Appendix for METAGDPO

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

A Prompts for Data Construction

The prompt template used to obtain metacognitive knowledge is as follows:

Prompt for Metacognitive Knowledge

Consider this mathematical / commonsense / safety question. Label this question with mathematical / commonsense / safety skills that would be required to solve the question. Basically, you should be able to use the fine-grained skill as a dictionary key in Python. The skill name should be in lower case letters only. The skill name should be in lower-case letters only. The skill name should be very descriptive, and you may use multiple words to describe the skills required in the question. If you do use multiple words per question, then join them by an underscore. You can provide multiple skills for complex questions.

Question: **[Question]**

Your answer should be in JSON format as follows:
{< name of the skill>: <simple reason for the skill>}

The prompt template used to merge the knowledge is as follows:

Prompt for Knowledge Merge

Here is a list of skills required to solve a safe question: Please merge the knowledge with the same word stems and present the same meanings.
Skills: **[skills]**
Your answer should be in Json format as follows: {<name of the skill>: [<existing skill1>, ...]}

The prompt template used to reconstruct the responses is as follows:

Prompt for Response Reconstruction

 # Example:
Question: Daniel had some noodles. He gave 12 noodles to William. Now Daniel has a certain number of noodles left. Daniel had 66 noodles to begin with. How many noodles does Daniel have now?
Reasoning: Okay, so Daniel had some noodles originally, right? The problem says he had 66 noodles to begin with. Then he gave 12 of them to William. The question is asking how many noodles Daniel has left now. Hmm, this seems straightforward. If he started with 66 and gave away 12, we just need to subtract 12 from 66. ... So the answer should be 54. I think that's it.
Response: Daniel initially had 66 noodles. He gave 12 noodles to William, which means we subtract 12 from his original amount: $66 - 12 = 54$
Thus, Daniel now has $\boxed{54}$ noodles.
Please expand the Solution to the given Questions into Reasoning and Response sections, following the format provided in the Example. Please put your final answer within \boxed{}. Please respond in Json format: {'reasoning': '', 'response': ''}

 # Question: [Question]

 # Solution: [Solution]

B Preliminary Analysis

B.1 Model Performance Analysis

To validate the fine-tuning performance of 32B-scale models, we evaluate them on the same benchmarks. As shown in Tables 1 and 2, fine-tuning improves performance compared to Qwen2.5-32B-Instruct, demonstrating its effectiveness at the 32B scale. In particular, training in challenging data also improves performance in simpler tasks such as GSM8K, rather than causing degradation. Taking training on LIMO as an example, when only using math training data, the model's general capabilities improve, with the only decline observed in jailbreak-related safety performance.

*These authors contributed equally.
†Corresponding author.

Model	AIME24	AMC	MATH500	GSM8K	Olympiad	Minerva	AVG	Overall AVG
Qwen2.5	17.08	67.50	60.4	92.87	30.37	23.16	48.56	67.34
R1-Qwen	**69.38**↑	**95.62**↑	90.2↑	94.01↑	56.00↑	47.43↑	75.44↑	70.16↑
s1.1-Qwen	60.62↑	92.81↑	93.0↑	**96.13**↑	61.63↑	**52.21**↑	**76.07**↑	65.86
LIMO-Qwen	56.04↑	91.56↑	**93.6**↑	94.62↑	**65.19**↑	48.53↑	74.92↑	**77.58**↑

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.

Model	MMLU	CQA	GPQA	AVG	TrustLLM		Strong	Wild	AVG
					Misuse	Jailbreak	Reject	Jailbreak	
Qwen2.5	78.35	**86.81**	58.59	74.58	**96.66**	95.93	99.36	**68.33**	**90.07**
R1-Qwen	71.38	83.37	59.09↑	71.28	68.48	68.43	51.44	57.24	61.40
s1.1-Qwen	**86.03**↑	83.95	67.17↑	**79.05**↑	47.76	52.71	18.21	43.94	40.66
LIMO-Qwen	74.90	73.96	**75.25**↑	74.70↑	**97.81**↑	93.43	**99.68**↑	43.94	83.72

Table 2: Evaluation results on commonsense 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.

B.2 Base Model Knowledge Analysis

To determine an effective way to reflect overall knowledge proficiency, we first measure the consistency among models. We consider two strategies to reflect the knowledge proficiency:

• Average proficiency: Calculated as the average accuracy of all models on each prompt, then averaged within each knowledge unit as the proficiency score.

• Strict proficiency: A prompt is marked as proficient only when all models answer it correctly. The knowledge proficiency is then calculated as the average proficiency across prompts within each knowledge unit.

Figure 1 shows the correlation of knowledge proficiency across base models. The base models' proficiency exhibits weak correlation. We consider the similarity by deriving similar training corpora sourced from the web. Besides, the average proficiency metric shows higher consistency across models. Therefore, we adopt average proficiency as the standard for choosing prompts.

