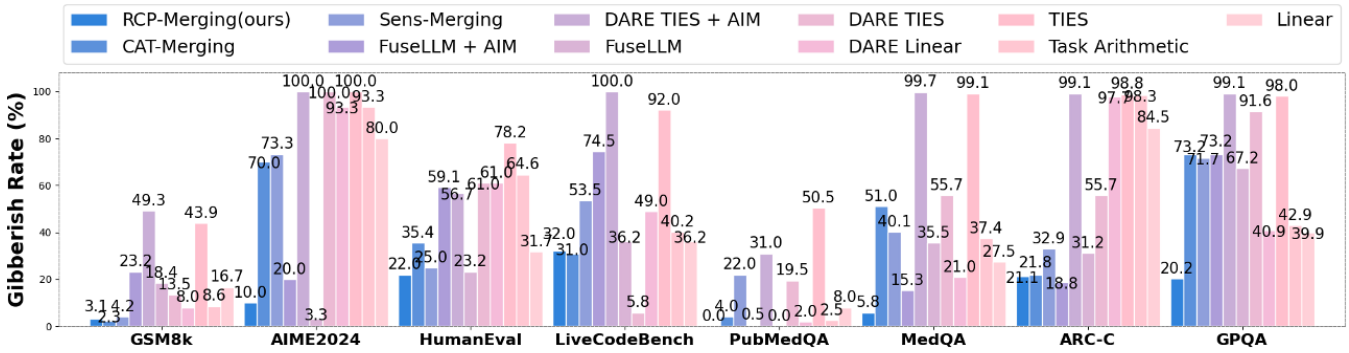


Figure 3: Gibberish rate comparison for merging Qwen2.5-7B (Base), Meditron3-Qwen2.5-7B (BioMedicine), and DeepSeek-R1-Distill-Qwen-7B (Reasoning) on all datasets, where a lower rate indicates higher-quality content.

RCP-Merging's Output Stability. To address model stability, we measure gibberish rate: the frequency of nonsensical outputs identified by a GPT4 evaluator (OpenAI 2023) to validate genuine performance against output degeneration. As shown in Figure 3, RCP-Merging demonstrates superior stability, achieving a low 14.3% average gibberish rate of 0% on PubMedQA and 5.8% on MedQA. This starkly contrasts with baseline methods like 82.3% on TIES and 79.5% on DARE TIES AIM, which suffer from significant output collapse. This confirms RCP-Merging's robust performance stems from genuine capability integration.

Different Domain-specific Task. To verify generalizability, we shifted the domain from BioMedicine to Finance, merging the WiroAI-Finance-Qwen-7B model with the same Base and Reasoning models. Shown in Table 2, RCP-Merging achieved the highest average score of 72.2, outperforming baselines across all benchmarks: ConvFinQA (Finance) (Cheng, Huang, and Wei 2024), GSM8k (Math), HumanEval (Code), and ARC-C (Knowledge). This result confirms our method's cross-domain scalability and its ability to balance domain knowledge with long-CoT capabilities.

Method/Task	Math	Code	Fin.	Know.	Avg.
Base	69.4	50.6	50.3	60.9	57.8
Finance	50.2	1.2	58.7	47.9	39.5
Reasoning	86.7	76.8	36.2	76.5	69.1
Linear	16.6	32.3	34.0	27.7	27.7
Task Arithmetic	8.4	39.6	17.4	43.3	27.2
TIES-Merging	7.2	21.3	18.8	42.0	22.3
DARE Linear	8.4	49.4	17.7	43.1	29.7
DARE TIES	7.6	38.4	18.4	43.6	27.0
FuseLLM	7.4	53.7	18.4	42.7	30.6
DARE TIES AIM	6.4	18.9	19.7	46.5	22.9
FuseLLM AIM	5.3	26.2	20.4	47.1	24.8
Sens-Merging	60.7	53.7	4.2	25.8	36.1
CAT-Merging	60.7	39.0	10.1	24.8	33.7
RCP-Merging	**82.0**	**71.3**	**59.2**	**76.4**	**72.2**

Table 2: Performance comparison of merging Qwen2.5-7B (Base), WiroAI-Finance-Qwen-7B (Finance) and DeepSeek-R1-Distill-Qwen-7B (Reasoning) on four datasets across Math, Code, Finance and Knowledge areas. Best performance of all merging methods on each dataset is highlighted in **bold**.

