MetaGDPO: Alleviating Catastrophic Forgetting with Metacognitive Knowledge Through Group Direct Preference Optimization

Lanxue Zhang[1,2*], **Yuqiang Xie**[3*], **Fang Fang**[1,2†], **Fanglong Dong**[1,2], **Rui Liu**[4], **Yanan Cao**[1,2]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] Independent Researcher
[4] JIUTIAN Research, Beijing, China
{zhanglanxue, fangfang0703}@iie.ac.cn

Abstract

Large Language Models demonstrate strong reasoning capabilities, which can be effectively compressed into smaller models. However, existing datasets and fine-tuning approaches still face challenges that lead to catastrophic forgetting, particularly for models smaller than 8B. First, most datasets typically ignore the relationship between training data knowledge and the model's inherent abilities, making it difficult to preserve prior knowledge. Second, conventional training objectives often fail to constrain inherent knowledge preservation, which can result in forgetting of previously learned skills. To address these issues, we propose a comprehensive solution that alleviates catastrophic forgetting from both the data and fine-tuning approach perspectives. On the data side, we construct a dataset of 5K instances that covers multiple reasoning tasks and incorporates metacognitive knowledge, making it more tolerant and effective for distillation into smaller models. We annotate the metacognitive knowledge required to solve each question and filter the data based on task knowledge and the model's inherent skills. On the training side, we introduce GDPO (Group Direction Preference Optimization), which is better suited for resource-limited scenarios and can efficiently approximate the performance of GRPO. Guided by the large model and by implicitly constraining the optimization path through a reference model, GDPO enables more effective knowledge transfer from the large model and constrains excessive parameter drift. Extensive experiments demonstrate that our approach significantly alleviates catastrophic forgetting and improves reasoning performance on smaller models.

Code — https://github.com/Qlanxue/MetaGDPO
Extended version — https://arxiv.org/abs/2511.12113

1 Introduction

Large Language Models (LLMs) have demonstrated strong capabilities, enhanced by advanced reasoning skills that enable them to solve complex tasks (Jaech et al. 2024; Guo et al. 2025; Yang et al. 2025; Anthropic 2025). However, enabling excellent reasoning capabilities typically requires models with a large number of parameters, which presents practical challenges for deployment, particularly in resource-constrained training or inference environments. This highlights the need to compress reasoning capabilities into smaller models with limited resources.

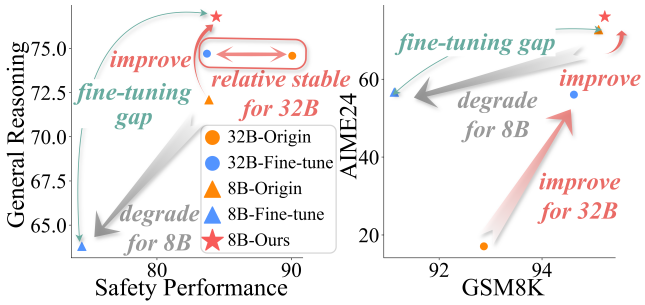

Figure 1: Small-scale fine-tuning on challenging mathematical data such as LIMO can enhance the performance of 32B models, whereas 8B models tend to suffer from severe performance degradation.

Existing studies usually compress reasoning abilities into smaller models through large-scale datasets or small-scale but high-quality data. DeepSeek R1 (Guo et al. 2025) employs large-scale data (800K) to distill the reasoning abilities. In contrast, LIMO (Ye et al. 2025) and s1k (Muennighoff et al. 2025) utilize high-quality and small-scale distilling data to 32B models based on question difficulty. However, compressing reasoning abilities into models smaller than 8B remains challenging, especially as fine-tuning with multi-perspective objectives to support broader applications often leads to performance degradation (Zheng et al. 2025; Arora and Zanette 2025; Wang et al. 2025). As shown in Figure 1, fine-tuning smaller models with small-scale but high-quality data often results in substantial performance degradation across multiple evaluation dimensions, especially on difficult tasks like AIME24. Even when trained on mathematical data, these models still exhibit decreased performance on simple math problems. These degradation are usually caused by the catastrophic forgetting during fine-tuning. Therefore, we desire to discuss one research question in this paper: ***How to improve the model's reasoning ability while decreasing the catastrophic forgetting under limited***

*These authors contributed equally.
†Corresponding author.

resources?

According to previous studies, it is effective to alleviate this issue from ***data*** (Resta and Bacciu 2024; Rebuffi et al. 2017) and ***fine-tuning*** (Hu et al. 2022; Houlsby et al. 2019; Zheng et al. 2025) perspective. From the ***data*** side, previous methods usually utilize difficulty-based dataset collection or experience replay. However, datasets (Ye et al. 2025; Muennighoff et al. 2025) using difficulty as the selection principle ignore the relationship between the data and the base models' inherent knowledge, leading the model to forget simple knowledge for difficult data learning. Experience replay (Resta and Bacciu 2024) involves constructing appropriate datasets aligned with the training set, which can be challenging when training data is unknown (Xiao et al. 2024). From the ***fine-tuning*** perspective, previous researches usually choose to freeze part of the models' parameters to preserve existing experience, like PEFT (Houlsby et al. 2019), LoRA (Hu et al. 2022), and freezing layers (Zheng et al. 2025). They focus on learning new tasks while ignoring constraints from prior knowledge, which easily leads to forgetting during training. Although GRPO (Shao et al. 2024) demonstrates strong performance in reasoning tasks with parameter constraints, it requires more resources for online sampling and model-based reward calculation.

To solve the above issues, we propose METAGDPO to improve models' entire performance during training from the data and fine-tuning perspective.