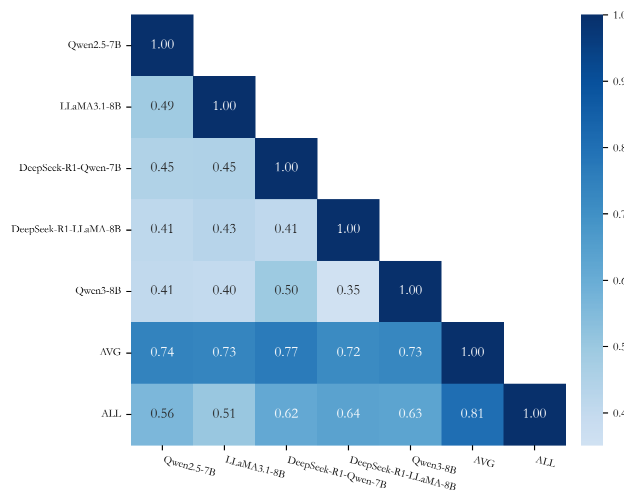

Figure 1: The correlation of knowledge proficiency between base models.

C Knowledge-based Selection

To reduce the size of training data, we adopt a greedy selection strategy, as detailed in Algorithm 1.

D Group Direct Preference Optimization

D.1 Preliminary

Reinforcement Learning Reinforcement learning (Jaques et al. 2017; Schulman et al. 2017; Stiennon et al. 2020; Ouyang et al. 2022) utilizes policy gradients for reinforcement learning, aiming to sample data from the environment to iteratively improve the policy. The optimization objective of PPO is:

$$\mathcal{J}_{RL}(\theta) = \max_{\pi_\theta} \mathbb{E}_{(q,y)\sim\mathcal{D}}\Big[r(q,y) - \beta \mathbb{D}_{\mathrm{KL}}(\pi_\theta || \pi_{ref})\Big] \tag{1}$$

Algorithm 1: Algorithm of Knowledge-based Selection

```
Input: The question set Q = q_1, q_2, ..., q_n; The knowledge set for each question K_q = {k_1, k_5, ..., k_m}; The knowledge count C; The knowledge selection ratio: r_k;
Output: The selected question set Q_f;
Initial selected knowledge count: C_s = for each knowledge with 0;
for q in Q do
    if len(K_q) ≥ r then
        update Q_f and C_s based on K_q;
    end if
end for
while True do
    Define maximum knowledge gap G_m = 0;
    for q in Q do
        Define knowledge gap G_q = 0 for q;
        for k in K_q do
            if C_{s_k} / C_k < r_k then
                update knowledge gap G_q += 1;
            end if
        end for
        if G_q > G_m then
            update G_m;
        end if
    end for
    update Q_f using the question with G_m;
    if no update then
        break;
    end if
end while
return Outputs
```

where $r(y|q)$ denotes the reward of a sampled response y for question q, and π_{old} represents the previous policy model.

DPO The original Direct Preference Optimization (DPO) (Rafailov et al. 2023) is designed to optimize the policy model based on paired preference data derived from general reinforcement learning, including a chosen response and a rejected response. The loss for DPO is formalized as follows:

$$\mathcal{L}_{DPO}(\theta) = -\mathbb{E}_{(q,y_w,y_l)\sim\mathcal{D}}\Big[\log\sigma\Big(\beta\log(\frac{\pi_\theta(y_w|q)}{\pi_{ref}(y_w|q)}) - \beta\log(\frac{\pi_\theta(y_l|q)}{\pi_{ref}(y_l|q)})\Big)\Big], \tag{2}$$

where θ denotes the policy model parameters, y_w and y_l represent the preferred and rejected responses for the question q, respectively.

GRPO The Group Relative Policy Optimization (GRPO) (Shao et al. 2024) considering the advantages within a group of sampled responses $\{y_1, y_2, ..., y_G\}$. It leverages the advantages of sample responses from the reference model to optimize the policy model θ. The corresponding loss is formalized as follows:

$$\mathcal{L}_{GRPO}(\theta) = -\mathbb{E}_{q\sim\mathcal{D}, y_i\sim\pi_\theta(y_i|q)}\Big[\frac{1}{G}\sum_{i=1}^{G}\Big(\frac{\pi_\theta(y_i \mid q)}{\pi_{old}(y_i \mid q)}A_i - \beta\mathbb{D}_{\text{KL}}(\pi_\theta||\pi_{ref})\Big)\Big], \tag{3}$$

where β is a hyper-parameter, $A_i = \frac{r_i - mean(\{r_1, r_2, ..., r_G\})}{std(\{r_1, r_2, ..., r_G\})}$ is the advantage of each response presented in the group, and $\mathbb{D}_{\text{KL}}(\pi_\theta||\pi_{ref}) = \frac{\pi_{ref}(y_i|q)}{\pi_\theta(y_i|q)} - \log\frac{\pi_{ref}(y_i|q)}{\pi_\theta(y_i|q)} - 1$.