Different Model Architecture

RCP-Merging demonstrates consistent performance across different architectures. We have verified this by conducting experiments on the Llama3.1-8B based models, which is distinct from our primary setup. In this alternative configuration, we used Llama3.1-8B (Grattafiori et al. 2024) as the Base model, Llama3-OpenBioLLM-8B (Ankit Pal 2024) as the BioMedicine model, and DeepSeek-R1-Distill-Llama-8B (Guo et al. 2025) as the Reasoning model.

We use GSM8k, HumanEval, ARC-C and PubMedQA to indicate the performance of different merge methods on Math, Code, Knowledge and BioMedicine domain. As the results in Table 3, RCP-Merging achieves the best average score of 68.3 among all merging methods. Although FuseLLM AIM shows a slightly better score in the specific BioMedicine domain, RCP-Merging has the best overall capability.

Method/Task	Math	Code	BioMed.	Know.	Avg.
Base	60.9	42.7	55.0	60.7	54.8
BioMedicine	39.4	37.8	58.0	56.0	47.8
Reasoning	68.8	89.6	51.5	84.0	73.5
Linear	3.2	37.2	31.0	59.0	32.6
Task Arithmetic	55.3	48.2	23.0	45.9	43.1
TIES-Merging	47.5	40.2	53.5	62.2	50.9
DARE Linear	58.3	40.2	23.0	45.9	41.9
DARE TIES	45.6	47.6	32.5	22.2	37.0
FuseLLM	48.8	61.0	55.5	53.3	54.7
DARE TIES AIM	38.1	49.4	13.0	26.0	31.6
FuseLLM AIM	56.1	59.8	**57.5**	59.3	58.2
Sens-Merging	65.7	46.3	55.5	65.5	58.3
CAT-Merging	62.5	55.5	54.0	64.3	59.1
RCP-Merging	**67.2**	**73.2**	57.0	**75.8**	**68.3**

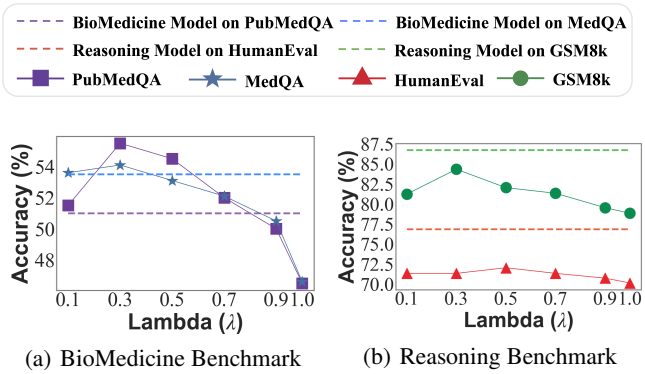
Table 3: Performance comparison of merging Llama-3.1-8B (Base), Llama3-OpenBioLLM-8B (BioMedicine) and DeepSeek-R1-Distill-Llama-8B (Reasoning) on four datasets across Math, Code, BioMedicine, and Knowledge areas. Best performance of all merging methods on each dataset is highlighted in **bold**.

Hypterparameter Analysis

We performed a hyperparameter search for the reasoning-preserving coefficient λ on the Qwen2.5-7B architecture to balance domain-specific knowledge integration with foundational reasoning. As detailed in Figure 3, our results identify $\lambda = 0.3$ as optimal for the 7B model. This value yields peak performance on the BioMedicine Benchmark with 55.5% on PubMedQA and 54.2% on MedQA in Figure 4(a) while maintaining high accuracy on reasoning tasks like GSM8k and HumanEval in Figure 4(b).

Ablation Study

This section performs an ablation study to evaluate the effectiveness of the parameter-specific trimming techniques in RCP-Merging, including the pruning of Knowledge Sensitivity and Reasoning Preservation. As results shown in Table 4, excluding Domain Sensitivity (w/o Domain Sensitivity) causes the average score to drop significantly from 68.3 to 48.7. The effect is even more severe when removing the Reasoning Preservation (w/o Reasoning Preservation), which plunges the average score to 41.4. These results underscore that both trimming strategies are indispensable.



(a) BioMedicine Benchmark (b) Reasoning Benchmark

Figure 4: Hyperparameter Analysis. Performance comparison when merging Qwen2.5-7B (Base), Meditron3-Qwen2.5-7B (BioMedicine), and DeepSeek-R1-Distill-Qwen-7B (Reasoning) on BioMedicine (Figure 4(a)) and Reasoning (Figure 4(b)) benchmarks using different reasoning-preserving coefficients λ.