To enhance the relationship between training ***data*** and models' inherent capabilities, we propose a knowledge-based data construction approach that leverages metacognitive knowledge as the selection principle, referring to the learner's accumulated understanding of specific knowledge types (Didolkar et al. 2024). To support comprehensive analysis, we collect different reasoning tasks that occurred in real applications. We label the metacognitive knowledge required to solve each question and analyze the base models' performance across different types of knowledge. Based on this analysis, we retain complex questions that combine multiple knowledge units and select representative questions for each knowledge unit according to the base models' proficiency. For knowledge that the model is already proficient in, we can retain only a small number of instances to serve as a reminder to keep the inherent abilities. Finally, we obtain METAKL with 5K training questions covering a wider knowledge range. This data collection method includes not only new knowledge that the model lacks but also previously acquired knowledge, thereby reinforcing existing capabilities and mitigating forgetting.

To enhance the improvement during ***fine-tuning*** progress, we propose Group Direct Preference Optimization (GDPO), which enables base models to learn group-wise response distributions guided by advantages derived from high-quality response groups generated by the capable model. By preserving constraints on prior knowledge, the model can alleviate catastrophic forgetting. Specifically, we sample a group of responses from the strong model and compute their corresponding advantages. The small model is then updated based on the preferences of nearby responses within the sorted group, which reduces the inter-group preference computation from $\mathcal{O}(G^2)$ to $\mathcal{O}(G)$. This method enables the small model to learn the response distribution along with the corresponding preferences. Besides, GDPO is suitable for source-limited training scenarios compared with GRPO. As demonstrated in Figure 1, our approach effectively enhances the performance of smaller models across multiple dimensions.

To further demonstrate the effectiveness of our method, we provide a detailed analysis with proof and conduct thorough experiments across 12 benchmarks and different training methods. Experimental results demonstrating METAGDPO can further improve the model's performance with little disturbance to inherent reasoning abilities.

Our contribution can be summarized as follows:

- We first introduce the METAKL, a dataset that provides diverse reasoning tasks from a metacognitive knowledge perspective to associate training data with models' inherent knowledge.
- We propose the Group Direct Preference Optimization to improve the performance of models while alleviating the catastrophic forgetting of efficient models.
- We conduct extensive experiments to demonstrate the effectiveness of METAGDPO.

2 Related Works

Unlike traditional large language models (LLMs) that prioritize immediate answer generation, large reasoning models (LRMs) distinguish themselves by leveraging a long chain-of-thought to solve complex tasks with thorough thinking. Since the release of OpenAI's o1 series in late 2024 (Jaech et al. 2024), which marked a significant leap in AI reasoning, several advanced LRMs have emerged. OpenAI's o3 series (OpenAI 2025) demonstrated doctoral-level problem-solving across disciplines, while Google's Gemini 2.5 Pro (**?**) and XAI's Grok-3 (xAI 2025) further pushed state-of-the-art performance in reasoning-intensive tasks. Anthropic's Claude 3.7 Sonnet (Anthropic 2025) combines dual reasoning modes to excel in code generation and multi-step logic. In the open-source domain, DeepSeek-R1 (Guo et al. 2025) and Qwen3 (Yang et al. 2025) represent notable advances, with Qwen3 pioneering a hybrid fast/slow mode architecture that rivals closed-source systems on challenging benchmarks. These developments highlight the rapid evolution of LRMs and their growing impact on real-world complex reasoning tasks.

There are some studies devoted to compressing large models' reasoning abilities into smaller models. Some studies collect LLMs' distillation datasets like LaMini (Wu et al. 2024), UltraChat (Ding et al. 2023), NuminaMath (Li et al. 2024), and SYNTHETIC-1 (Mattern et al. 2025), focusing on providing large-scale datasets to cover diverse instructions. These datasets usually need more resources to fine-tune the LLM. Therefore, there are also works advocating for utilizing small-scale high-quality datasets to enhance the LLMs' performance. STILL-2 (Min et al. 2024) constructed a 4.9k-sample long-form reasoning dataset through DeepSeek-R1 and QwQ (Qwen Team 2025). O1-journey-part2 (Huang et al. 2024) involved the direct utilization of

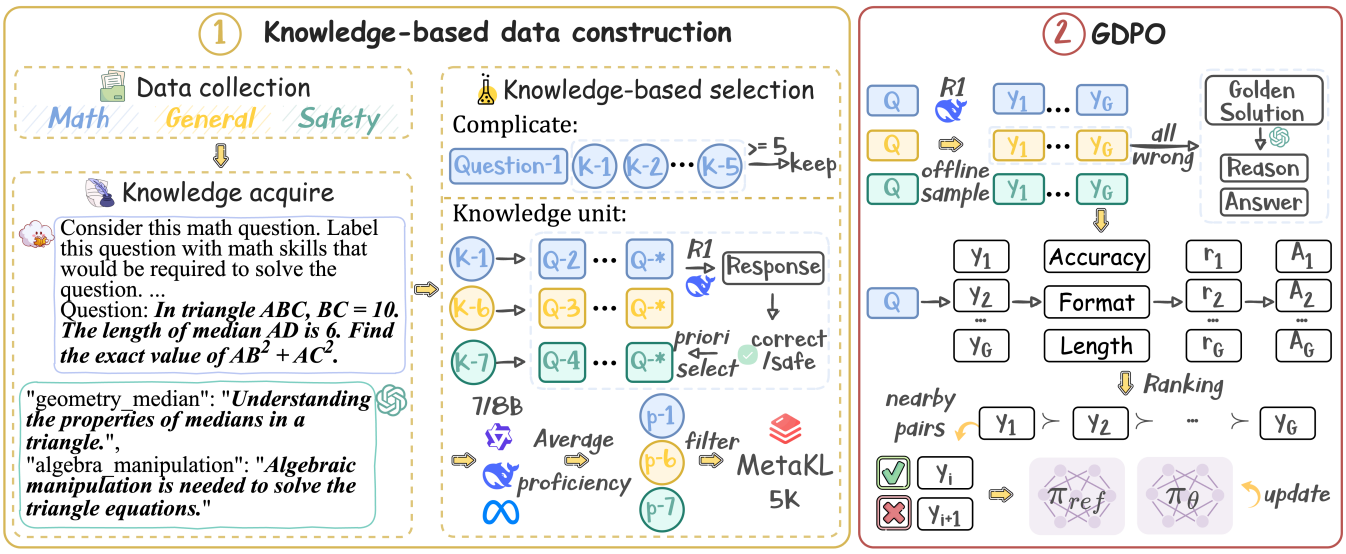

Figure 2: The framework of our paper. We first construct the data based on metacognitive knowledge with analyzing base models' inherent ability. Then, we utilize the Group Direct Preference Optimization to fine-tune the base model.