However, DPO ignores the advantage distribution within groups of responses. Besides, the GRPO online sampling strategy is time-consuming, and its convergence speed is highly dependent on the inherent capabilities of the reference model. Therefore, we propose Group Direct Policy Optimization (GDPO), which mitigates these limitations by providing high-quality offline samples and a more stable optimization process.

D.2 Extend GRPO to GDPO

We first extend GRPO to GDPO to reduce the inference expansion during GRPO. The key benefit of GDPO for knowledge distillation is that it enables the student model to learn the underlying response distribution more effectively. The benefit of GDPO in knowledge distillation is that the student model can learn the response distribution. To precisely determine the weight in GDPO and fully preserve the feature of GRPO, we derive the loss of GDPO step by step, similar to DPO.

KL-Constrained Reward We first derive the optimal KL-constrained reward objective from the original objective. For simplicity, we denote $r(q, y_i)$ as the reward assigned to response y_i for question q. Then, we can reformulate the objective with explicit advantages as follows:

$$\begin{aligned}
&\max_{\pi_\theta}\mathbb{E}_{q\sim\mathcal{D}, y_i\sim\pi_\theta(y_i|q)}\Big[\frac{1}{G}\sum_{i=1}^{G}\Big(r(q,y_i)A_i - \beta\mathbb{D}_{KL}(\pi_\theta||\pi_{ref})\Big)\Big] \\
&= \max_{\pi_\theta}\mathbb{E}_{q\sim\mathcal{D}, y_i\sim\pi_\theta(y_i|q)}\Big[\frac{1}{G}\sum_{i=1}^{G}\Big(r(q,y_i)A_i - \beta\Big(\frac{\pi_{ref}(y_i \mid q)}{\pi_\theta(y_i \mid q)} - \log\frac{\pi_{ref}(y_i \mid q)}{\pi_\theta(y_i \mid q)} - 1\Big)\Big)\Big]
\end{aligned} \tag{4}$$

As $\mathbb{E}_{y_i\sim\pi_\theta}[\frac{1}{r(y_i,q)}] = \mathbb{E}_{y_i\sim\pi_\theta}[\frac{\pi_{ref}(y_i|q)}{\pi_\theta(y_i|q)}] = \int\pi_\theta(y_i)\frac{\pi_{ref}(y_i)}{\pi_\theta(y_i)}dy_i = \int\pi_{ref}(y_i)dy_i = 1$, the above equation can be transferred as

$$\min_{\pi_\theta}\mathbb{E}\Big[\frac{1}{G}\sum_{i=1}^{G}\Big(\log\frac{\pi_\theta(y_i \mid q)}{\pi_{ref}(y_i \mid q)} - \frac{A_i}{\beta}r(q,y_i)\Big)\Big] \tag{5}$$

We introduce the partition function: $Z(q) = \sum_{y_i}\pi_{ref}(y_i|q)\exp(\frac{A_i}{\beta}r(q,y_i))$, which is independent from π_θ, to re-organize the objection. Based on Jensen's Inequality and Variational Inference, we can obtain the following variable representation:

$$
\begin{aligned}
&\log Z(q) = \log \mathbb{E}_{y_i \sim \pi_{ref}}[\pi_{ref}(y_i|q) \exp(\frac{A_i}{\beta} r(q, y_i))] \\
&= \log \int \rho(y_i|q) \frac{\pi_{ref}(y_i|q) \exp(\frac{A_i}{\beta} r(q, y_i))}{\rho(y_i|q)} dy_i \\
&\geq \int \rho(y_i|q) \log \frac{\pi_{ref}(y_i|q) \exp(\frac{A_i}{\beta} r(q, y_i))}{\rho(y_i|q)} dy_i \\
&= \int \rho(y_i|q) \Big(\log(\frac{\pi_{ref}(y_i|q)}{\rho(y_i|q)} + \frac{A_i}{\beta} r(q, y_i)) \Big) dy_i \\
&= \max_{\rho} \Big\{ \mathbb{E}_{y_i \sim \rho}\Big[\frac{A_i}{\beta} r(q, y_i) \Big] - \mathbb{D}_{\mathrm{KL}}(\rho(y_i|q) || \pi_{ref}(y_i|q)) \Big\},
\end{aligned} \tag{6}
$$

where ρ is an arbitrary distribution that satisfies $\int \rho(y_i|q) dy_i = 1$ and $\rho > 0$.