Method/Task	Math	Code	BioMed.	Know.	Avg.
Base	60.9	42.7	55.0	60.7	54.8
BioMedicine	39.4	37.8	58.0	56.0	47.8
Reasoning	68.8	89.6	51.5	84.0	73.5
w/o Domain	58.4	56.1	33.0	47.4	48.7
w/o Reason.	57.1	37.2	30.5	40.9	41.4
RCP-Merging	**67.2**	**73.2**	**57.0**	**75.8**	**68.3**

Table 4: Ablation Study. Performance comparison when merging Qwen2.5-7B (Base), Meditron3-Qwen2.5-7B (BioMedicine) and DeepSeek-R1-Distill-Qwen-7B (Reasoning) on four datasets across Math, Code, BioMedicine, and Knowledge areas. The best performance under certain dataset is highlighted in **bold**.

Conclusion

We propose a novel model merging framework, RCP-Merging, which effectively integrates domain-specific models with long-chain-of-thought reasoning models by treating reasoning ability as a prior. Our method applies a reasoning capability penalty to preserve core reasoning parameters while selectively merging essential domain-specific weights. Notably, RCP-Merging enhances performance in the BioMedicine and Finance domains by 9.5% and 9.2% respectively, compared to state-of-the-art methods. Our approach creates powerful, unified models that excel in both domain-specific knowledge and general long-chain-of-thought reasoning, effectively addressing the challenge of balancing domain performance with reasoning capability.

Acknowledgement

This work was supported by National Natural Science Foundation of China (62406114,62472181,62306117), the Fundamental Research Funds for the Central Universities (2024ZYGXZR074), Guangdong Basic and Applied Basic Research Foundation (2025A1515011413,2024A04J3681,2024A1515010220), and GJYC program of Guangzhou (2024D03J0005), National Key R & D Project from Minister of Science and Technology (2024YFA1211500), and CCF-Baidu Open Fund.

References

Ankit Pal, M. S. 2024. OpenBioLLMs: Advancing Open-Source Large Language Models for Healthcare and Life Sciences. https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; Pinto, H. P. D. O.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; et al. 2021. Evaluating Large Language Models Trained on Code. *CoRR*, abs/2107.03374.

Chen, Z.; Hernández-Cano, A.; Romanou, A.; Bonnet, A.; Matoba, K.; Salvi, F.; Pagliardini, M.; Fan, S.; Köpf, A.; Mohtashami, A.; Sallinen, A.; Sakhaeirad, A.; Swamy, V.; Krawczuk, I.; Bayazit, D.; Marmet, A.; Montariol, S.; Hartley, M.; Jaggi, M.; and Bosselut, A. 2023. MEDITRON-70B: Scaling Medical Pretraining for Large Language Models. *CoRR*, abs/2311.16079.

Cheng, D.; Huang, S.; and Wei, F. 2024. Adapting Large Language Models via Reading Comprehension. In *The Twelfth International Conference on Learning Representations*.

Clark, P.; Cowhey, I.; Etzioni, O.; Khot, T.; Sabharwal, A.; Schoenick, C.; and Tafjord, O. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *CoRR*, abs/1803.05457.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. *CoRR*, abs/2110.14168.

Cui, H.; Shamsi, Z.; Cheon, G.; Ma, X.; Li, S.; Tikhanovskaya, M.; Norgaard, P. C.; Mudur, N.; Plomecka, M. B.; Raccuglia, P.; Bahri, Y.; Albert, V. V.; Srinivasan, P.; Pan, H.; Faist, P.; Rohr, B. A.; Statt, M. J.; Morris, D.; Purves, D.; Kleeman, E.; Alcantara, R.; Abraham, M.; Mohammad, M.; VanLee, E. P.; Jiang, C.; Dorfman, E.; Kim, E.-A.; Brenner, M.; Ponda, S. S.; and Venugopalan, S. 2025. CURIE: Evaluating LLMs on Multitask Scientific Long-Context Understanding and Reasoning. In *The Thirteenth International Conference on Learning Representations*.

Dong, Z.; Li, J.; Jiang, J.; Xu, M.; Zhao, W. X.; Wang, B.; and Chen, W. 2025. LongReD: Mitigating Short-Text Degradation of Long-Context Large Language Models via Restoration Distillation. *CoRR*, abs/2502.07365.