the OpenAI O1's API to synthesize lengthy thought chains. LIMO (Ye et al. 2025) and s1k (Muennighoff et al. 2025) identify 817 and 1K datasets of high-quality mathematical reasoning datasets. STAR-1 (Wang et al. 2025) filters out 1K to improve the safeguard ability of reasoning models. However, these works usually adopt supervised fine-tuning based on difficult datasets, which leads to the catastrophic forgetting phenomenon.

3 Method

To comprehensively alleviate catastrophic forgetting during distillation in small LLMs, we design our approach from both the data and training perspectives. On the data side, we collect training instances based on the metacognitive knowledge required to solve each question and the model's inherent knowledge. This helps align the small model's metacognitive understanding with that of the large model. On the training side, we propose Group Direction Preference Optimization (GDPO), which further enhances model performance while preserving as much inherent knowledge as possible.

3.1 Knowledge-based Data Construction

Data Collection To support real-world applications from multiple perspectives, we collect existing training datasets targeting mathematical reasoning (NuminaMath-CoT (Li et al. 2024)), non-mathematical / general reasoning (MMLU (Hendrycks et al. 2021a), CommonsenseQA (Talmor et al. 2019), CommonsenseQA 2.0 (Talmor et al. 2022), LogiQA (Liu et al. 2023)), and safety-related tasks. Detailed information on these datasets is provided in the Appendix, which can be found in our extended version. To reduce the number of similar prompts, we first perform coarse filtering following the steps in STAR-1 (Wang et al. 2025), including n-gram filtering, TF-IDF similarity filtering, and semantic embedding similarity filtering. To ensure fair evaluation, we strictly avoid using any data similar to the evaluation benchmarks and only adopt existing training datasets. After this process, we obtain 38,838 data instances and carefully check for any overlap between these prompts and the benchmarks used in our experiments, removing any duplicates to prevent data leakage.

Metacognitive knowledge acquire To determine the metacognitive knowledge required to solve each question and to better control the scope of training data, we identify the relevant metacognitive knowledge for each question using a similar approach to (Didolkar et al. 2024). We first instruct GPT-4o to extract the knowledge needed for problem-solving. Then, we cluster the knowledge names to group similar knowledge together, which reduces redundancy and enables more precise filtering. After clustering, we obtain 8,325 knowledge. The detailed prompts used for this process are provided in the Appendix. To demonstrate the reasonability of the metacognitive knowledge, we conducted human annotation. Specifically, we randomly sample 500 prompts along with their extracted knowledge and engage five expert annotators to evaluate the consistency between the knowledge and the corresponding questions. Each annotation is paid $0.20 per instance. The average consistency score reaches 92.18%, demonstrating the reliability of the generated knowledge.

Knowledge-based selection In order to make the selection process more tight with the base model itself, we first analyze the performance of small commonly used LLMs including Qwen2.5-7B-Instruct, LLaMA3.1-8B, DeepSeek-R1-Qwen-7B, DeepSeek-R1-LLaMA-8B, and Qwen3-8B. Figure 3 illustrates the performance of each model across the skill numbers. We can observe that with the increase in skill combined numbers, the models' performance reduced.

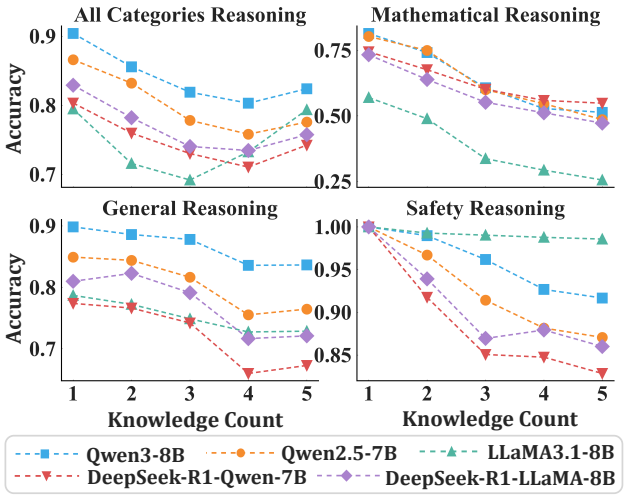

Figure 3: The performance on metacognitive knowledge of base models.

Considering both the difficulty of the dataset and the model's inherent ability, we first reserve all complicated instances, solving which requires more than 5 skills. Then, we consider the base model's skill accuracy as a skill filter ratio and select based on the knowledge unit. To ensure the skill while reducing the training instance numbers, we adopt a greedy selection strategy. We first keep 20 questions with a priori correct and safe responses for each knowledge unit. Then we select a question based on average proficiency, as it is more relevant to all models' knowledge levels. We prioritize keeping the questions more adequate. The pseudo-algorithm can be found in the Appendix.

3.2 Group Direct Preference Optimization

While GRPO demonstrates strong performance in reasoning training, its reliance on online sampling incurs substantial resource costs and can lead to uncontrollable exploration, particularly when the model's initial capabilities are limited. Therefore, we propose Group Direct Preference Optimization (GDPO), which adapts the distillation process by leveraging high-quality responses from the capable model.