Then, the objective in Eq. 5 can be rewritten as:

$$
\begin{aligned}
\min_{\pi_\theta} \mathbb{E}\Big[\frac{1}{G} \sum_{i=1}^{G} \Big(\log \frac{\pi_\theta(y_i \mid q)}{\frac{1}{Z(q)} \pi_{ref} \exp(\frac{A_i}{\beta} r(q, y_i))} \\
- \log Z(q) \Big) \Big].
\end{aligned} \tag{7}
$$

We define $\pi^* = \frac{1}{Z(q)} \pi_{ref}(y_i|q) \exp(\frac{A_i}{\beta} r(q, y_i))$, then the final optimization objective of π_θ can be simplified as:

$$
\min_{\pi_\theta} \mathbb{E}\Big[\frac{1}{G} \sum_{i=1}^{G} \Big(\log \frac{\pi_\theta(y_i|q)}{\pi^*(y_i|q)} - \log Z(q) \Big) \Big]. \tag{8}
$$

Finally, we can derive the optimal solution by leveraging the properties of the KL-divergence:

$$
\pi_\theta(y_i|q) = \pi^*(y_i|q) = \frac{1}{Z(q)} \pi_{ref}(y_i|q) \exp(\frac{A_i}{\beta} r(q, y_i)). \tag{9}
$$

GDPO Objective We first sort the responses based on the advantages and obtain $y_1 \succ y_2 \succ \cdots \succ y_G$, where the corresponding advantages satisfy $A_1 \geq A_2 \geq \cdots \geq A_G$. Similar to DPO, we use the Bradley-Terry model (Bradley and Terry 1952) to model the pairwise preference, treating the advantage difference as the preference intensity. Based on Eq. 9, the ground-truth reward for each response can be expressed as $r^* = \frac{\beta}{A_i} \log \frac{\pi^*(y_i|q)}{\pi_{ref}(y_i|q)} + \frac{\beta}{A_i} \log Z(q)$. Considering the contributions of all pairs within the group (totally $\frac{G(G-1)}{2}$ unique pairs), the preference model can be formalized as follows:

$$
\begin{aligned}
p^*(yi \succ y_j) &= \frac{\exp(r(q, y_i))}{\exp(r(q, y_i)) + \exp(r(q, y_j))} \\
&= \frac{1}{1 + \frac{\exp(\frac{\beta}{A_j} \log \frac{\pi^*(y_j|q)}{\pi_{ref}(y_j|q)} + \frac{\beta}{A_j} \log Z(q))}{\exp(\frac{\beta}{A_i} \log \frac{\pi^*(y_i|q)}{\pi_{ref}(y_i|q)} + \frac{\beta}{A_i} \log Z(q))}} \\
&= \frac{1}{1 + \exp\Big(\frac{\beta}{A_j} \log \frac{\pi^*(y_j|q)}{\pi_{ref}(y_j|q)} - \frac{\beta}{A_i} \log \frac{\pi^*(y_i|q)}{\pi_{ref}(y_i|q)} - f(Z(q)) \Big)} \\
&= \sigma\Big(\frac{\beta}{A_i} \log \frac{\pi^*(y_i|q)}{\pi_{ref}(y_i|q)} - \frac{\beta}{A_j} \log \frac{\pi^*(y_j|q)}{\pi_{ref}(y_j|q)} + f(Z(q)) \Big)
\end{aligned} \tag{10}
$$

where $f(Z(q)) = \beta \log Z(q) (\frac{1}{A_i} - \frac{1}{A_j})$.

$$
\begin{aligned}
\mathcal{L}_{\mathrm{GDPO}}(\theta) &= -\frac{2}{G(G-1)} \sum_{i=1}^{G-1} \sum_{j>i}^{G} p_\theta(yi \succ y_j) \\
&= -\frac{2}{G(G-1)} \sum_{i=1}^{G-1} \sum_{j>i}^{G} \sigma\Big(\frac{\beta}{A_i} \log \frac{\pi_\theta(y_i|q)}{\pi_{ref}(y_i|q)} - \\
&\qquad \frac{\beta}{A_j} \log \frac{\pi_\theta(y_j|q)}{\pi_{ref}(y_j|q)} + f(Z(q)) \Big).
\end{aligned} \tag{11}
$$

As $f(Z(q))$ is irrelevant to π_θ, we can ignore this part without disrupting the training progress. The optimization loss can be rewritten as:

$$
\begin{aligned}
\widetilde{\mathcal{L}}_{\mathrm{GDPO}}(\theta) &= -\frac{2}{G(G-1)} \sum_{i=1}^{G-1} \sum_{j>i}^{G} \sigma\Big(\frac{\beta}{A_i} \log \frac{\pi_\theta(y_i|q)}{\pi_{ref}(y_i|q)} \\
&\qquad - \frac{\beta}{A_j} \log \frac{\pi_\theta(y_j|q)}{\pi_{ref}(y_j|q)} \Big), \\
&= -\frac{2}{G(G-1)} \sum_{i=1}^{G-1} \sum_{j>i}^{G} \sigma(\Delta \widetilde{r}_{ij}),
\end{aligned} \tag{12}
$$

where $\Delta \widetilde{r}_{ij} = \frac{\beta}{A_i} \log \widetilde{r}_i - \frac{\beta}{A_j} \log \widetilde{r}_j$ and $\widetilde{r}_i = \frac{\pi_\theta(y_i|q)}{\pi_{ref}(y_i|q)}$.