Du, G.; Lee, J.; Li, J.; Jiang, R.; Guo, Y.; Yu, S.; Liu, H.; Goh, S. K.; Tang, H.; He, D.; and Zhang, M. 2024. Parameter Competition Balancing for Model Merging. In Globersons, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J. M.; and Zhang, C., eds., *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Fisher, R. A. 1925. Theory of statistical estimation. In *Mathematical proceedings of the Cambridge philosophical society*, volume 22, 700–725. Cambridge University Press.

Goddard, C.; Siriwardhana, S.; Ehghaghi, M.; Meyers, L.; Karpukhin, V.; Benedict, B.; McQuade, M.; and Solawetz, J. 2024a. Arcee's MergeKit: A Toolkit for Merging Large Language Models. In Dernoncourt, F.; Preoctiuc-Pietro, D.; and Shimorina, A., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 477–485. Miami, Florida, US: Association for Computational Linguistics.

Goddard, C.; Siriwardhana, S.; Ehghaghi, M.; Meyers, L.; Karpukhin, V.; Benedict, B.; McQuade, M.; and Solawetz, J. 2024b. Arcee's MergeKit: A Toolkit for Merging Large Language Models. In Dernoncourt, F.; Preoctiuc-Pietro, D.; and Shimorina, A., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 477–485. Miami, Florida, US: Association for Computational Linguistics.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The Llama 3 Herd of Models. *CoRR*, abs/2407.21783.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Hao, S.; Sukhbaatar, S.; Su, D.; Li, X.; Hu, Z.; Weston, J. E.; and Tian, Y. 2025. Training Large Language Model to Reason in a Continuous Latent Space.

Ilharco, G.; Ribeiro, M. T.; Wortsman, M.; Schmidt, L.; Hajishirzi, H.; and Farhadi, A. 2023. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Izmailov, P.; Podoprikhin, D.; Garipov, T.; Vetrov, D. P.; and Wilson, A. G. 2018. Averaging Weights Leads to Wider Optima and Better Generalization. In Globerson, A.; and Silva, R., eds., *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*, 876–885. AUAI Press.

Jaech, A.; Kalai, A.; Lerer, A.; Richardson, A.; El-Kishky, A.; Low, A.; Helyar, A.; Madry, A.; Beutel, A.; Carney, A.; et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Jain, N.; Han, K.; Gu, A.; Li, W.-D.; Yan, F.; Zhang, T.; Wang, S. I.; Solar-Lezama, A.; Sen, K.; and Stoica, I. 2024. LiveCodeBench: Holistic and Contamination Free Evaluation of Large Language Models for Code. *arXiv preprint arXiv:2403.07974*.

Jin, D.; Pan, E.; Oufattole, N.; Weng, W.; Fang, H.; and Szolovits, P. 2020. What Disease does this Patient Have?

A Large-scale Open Domain Question Answering Dataset from Medical Exams. *CoRR*, abs/2009.13081.

Jin, Q.; Dhingra, B.; Liu, Z.; Cohen, W. W.; and Lu, X. 2019. PubMedQA: A Dataset for Biomedical Research Question Answering. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, 2567–2577. Association for Computational Linguistics.

Jin, X.; Ren, X.; Preotiuc-Pietro, D.; and Cheng, P. 2023. Dataless Knowledge Fusion by Merging Weights of Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N. C.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; Hassabis, D.; Clopath, C.; Kumaran, D.; and Hadsell, R. 2016. Overcoming catastrophic forgetting in neural networks. *CoRR*, abs/1612.00796.

Li, W.; Peng, Y.; Zhang, M.; Ding, L.; Hu, H.; and Shen, L. 2023. Deep Model Fusion: A Survey. *CoRR*, abs/2309.15698.

Liu, S.; Wu, H.; He, B.; Han, X.; Yuan, M.; and Song, L. 2025. Sens-Merging: Sensitivity-Guided Parameter Balancing for Merging Large Language Models. *CoRR*, abs/2502.12420.

Liu, Y.; Liu, Y.; Chen, X.; Chen, P.-Y.; Zan, D.; Kan, M.-Y.; and Ho, T.-Y. 2024. The devil is in the neurons: Interpreting and mitigating social biases in language models. In *The Twelfth International Conference on Learning Representations*.