Given a question q, we generate a group of responses $\{r_1, r_2, \ldots, r_G\}$ from large model. The objective is to obtain a policy model π_θ derived from a reference model π_{ref}, based on the group advantages of the responses.

We first derive the GDPO objective similar to DPO (Rafailov et al. 2023). We retain rule-based advantage weights and derive the optimal solution π_θ from the reinforcement learning objective, following the prior works (Schulman et al. 2017; Shao et al. 2024):

$$\pi_\theta = \frac{1}{Z(q)} \pi_{ref}(y_i|q) \exp\left(\frac{A_i}{\beta} r(q, y_i)\right), \tag{1}$$

where $Z(q) = \sum_{y_i} \pi_{ref}(y_i|q) \exp\left(\frac{A_i}{\beta} r(q, y_i)\right)$ is the partition function, which is independent from π_θ. The ground-truth reward can then be expressed as $r^* = \frac{\beta}{A_i} \log \frac{\pi^*(y_i|q)}{\pi_{ref}(y_i|q)} + \frac{\beta}{A_i} \log Z(q)$.

Based on the Bradley–Terry model $p^*(yi \succ y_j) = \frac{\exp(r(q,y_i))}{\exp(r(q,y_i)) + \exp(r(q,y_j))}$ (Bradley and Terry 1952), we further extend this formulation to incorporate pairwise comparisons within each response group. Then, we can obtain the following optimization loss:

$$\mathcal{L}_{\text{GDPO}}(\theta) = -\frac{1}{G(G-1)} \sum_{i=1}^{G} \sum_{j \neq i} \sigma\Big(\frac{\beta}{A_i} \log \frac{\pi_\theta(y_i|q)}{\pi_{ref}(y_i|q)} - \frac{\beta}{A_{i+1}} \log \frac{\pi_\theta(y_{i+1}|q)}{\pi_{ref}(y_{i+1}|q)} + f(Z(q)) \Big), \tag{2}$$

where $f(Z(q)) = \beta \log Z(q)(\frac{1}{A_i} - \frac{1}{A_j})$. Based on rigorous proof in the Appendix, we can ignore $f(Z(q))$ without disturbing the convergence progress, which is irrelevant to π_θ.

To further reduce the computation, we optimize the loss to the nearby pairs chain, then the optimization objective can be transferred as follows:

$$\widetilde{\mathcal{L}}_{\text{approx}}(\theta) = -\frac{1}{G-1} \sum_{i=1}^{G-1} \sigma\Big(\frac{\beta}{A_i} \log \frac{\pi_\theta(y_i|q)}{\pi_{ref}(y_i|q)} - \frac{\beta}{A_j} \log \frac{\pi_\theta(y_j|q)}{\pi_{ref}(y_j|q)} \Big). \tag{3}$$

To determine how to choose G to minimize the error of the above calculation optimization, we derive the estimated error of the gradient as follows:

$$\varepsilon \leq (\mu_{adj} - \mu'_{adj})^2 + \frac{\text{Var}(\mu'_{\text{adj}})}{G-1}, \tag{4}$$

where $\mu_{adj} = \mathbb{E}[\sigma(\Delta \widetilde{r}_{i,i+1})]$ is the ideal expectation of nearby pairs and the μ'_{adj} is the expectation of sampling G responses. When $G >= 10$, the relative error of the gradient is lower than 10% compared with $G = 2$. The corresponding proof can be found in the Appendix.

Therefore, we offline generate $G = 10$ responses for each question to ensure both efficiency and effectiveness. If all responses are incorrect or unsafe, we use GPT-4o to reconstruct the reasoning and answer based on the golden solution. To calculate the advantages A_i, we adopt three reward functions: accuracy, format, and length, with weights of 1, 0.5, and 0.5, respectively, indicating that correctness is prioritized. When all responses are correct, we prefer shorter responses. To prevent long reasoning chains from dominating the reward, we normalize the response length within each group to calculate the reward as: $A_i^l = 1 - \frac{l_i - min(\{l_1, l_2, \ldots, l_G\})}{max(\{l_1, l_2, \ldots, l_G\}) - min(\{l_1, l_2, \ldots, l_G\})}$. Additionally, we manually check each response to account for discrepancies in output format that may not exactly match the standard answers, ensuring the accuracy and quality of the training data.

To further support the feasibility of GDPO, we provide a detailed proof showing that its derivation and optimization process can approximate the loss of GRPO, along with a comprehensive comparison to GRPO, DPO, and SFT.

Model		AIME24	AMC	MATH500	GSM8K	Olympiad	Minerva	AVG	Overall AVG
Large Model									
DeepSeek-R1-0528		82.92	98.26	95.0	95.28	69.19	52.21	82.15	81.91
7/8B Models									
Qwen3	Origin	72.71	95.16	93.8	95.10	64.89	53.31	79.16	78.98
	LIMO	56.67	89.06	87.2	91.13	52.44	47.06	70.59	70.20
	STAR-1	72.71	93.91	92.6	74.45	64.15	48.90	74.45	73.59
	L+S	69.17	92.34	92.8	93.48	62.67	52.21	77.11	75.15
	METAGDPO	**76.04**↑	**94.69**	**94.0**↑	**95.22**↑	**64.30**	**56.25**↑	**80.08**↑	**80.86**↑
R1-Qwen	Origin	56.46	88.44	86.2	89.69	48.74	41.91	68.57	59.52
	LIMO	42.71	83.44	83.4	84.46	37.19	34.56	60.96	54.42
	STAR-1	**54.79**	89.84↑	86.2	89.39	48.15	43.01↑	68.56	**72.68**↑
	L+S	52.92	**90.16**↑	84.4	46.47	**53.48**↑	36.03	60.58	64.23↑
	METAGDPO	53.54	89.53↑	**88.0**↑	**89.46**	48.39	**45.22**↑	**69.02**↑	60.14↑
R1-LLaMA	Origin	43.75	86.72	79.2	77.94	41.33	31.25	60.03	55.37
	LIMO	33.96	80.47	80.8↑	67.10	46.96↑	27.57	56.14	54.18
	STAR-1	41.46	83.44	79.0	66.49	45.63↑	26.84	57.14	**66.96**↑
	L+S	**51.25**↑	86.88↑	**86.4**↑	61.11	**50.37**↑	**34.93**↑	**61.82**↑	66.12↑
	METAGDPO	43.54	**89.53**↑	79.2	**76.88**	41.48↑	29.78	60.07↑	59.39↑

Table 1: Evaluation results on mathematical benchmarks. The bold results denote the best results across different fine-tune baselines. The uparrow denotes the result improved compared with the original model without finetuning. Overall AVG denotes the overall performance of the models, deriving from the average score of all benchmarks.