D.3 Optimize GDPO

According to Eq. 12, the computational complexity of all preference pairs within the group is $\mathcal{O}(G^2)$. To reduce computational complexity, we only consider adjacent index pairs $(i, i+1)$ within each sorted group and define the approximate loss function as follows:

$$
\widetilde{\mathcal{L}}_{\mathrm{approx}}(\theta) = -\frac{1}{G-1} \sum_{i=1}^{G-1} \sigma(\Delta \widetilde{r}_{i,i+1}). \tag{13}
$$

The approximate loss only involves $G-1$ pairwise computations, thereby reducing the overall computational complexity to $\mathcal{O}(G)$.

We denote the expected score of adjacent pairs as $\mu_{adj} = \mathbb{E}[\sigma(\Delta \widetilde{r}_{i,i+1})]$, and non-adjacent pairs as $\mu_{non} = \mathbb{E}[\sigma(\Delta \widetilde{r}_{i,j})]$:

$$
\begin{aligned}
\mathbb{E}[\widetilde{\mathcal{L}}_{\text{GDPO}}] &= -\frac{2}{G(G-1)}\sum_{i<j}\mathbb{E}\left[\sigma\left(\Delta\widetilde{r}_{ij}\right)\right] \\
&= -\frac{2}{G(G-1)}\Big((G-1)\mu_{\text{adj}} \\
&\qquad + \frac{(G-1)(G-2)}{2}\mu_{\text{non}}\Big),
\end{aligned} \tag{14}
$$

$$
\begin{aligned}
\mathbb{E}[\widetilde{\mathcal{L}}_{\text{approx}}] &= -\frac{1}{(G-1)}(G-1)\mu_{adj} \\
&= -\mu_{adj}.
\end{aligned} \tag{15}
$$

Considering the properties of $\sigma(\cdot)$, when the group size G is huge, the reward difference $|\Delta\widetilde{r}_{k,k+1}|$ becomes small. Therefore, we can approximate $\sigma\left(\Delta\widetilde{r}_{ij}\right) \approx \frac{1}{2} + \frac{1}{4}\sum_{k=i}^{j}\mathbb{E}[\Delta\widetilde{r}_{k,k+1}]$. Consequently, the expected value over adjacent pairs can be expressed as $\mu_{adj} \approx \frac{1}{2} + \frac{1}{4}\mathbb{E}[\Delta\widetilde{r}_{k,k+1}]$. When $G \to \infty$, both μ_{adj} and μ_{non} converge to $\frac{1}{2}$. Then, the expected approximate loss converges to $\mathbb{E}[\widetilde{L}_{\text{GDPO}}] \approx -\mu_{adj}$, indicating that, as G becomes large, the expectations of the full GDPO loss and the approximate loss are approximately consistent.

We consider the sampling of responses as a Monte Carlo process, where N samples are drawn from a huge group of size G. The discrepancies between the empirical expectations of the approximate loss $\mathbb{E}'[\widetilde{\mathcal{L}}_{\text{approx}}]$ and $\mathbb{E}'[\widetilde{\mathcal{L}}_{\text{GDPO}}]$, relative to the ideal expectation $\mathbb{E}[\widetilde{\mathcal{L}}_{\text{GDPO}}]$ are:

$$
\begin{aligned}
\epsilon_{\text{GDPO}} =& |\mathbb{E}[\widetilde{\mathcal{L}}_{\text{GDPO}}] - \mathbb{E}'[\widetilde{\mathcal{L}}_{\text{GDPO}}]| \\
=& |\mu_{adj} - \frac{2}{N}\mu'_{adj} - \frac{N-2}{N}\mu'_{non}| \\
=& |\mu_{adj} + \frac{N-2}{N}(\mu'_{adj} - \mu'_{non}) - \frac{N-4}{N}\mu'_{adj}|, \\
\epsilon_{\text{approx}} =& |\mathbb{E}[\widetilde{\mathcal{L}}_{\text{GDPO}}] - \mathbb{E}'[\widetilde{\mathcal{L}}_{\text{approx}}]| \\
=& |\mu_{adj} - \mu'_{adj}|
\end{aligned} \tag{16}
$$

We first analyze the variance of two losses:

$$
\operatorname{Var}\left(\widetilde{\mathcal{L}}_{\text{GDPO}}\right) = \frac{4}{N^2(N-1)^2}\operatorname{Var}\left(\sum_{i<j}\sigma(\Delta_{ij})\right). \tag{17}
$$

When the group size G is relatively large, the sampled responses can be considered dense. We consider that the finite G setting can be viewed as a Monte Carlo approximation of the ideal optimization scenario. As $\sigma(\cdot) \in [\frac{1}{2}, 1)$ due to $\Delta_{ij} \geq 0$, we can obtain that the upper boundary of $\operatorname{Var}(\sigma)$ is $V_u = \frac{1}{16}$. Therefore, we can have the following bound: $\sum_{i<j}\operatorname{Var}(\sigma(\Delta_{ij})) \leq \frac{G(G-1)}{2}V_u$. As the samples are dependent on each other, we can ignore the covariance terms $\operatorname{Cov}(\sigma(\Delta_{ij}), \sigma(\Delta_{kl}))$. Then, we can have:

$$
\operatorname{Var}\left(\widetilde{\mathcal{L}}_{\text{GDPO}}\right) \leq \frac{2}{N(N-1)}\operatorname{Var}(\mu'_{non}). \tag{18}
$$