Ma, Q.; Liu, D.; Qian, C.; Zhang, L.; and Shao, J. 2025. LED-Merging: Mitigating Safety-Utility Conflicts in Model Merging with Location-Election-Disjoint. *CoRR*.

Maini, P.; Mozer, M. C.; Sedghi, H.; Lipton, Z. C.; Kolter, J. Z.; and Zhang, C. 2023. Can Neural Network Memorization Be Localized? In *ICML*.

Matena, M.; and Raffel, C. 2022. Merging Models with Fisher-Weighted Averaging. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Michel, P.; Levy, O.; and Neubig, G. 2019. Are sixteen heads really better than one? *Advances in neural information processing systems*, 32.

Nobari, A. H.; Alimohammadi, K.; ArjomandBigdeli, A.; Srivastava, A.; Ahmed, F.; and Azizan, N. 2025. Activation-Informed Merging of Large Language Models. *CoRR*, abs/2502.02421.

OpenAI. 2023. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774*. Accessed: 2025-12-05.

OpenAI. 2025. OpenAI o3 and o4-mini System Card. Accessed: 2025-12-05.

Qwen, T. 2024. Qwen2.5 Technical Report. *CoRR*, abs/2412.15115.

Rein, D.; Hou, B. L.; Stickland, A. C.; Petty, J.; Pang, R. Y.; Dirani, J.; Michael, J.; and Bowman, S. R. 2023. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. *arXiv preprint arXiv:2311.12022*.

Ruan, W.; Yang, T.; Zhou, Y.; Liu, T.; and Lu, J. 2025. From Task-Specific Models to Unified Systems: A Review of Model Merging Approaches. *CoRR*, abs/2503.08998.

Springenberg, J.; Dosovitskiy, A.; Brox, T.; and Riedmiller, M. 2015. Striving for Simplicity: The All Convolutional Net. In *ICLR (workshop track)*.

Sun, W.; Li, Q.; Geng, Y.-a.; and Li, B. 2025a. Cat merging: A training-free approach for resolving conflicts in model merging. *arXiv preprint arXiv:2505.06977*.

Sun, W.; Li, Q.; Wang, W.; ao Geng, Y.; and Li, B. 2025b. Task Arithmetic in Trust Region: A Training-Free Model Merging Approach to Navigate Knowledge Conflicts. *ArXiv*, abs/2501.15065.

Tang, X.; Wang, X.; Lv, Z.; Min, Y.; Zhao, W. X.; Hu, B.; Liu, Z.; and Zhang, Z. 2025. Unlocking General Long Chain-of-Thought Reasoning Capabilities of Large Language Models via Representation Engineering. *CoRR*, abs/2503.11314.

Thapa, R.; Wu, Q.; Wu, K.; Zhang, H.; Zhang, A.; Wu, E.; Ye, H.; Bedi, S.; Aresh, N.; Boen, J.; Reddy, S.; Athiwaratkun, B.; Song, S. L.; and Zou, J. 2025. Disentangling Reasoning and Knowledge in Medical Large Language Models. *ArXiv*, abs/2505.11462.

Veeraboina, H. 2023. AIME Problem Set 1983-2024.

Wan, F.; Huang, X.; Cai, D.; Quan, X.; Bi, W.; and Shi, S. 2024. Knowledge Fusion of Large Language Models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

xAI. 2025. Grok 3.5: Advanced Reasoning AI Model by xAI. https://grok.x.ai/. Accessed: 2025-05-15.

Yadav, P.; Tam, D.; Choshen, L.; Raffel, C.; and Bansal, M. 2023. TIES-Merging: Resolving Interference When Merging Models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Yang, E.; Shen, L.; Guo, G.; Wang, X.; Cao, X.; Zhang, J.; and Tao, D. 2024a. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666*.

Yang, E.; Wang, Z.; Shen, L.; Liu, S.; Guo, G.; Wang, X.; and Tao, D. 2024b. AdaMerging: Adaptive Model Merging for Multi-Task Learning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Yu, L.; Yu, B.; Yu, H.; Huang, F.; and Li, Y. 2024. Language Models are Super Mario: Absorbing Abilities from Homologous Models as a Free Lunch. arXiv:2311.03099.

Zeng, Z.; Cheng, Q.; Yin, Z.; Zhou, Y.; and Qiu, X. 2025. Revisiting the Test-Time Scaling of o1-like Models: Do they Truly Possess Test-Time Scaling Capabilities? *CoRR*, abs/2502.12215.