4 Experimental Results

4.1 Experimental Settings

Datasets In this paper, we utilize mathematical reasoning, commonsense reasoning and safety benchmarks to comprehensively analysis the performance on multiple application perspectives:

• **Mathematical reasoning:** We first adopt various mathematical reasoning datasets to assess models' reasoning capabilities, including the American Invitational Mathematics Examination (**AIME24**) (MAA 2024), **MATH-500** (Hendrycks et al. 2021b), the American Mathematics Competitions (**AMC23**), **GSM8k** (Cobbe et al. 2021), **OlympiadBench** (He et al. 2024), and **Miverva** (Lewkowycz et al. 2022).

• **General reasoning:** Besides, we use **MMLU** (Hendrycks et al. 2021a), **Commonsense QA** (Talmor et al. 2019), and **GPQA** (Rein et al. 2024) to reflect the non-mathematical reasoning abilities of LRMs.

• **Safety evaluation:** We utilize three safety benchmarks to reflect the safety level of models: **TrustLLM**(Sun et al. 2024), **StrongReject**(Souly et al. 2024), and **WildJailbreak**(Jiang et al. 2024). We utilize the LLaMA-GUARD-3-8B (Inan et al. 2023) to judge the safety of the responses, and report the safety ratio. High score denotes the model safer.

For the dataset lower than 50 questions (**AIME24** and **AMC23**), we adopt generating 16 samples with a temperature setting of 0.7 and calculating the unbiased *pass@1* metric as introduced in (Chen et al. 2021). To ensure the comparison fairness, we utilize the same sample parameters and set the temperature as 0 to provide the evaluation results for other datasets. To analyze overall model performance, we compute the average score to represent its overall capability.

Baselines To assess the generalizability across different model architectures, we fine-tune five base models: Qwen3-8B, DeepSeek-R1-Qwen-7B, and DeepSeek-R1-LLaMA-8B. We also provide the analysis on Qwen2.5-Instruct-7B and LLaMA3.1-Instruct-8B in the Appendix due to space limitation. Furthermore, to investigate the influence of data composition on training dynamics, we conduct comparative experiments using LIMO, STAR-1, and their mixture (denoted as L+S for simplicity).

4.2 Results

Tables 1 and 2 report the performance of the teacher model and 7B/8B-scale student models. The results demonstrate that our proposed framework substantially preserves the model's original reasoning capabilities while achieving consistent improvements in overall performance. Compared to conventional fine-tuning on existing datasets, our method improves relative gains of approximately 5–10% on overall performance. Besides, we observe that directly fine-tuning on mathematically challenging datasets may induce catastrophic forgetting on simpler tasks. For instance, training with L+S leads to a 40% performance degradation on GSM8K for DeepSeek-R1-Qwen and over 50% on Qwen3. In contrast, training with METAGDPO effectively mitigates this degradation and better maintains generalization across different task complexities. For all models, we can improve the general abilities stably. While STAR-1 contributes significantly to enhancing safety alignment, it adversely affects the model's general utility, often resulting in over-

Model		MMLU	CQA	GPQA	AVG	TrustLLM Misuse	TrustLLM Jailbreak	Strong Reject	Wild Jailbreak	AVG
Large Model										
DeepSeek-R1-0528		83.36	79.70	84.26	82.44	95.61	53.50	98.40	77.15	81.17
7/8B Models										
Qwen3	Origin	79.28	77.89	59.09	72.09	92.98	83.50	94.89	64.12	83.87
	LIMO	78.20	78.87↑	34.34	63.80	85.25	77.07	77.32	58.01	74.41
	STAR-1	46.53	30.38	47.47	41.46	**99.91**↑	98.21↑	**100.0**↑	87.42↑	96.39↑
	L+S	55.73	27.27	42.93	41.98	99.65↑	**99.57**↑	99.68↑	**89.46**↑	**97.09**↑
	MetaGDPO	**83.37**↑	**84.11**↑	**62.12**↑	**76.79**↑	92.89	84.79↑	95.85↑	64.12	84.41↑
R1-Qwen	Origin	53.85	65.52	35.86	51.74	65.86	55.84	33.55	51.76	51.75
	LIMO	47.80	59.54	29.80	45.71	58.47	62.21↑	35.14↑	48.69	51.13
	STAR-1	54.20↑	58.31	37.88↑	50.13	**99.30**↑	**99.68**↑	98.36↑	**85.70**↑	**95.76**↑
	L+S	30.85	18.67	**42.42**↑	30.65	99.21↑	**99.43**↑	**99.04**↑	81.86↑	94.89↑
	MetaGDPO	**55.93**↑	**66.01**↑	34.34	**52.09**↑	56.8↑	66.57↑	37.06↑	50.95	52.85↑
R1-LLaMA	Origin	60.47	70.76	29.29	53.51	64.62	66.93	45.05	54.52	57.78
	LIMO	**69.63**↑	51.27	29.29	50.06	66.02↑	69.64↑	42.81	54.30	58.19↑
	STAR-1	50.78	49.14	36.87↑	45.60	99.39↑	**99.68**↑	**99.68**↑	92.08↑	**97.71**↑
	L+S	38.14	20.39	**39.39**↑	32.64	**99.65**↑	98.64↑	**99.68**↑	**92.76**↑	97.68↑
	MetaGDPO	66.88↑	**70.76**	37.37↑	**58.34**↑	65.14↑	67.93↑	49.20↑	54.39	59.17↑

Table 2: Evaluation results on general reasoning and safety benchmarks. The bold results denote the best results across different fine-tune baselines. The uparrow denotes the result improved compared with the original model without finetuning.

conservative behaviors that compromise task completion and practical usability.