We denote the variance of the sigmoid over adjacent pairs as $\operatorname{Var}(\sigma)$. Then, the variance of the approximate loss $\operatorname{Var}\left(\widetilde{\mathcal{L}}_{\text{approx}}\right)$ can be expressed as:

$$
\begin{aligned}
\operatorname{Var}\left(\widetilde{\mathcal{L}}_{\text{approx}}\right) &\leq \frac{1}{(N-1)^2}\sum_{i=1}^{N-1}\operatorname{Var}(\sigma(\Delta\widetilde{r}_{i,i+1})) \\
&\leq \frac{\operatorname{Var}(\mu'_{adj})}{N-1}.
\end{aligned} \tag{19}
$$

Then, we estimate the error as $\text{MSE} \leq \epsilon^2 + \text{Var}$. Given that $\mu'_{adj} \approx \frac{1}{2} + \frac{G}{N}\Delta\widetilde{r}_{k,k+1}$, its variance can be approximated as $\mu'_{adj} \approx \frac{G^2}{N^2}\Delta\widetilde{r}^2_{k,k+1}$. The term $G\Delta\widetilde{r}_{k,k+1}$ represents the score difference between the best and worst samples within the group, which is denoted as Δ_{max}. Due to the difference $\mu_{adj} - \mu_{non}$ decreases at a rate of $\mathcal{O}(\frac{1}{G})$. Accordingly, the approximation error of $\widetilde{\mathcal{L}}_{\text{GDPO}}$ reduces as a speed of $\mathcal{O}(\frac{1}{G^2})$, while the error in $\widetilde{\mathcal{L}}_{\text{approx}}$ reduces at $\mathcal{O}(\frac{1}{G})$. We compare the relative error reduction when increasing the sample size from $N = 2$ to $N = 10$ as $\frac{0.5+\frac{G^2}{4}\Delta\widetilde{r}^2_{k,k+1} - 0.057 - 0.01G^2\Delta\widetilde{r}^2_{k,k+1}}{0.5+\frac{G^2}{4}\Delta\widetilde{r}^2_{k,k+1}}$. Assuming $G\Delta\widetilde{r}_{k,k+1} = 1$, the approximation error can reduce around 90%. Moreover, this reduction improves further as the total ranking gap increases.

Therefore, when we adopt $N = 10$ as the small variance and error, we make the optimization more stable and closer to convergence to the ideal GDPO objective.

E Comparisons

E.1 Approximate Proof

In this section, we prove that the GDPO approximates the optimization process of GRPO.

Proof. We assume $\pi_{old} = \pi_{ref}$ (Shao et al. 2024), then the loss of GRPO is estimated to be:

$$
\widetilde{L}(\theta) = -\frac{1}{G}\sum_{i=1}^{G}\Big(\frac{\pi_\theta(y_i|q)}{\pi_{ref}(y_i|q)}A_i - \beta\log\frac{\pi_\theta(y_i|q)}{\pi_{ref}(y_i|q)}\Big). \tag{20}
$$

The optimize objective can be written as:

$$
\mathcal{J} = \int \pi_\theta(y_i)\left(e^{\log\widetilde{r}_i}A_i - \beta\log\widetilde{r}_i\right)dy. \tag{21}
$$

To determine the optimal strategy, we compute the derivative variable of the optimization objective and set it to zero:

$$
\begin{aligned}
&\frac{\delta\mathcal{J}}{\delta\pi_\theta(y_i|q)} = e^{\log\widetilde{r}_i}A_i - \beta\log\widetilde{r}_i + \lambda = 0 \\
&\Rightarrow \frac{A_i}{\pi_{ref}} + \beta(1 - \log\frac{\pi_\theta}{\pi_{ref}}) + \lambda = 0 \\
&\Rightarrow \pi_\theta(y_i|q) = \frac{1}{Z_{\text{GRPO}}(q)}\pi_{ref}(y_i|q)\exp(\frac{A_i}{\beta\pi_{ref}(y_i|q)} - 1) \\
&\Rightarrow \pi_\theta(y_i|q) \propto \pi_{ref}(y_i|q)\exp(\frac{A_i}{\beta}),
\end{aligned} \tag{22}
$$

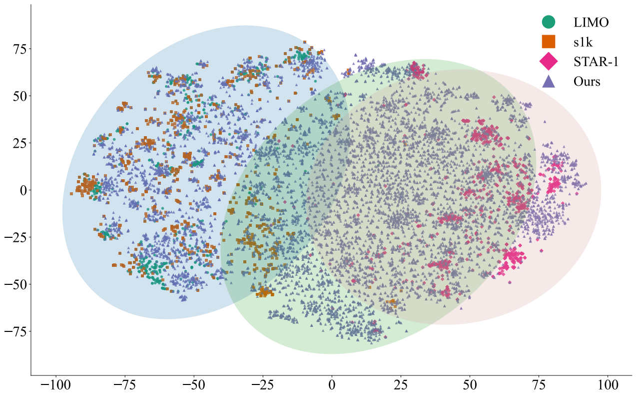

Figure 2: The visualization of the knowledge semantic distribution. Blue background indicates mathematical data, green represents general data, and pink denotes safety-related data.