Furthermore, we observe that SFT is more effective for safety-oriented learning, while GDPO demonstrates superior performance on reasoning tasks. We hypothesize that improvements in safety may require the model to partially overwrite or forget certain prior harmful thoughts. In contrast, our GDPO framework is designed to leverage the model's initial knowledge, which is more closely related to memory retention.

5 Analysis

5.1 Ablation Study

To locate the effect of advantages in training enhancement, we remove the advantage weight to optimize the model, which can be considered as expanded response pairs for each prompt from DPO. The average results are 85.24, 69.99, 84.27, and 80.25 for math, general, safety, and overall, respectively. We can find that even when removing the advantages during preference learning, increasing the number of responses can still benefit models' performance.

5.2 Scaling Law Analysis

Group number impaction To investigate how the group size impacts the training results, we experiment on $G = 2, 4, 6, 8, 10$ respectively. Figure 4 presents the results of varying the group number. For $G < 10$, we preserve the best response and randomly select the remaining responses. We observe that as the group size increases, overall performance improves. However, when $G < 10$, certain dimensions may slightly degrade, which is consistent with the proof in the Appendix that small group sizes can lead to high training variance.

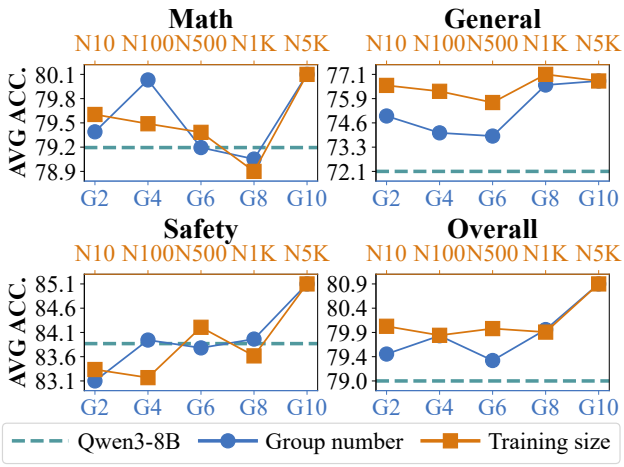

Figure 4: The performance varies with the number of groups and training data size.

Training Size Scaling In this section, we explore the impact of the training data size on training effectiveness. We iterate the training number as $G = [10, 100, 500, 1000, 5000]$. Figure 4 illustrates how performance varies with data size. We observe that using as few as 10 instances can still enhance the model's overall performance. We speculate that this improvement stems from the group-wise distribution of

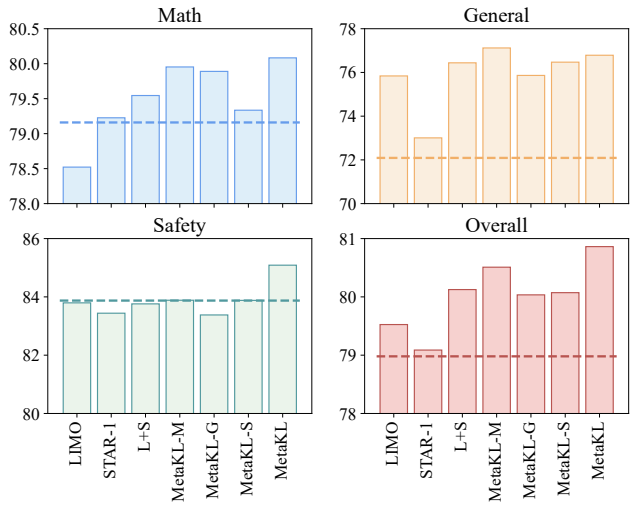

Figure 5: Comparison of model performance across different training data compositions on Qwen3-8B.

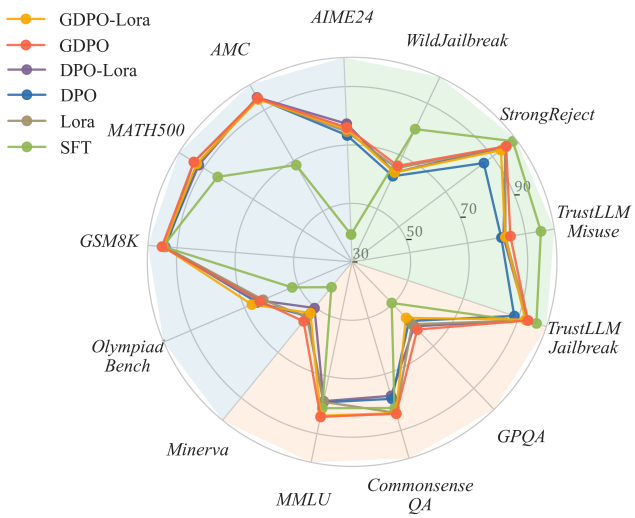

Figure 6: The comparison between different training methods.

responses, which encourages the model to better leverage the potential embedded in its inherent knowledge.

5.3 Data Composition Analysis

To examine the influence of data composition, we train models using different datasets under the GDPO framework, including LIMO, STAR-1, LIMO+STAR-1, METAKL-Math, METAKL-General, and METAKL-Safety. Figure 5 presents the average scores across various categories, with detailed results provided in the Appendix.

We observe that training solely with LIMO slightly reduces the model's mathematical reasoning ability, though the degradation is less severe compared to SFT. In contrast, our METAKL-Math data significantly enhances mathematical reasoning, highlighting the importance of bounding metacognitive learning with training data. While all training configurations improve overall performance, combining LIMO and STAR-1 mitigates performance degradation during GDPO training, suggesting that learning response distributions helps retain general capabilities.

Consequently, our method supports multi-task fine-tuning, even with single-task data. It preserves the model's inherent capabilities, making it possible to enhance performance across multiple application domains.

5.4 Training Method Comparison

To intuitively compare the impact of different training methods on model performance, we take Qwen3-8B as the experimental subject to explore SFT, DPO, and GDPO, along with their LoRA-augmented variants across mathematical reasoning, commonsense reasoning, and safety benchmarks. For DPO, we utilize the best responses as preferred responses while randomly choosing one as the rejected response.

As shown in Figure 6, GDPO exhibits distinct advantages over other training methods. Compared to SFT and DPO, GDPO outperforms SFT by an average of nearly 17% and 3%, respectively, denoting the superiority of learning responses with a large group. In contrast, SFT excels in safety tasks as its supervised training directly aligns with safety-specific labels, while RLHF methods like DPO and GDPO prioritize reasoning enhancement, temporarily sacrificing some safety alignment during optimization.

Therefore, improving performance on general and safety tasks is relatively easier than on mathematical tasks for most methods, suggesting that catastrophic forgetting is more likely to occur in mathematical reasoning. For weaker training methods such as SFT and DPO, applying LoRA proves effective in mitigating this forgetting. In contrast, our method achieves superior performance through full-parameter training.

6 Conclusion

In this paper, we address the catastrophic forgetting during distillation of reasoning abilities from LLMs to smaller models. We solve this issue by improving training data and training methods. From the data perspective, we fully leverage the model's inherent knowledge and minimize catastrophic forgetting. We filter and organize the data based on the metacognitive skills required for each question. This ensures that the learning process remains well-aligned with the model's cognitive abilities and preserves previously acquired knowledge. From the training side, we propose GDPO, which efficiently utilizes the preference characters in group responses of the teacher model, which approximates GRPO and is better adapted for distillation learning. Experimental results demonstrate that our approach effectively supports reasoning generalization while alleviating catastrophic forgetting, providing a valuable resource and experimental guidance for real-world applications of efficient LLMs.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No.U2336202).

References

Anthropic. 2025. Claude 3.7 Sonnet and Claude Code. Accessed: 2025-06-07, https://www.anthropic.com/news/claude-3-7-sonnet.

Arora, D.; and Zanette, A. 2025. Training Language Models to Reason Efficiently. arXiv:2502.04463.

Bradley, R. A.; and Terry, M. E. 1952. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39(3/4): 324–345.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; de Oliveira Pinto, H. P.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; Ray, A.; Puri, R.; Krueger, G.; Petrov, M.; Khlaaf, H.; Sastry, G.; Mishkin, P.; Chan, B.; Gray, S.; Ryder, N.; Pavlov, M.; Power, A.; Kaiser, L.; Bavarian, M.; Winter, C.; Tillet, P.; Such, F. P.; Cummings, D.; Plappert, M.; Chantzis, F.; Barnes, E.; Herbert-Voss, A.; Guss, W. H.; Nichol, A.; Paino, A.; Tezak, N.; Tang, J.; Babuschkin, I.; Balaji, S.; Jain, S.; Saunders, W.; Hesse, C.; Carr, A. N.; Leike, J.; Achiam, J.; Misra, V.; Morikawa, E.; Radford, A.; Knight, M.; Brundage, M.; Murati, M.; Mayer, K.; Welinder, P.; McGrew, B.; Amodei, D.; McCandlish, S.; Sutskever, I.; and Zaremba, W. 2021. Evaluating Large Language Models Trained on Code.

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. *arXiv preprint arXiv:2110.14168*.

Didolkar, A. R.; Goyal, A.; Ke, N. R.; Guo, S.; Valko, M.; Lillicrap, T. P.; Rezende, D. J.; Bengio, Y.; Mozer, M. C.; and Arora, S. 2024. Metacognitive Capabilities of LLMs: An Exploration in Mathematical Problem Solving. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Ding, N.; Chen, Y.; Xu, B.; Qin, Y.; Hu, S.; Liu, Z.; Sun, M.; and Zhou, B. 2023. Enhancing Chat Language Models by Scaling High-quality Instructional Conversations. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 3029–3051. Singapore: Association for Computational Linguistics.

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*.

He, C.; Luo, R.; Bai, Y.; Hu, S.; Thai, Z.; Shen, J.; Hu, J.; Han, X.; Huang, Y.; Zhang, Y.; Liu, J.; Qi, L.; Liu, Z.; and Sun, M. 2024. OlympiadBench: A Challenging Benchmark for Promoting AGI with Olympiad-Level Bilingual Multimodal Scientific Problems. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 3828–3850. Bangkok, Thailand: Association for Computational Linguistics.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2021a. Measuring Massive Multitask Language Understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021b. Measuring Mathematical Problem Solving With the MATH Dataset. *NeurIPS*.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International conference on machine learning*, 2790–2799. PMLR.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Huang, Z.; Zou, H.; Li, X.; Liu, Y.; Zheng, Y.; Chern, E.; Xia, S.; Qin, Y.; Yuan, W.; and Liu, P. 2024. O1 Replication Journey–Part 2: Surpassing O1-preview through Simple Distillation, Big Progress or Bitter Lesson? *arXiv preprint arXiv:2411.16489*.

Inan, H.; Upasani, K.; Chi, J.; Rungta, R.; Iyer, K.; Mao, Y.; Tontchev, M.; Hu, Q.; Fuller, B.; Testuggine, D.; and Khabsa, M. 2023. Llama Guard: LLM-based Input-Output Safeguard for Human-AI Conversations. arXiv:2312.06674.