where the partition function is $Z_{\text{GRPO}}(q) = \int \pi_{ref}(y_i|q)\exp(\frac{A_i}{\beta\pi_{ref}(y_i|q)} - 1)dy_i$ leading the Lagrange multiplier $\lambda = \beta \log \frac{1}{Z_{\text{GRPO}}(q)}$. The GRPO objective maximizes the expected reward while constraining the policy π_θ to be close to the reference policy π_{ref} through KL-divergence constraint. The resulting optimal policy meets $\pi_\theta(y_i|q) \propto \pi_{ref}(y_i|q)\exp\left(\frac{A_i}{\beta}\right)$.

Meanwhile, we derive the variable derivation of $\mathcal{L}_{\text{GDPO}}$:

$$\begin{aligned}\frac{\delta J}{\delta\pi_\theta(y_i|q)} &= \sigma'(\Delta\widetilde{r}_{i,i+1})\frac{\beta}{A_i\pi_\theta(y_i|q)} \\ &\quad - \sigma'(\Delta\widetilde{r}_{i-1,i})\frac{\beta}{A_i\pi_\theta(y_i|q)}.\end{aligned} \tag{23}$$

To obtain the optimal policy, we set the functional derivative of $\delta\mathcal{J}$ with respect to $\delta\pi_\theta(y_i|q)$ to zero. Therefore, we can have $\frac{\log\widetilde{r}_i}{A_i} = \frac{\log\widetilde{r}_{i+1}}{A_{i+1}} = C$, where C is a constant. Finally, we can have the optimal policy: $\pi_\theta \propto \pi_{ref}(y_i|q)\exp(CA_i)$. We can observe that GDPO and GRPO share the same optimization direction.

When $C = \frac{1}{\beta}$, the optimization of GDPO is the same as GRPO. When $C < \frac{1}{\beta}$, GDPO adopts a more aggressive optimization strategy, accelerating convergence at the cost of stability. In contrast, when $C > \frac{1}{\beta}$, GDPO becomes more conservative, prioritizing stable updates at the expense of slower convergence.

□

E.2 Compare with GRPO

Based on the above analysis, we summarize the comparison between GDPO and GRPO in this section. The core difference lies in how the response y_i is obtained: in GRPO, it is sampled by the online policy model π_θ, whereas in GDPO, y_i can be predefined by powerful LLMs. This discrepancy indicates that GRPO is better suited for tasks requiring creative exploration and dynamic adaptation to the environment, while GDPO is more appropriate for knowledge distillation. Table 3 presents a detailed comparison between the two methods. It is obvious that GRPO tends to converge slowly early due to the uncontrollable quality of sampled responses, whereas GDPO benefits from high-quality samples that accelerate early-stage learning. Therefore, GDPO can be used as a warm-up stage to speed up learning and bring performance closer to advanced LLMs.

E.3 Compare with DPO

In this section, we compare the differences between GDPO and DPO. First, DPO directly optimizes the implicit reward, while GDPO allows for adjustment of the optimization density. Samples with high advantage can be reduced in the gradient update to avoid overfitting, while those with lower advantage can be enhanced on refinement. In contrast, DPO assigns equal gradient weights to all samples, which may lead to overfitting on high-reward responses. Besides, GDPO emphasizes distinctions in high-advantage regions while suppressing fluctuations in low-advantage areas during optimization. This enables GDPO to refine the probability distribution more effectively and preserve generation diversity.

E.4 Compared with SFT

We observe that SFT easily leads to catastrophic forgetting in small models' distillation. Recall the loss of SFT:

$$\mathcal{L}_{SFT}(\theta) = -\mathbb{E}_{(q,y)\sim\mathcal{D}_{\text{sft}}}\left[\log\pi_\theta(y|q)\right],$$

where $\mathcal{D}_{\text{sft}}$ is the fine-tuning dataset, we can find that SFT directly maximizes the likelihood of new data without introducing any constraints to retain previously knowledge. Gradient updates solely rely on SFT data, leading to excessive adjustments of model parameters to fit the new task, which may overwrite the parameter distribution associated with the old task. Additionally, SFT does not explicitly reference the original model, and its optimization process is entirely biased toward the new data, lacking any mechanism to preserve prior knowledge. In contrast, GDPO implicitly incorporates the reference model as an anchor, constraining the optimization trajectory to prevent drastic parameter changes.

F Data Analysis

F.1 Training Data Composition

Table 4 lists the relationship between the existing training datasets used in this paper and the corresponding number of samples included in our dataset.

F.2 Data Distribution

Knowledge Semantic Analysis To characterize the semantic properties of our constructed dataset METAKL relative to existing high-quality datasets (eg, LIMO and slk), we visualize their semantic embedding to analyze the distribution.