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*.

Jiang, L.; Rao, K.; Han, S.; Ettinger, A.; Brahman, F.; Kumar, S.; Mireshghallah, N.; Lu, X.; Sap, M.; Choi, Y.; and Dziri, N. 2024. WildTeaming at Scale: From In-the-Wild Jailbreaks to (Adversarially) Safer Language Models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Lewkowycz, A.; Andreassen, A.; Dohan, D.; Dyer, E.; Michalewski, H.; Ramasesh, V.; Slone, A.; Anil, C.; Schlag, I.; Gutman-Solo, T.; et al. 2022. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35: 3843–3857.

Li, J.; Beeching, E.; Tunstall, L.; Lipkin, B.; Soletskyi, R.; Huang, S. C.; Rasul, K.; Yu, L.; Jiang, A.; Shen, Z.; Qin, Z.; Dong, B.; Zhou, L.; Fleureau, Y.; Lample, G.; and Polu, S. 2024. NuminaMath. Accessed: 2025-06-07. Report: https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf.

Liu, H.; Liu, J.; Cui, L.; Teng, Z.; Duan, N.; Zhou, M.; and Zhang, Y. 2023. LogiQA 2.0—An Improved Dataset for Logical Reasoning in Natural Language Understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31: 2947–2962.

MAA. 2024. American Invitational Mathematics Examination - AIME. In *American Invitational Mathematics Examination - AIME 2024*.

Mattern, J.; Jaghouar, S.; Basra, M.; Straube, J.; Ferrante, M. D.; Gabriel, F.; Ong, J. M.; Weisser, V.; and Hagemann, J. 2025. SYNTHETIC-1: Two Million Collaboratively Generated Reasoning Traces from Deepseek-R1.

Min, Y.; Chen, Z.; Jiang, J.; Chen, J.; Deng, J.; Hu, Y.; Tang, Y.; Wang, J.; Cheng, X.; Song, H.; et al. 2024. Imitate, explore, and self-improve: A reproduction report on slow-thinking reasoning systems. *arXiv preprint arXiv:2412.09413*.

Muennighoff, N.; Yang, Z.; Shi, W.; Li, X. L.; Fei-Fei, L.; Hajishirzi, H.; Zettlemoyer, L.; Liang, P.; Candès, E.; and Hashimoto, T. 2025. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*.

OpenAI. 2025. OpenAI o3-mini. Accessed: 2025-06-07.

Qwen Team. 2025. QwQ-32B: Embracing the Power of Reinforcement Learning. Accessed: 2025-06-07, https://qwenlm.github.io/blog/qwq-32b.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Rebuffi, S.-A.; Kolesnikov, A.; Sperl, G.; and Lampert, C. H. 2017. iCaRL: Incremental Classifier and Representation Learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5533–5542.

Rein, D.; Hou, B. L.; Stickland, A. C.; Petty, J.; Pang, R. Y.; Dirani, J.; Michael, J.; and Bowman, S. R. 2024. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Resta, M.; and Bacciu, D. 2024. Self-generated Replay Memories for Continual Neural Machine Translation. In Duh, K.; Gomez, H.; and Bethard, S., eds., *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 175–191. Mexico City, Mexico: Association for Computational Linguistics.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y. K.; Wu, Y.; and Guo, D. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300.

Souly, A.; Lu, Q.; Bowen, D.; Trinh, T.; Hsieh, E.; Pandey, S.; Abbeel, P.; Svegliato, J.; Emmons, S.; Watkins, O.; et al. 2024. A strongreject for empty jailbreaks. *arXiv preprint arXiv:2402.10260*.

Sun, L.; Huang, Y.; Wang, H.; Wu, S.; Zhang, Q.; Gao, C.; Huang, Y.; Lyu, W.; Zhang, Y.; Li, X.; et al. 2024. Trustllm: Trustworthiness in large language models. *arXiv preprint arXiv:2401.05561*, 3.

Talmor, A.; Herzig, J.; Lourie, N.; and Berant, J. 2019. CommonsenseQA: A Question Answering Challenge Targeting Commonsense Knowledge. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4149–4158. Minneapolis, Minnesota: Association for Computational Linguistics.

Talmor, A.; Yoran, O.; Bras, R. L.; Bhagavatula, C.; Goldberg, Y.; Choi, Y.; and Berant, J. 2022. Commonsenseqa 2.0: Exposing the limits of ai through gamification. *arXiv preprint arXiv:2201.05320*.

Wang, Z.; Tu, H.; Wang, Y.; Wu, J.; Mei, J.; Bartoldson, B. R.; Kailkhura, B.; and Xie, C. 2025. STAR-1: Safer Alignment of Reasoning LLMs with 1K Data. arXiv:2504.01903.

Wu, M.; Waheed, A.; Zhang, C.; Abdul-Mageed, M.; and Aji, A. F. 2024. LaMini-LM: A Diverse Herd of Distilled Models from Large-Scale Instructions. In Graham, Y.; and Purver, M., eds., *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, 944–964. St. Julian's, Malta: Association for Computational Linguistics.

xAI. 2025. Grok 3 Beta — The Age of Reasoning Agents. Accessed: 2025-06-07.

Xiao, S.; Liu, Z.; Zhang, P.; and Xing, X. 2024. LM-Cocktail: Resilient Tuning of Language Models via Model Merging. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 2474–2488. Bangkok, Thailand: Association for Computational Linguistics.

Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Gao, C.; Huang, C.; Lv, C.; et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Ye, Y.; Huang, Z.; Xiao, Y.; Chern, E.; Xia, S.; and Liu, P. 2025. LIMO: Less is More for Reasoning. arXiv:2502.03387.

Zheng, J.; Cai, X.; Qiu, S.; and Ma, Q. 2025. Spurious Forgetting in Continual Learning of Language Models. In *The Thirteenth International Conference on Learning Representations*.