For each dataset, we obtain semantic features by feeding the concatenation of the knowledge name and its explanation into Qwen3-Embedding-4B (Zhang et al. 2025). Then, we applied t-SNE dimensionality reduction to project high-dimensional embeddings into a 2D space. The resulting visualization is exhibited in Figure 2. We can observe that our

Dimension	GRPO	GDPO
	Convergency speed	
Sampling Efficiency	Low (repeat generating new samples.)	High (fixed samples with controllable quality)
Gradient Variance	Large (strategy changes may lead to sample distribution drift)	Small (static sample distribution)
Convergency Speed	Slow (needs more iterative stability strategies)	Fast (directly learn teacher knowledge)
Computation Expense	High (resampling is required for each step)	Low (sample pre-calculation)
	Optimization performance	
Upper Boundary	Higher (self-exploring can trigger new behaviors)	Lower (limited by the ability of the teacher LLM)
Alignment Quality	Depend on the design of reward functions	Depend on teacher LLM's response quality

Table 3: Comparison between GRPO and GDPO.

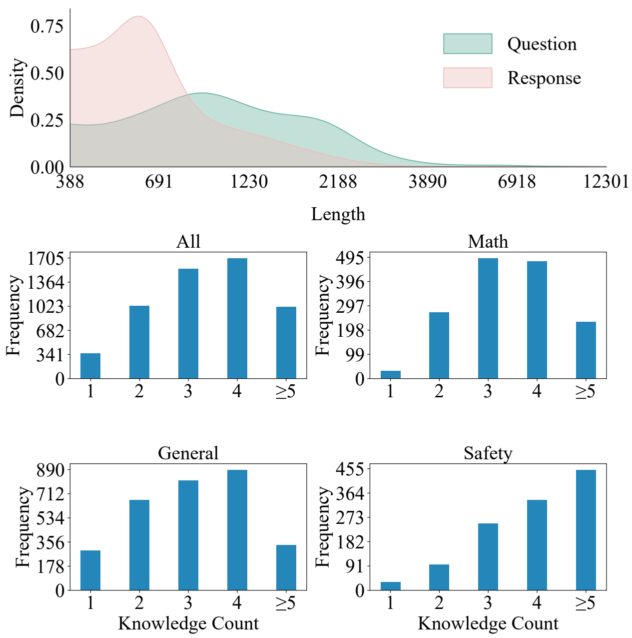

Figure 3: The statistics visualization of METAKL.

dataset cover more knowledge range compared with other datasets. We also observe that the semantic spaces of general and safety data partially overlap, while the mathematical domain remains relatively independent, indicating the difficulty in learning mathematical knowledge patterns.

Statistic Analysis We present dataset statistics in Figure 3. The top part displays the length distributions of questions and responses, reflecting the dataset's diversity and complexity. The bottom panel illustrates the distribution of knowledge types across different data categories. Figure 4 shows the relationship between reward and knowledge count from two perspectives: individual response rewards and the

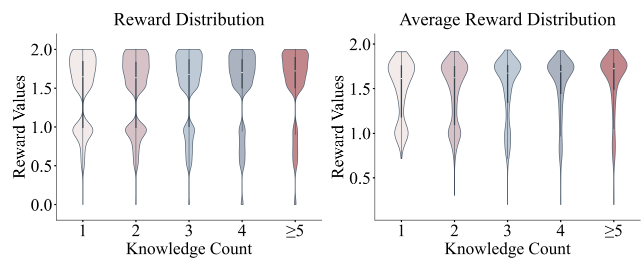

Figure 4: The correlation of knowledge proficiency between base models.

average reward across ten responses for the prompt. We observe that reward discrepancies across knowledge counts are minimal. However, as the knowledge count increases, the likelihood of generating longer responses without producing a final answer also increases.

G Experimental Settings

For all SFT experiments, we follow fine-tuning hyperparameters consistent with previous works: the learning rate is set to 1e-5, batch size to 128, and the number of training epochs to 5. For GDPO experiments, we use a learning rate of 2e-6, β of 0.1, and a batch size of 128. All experiments are conducted using two A800 GPUs.

H Experimental Results

H.1 Results on Ablation Study

Tables 5 and 6 present the results of removing the advantage weight from the optimization loss. While most benchmarks still show improvement without the advantage weight, the performance remains weaker than GDPO, indicating the effectiveness of incorporating the advantage weights.

Kind	Data Source	Description	Count
Reasoning	NuminaMath-CoT (LI et al. 2024)	Around 860K math problems, each presented with Chain-of-Thought reasoning. The problems are sourced from Chinese high school exams, U.S. competitions, and international Olympiads.	1,503
	OpenAIMath (Muennighoff et al. 2025)	A structured mathematical reasoning dataset including 12K training samples and 500 test set sourced from MATH for s1k.	44
General	MMLU (Hendrycks et al. 2021)	A multiple-choice dataset covering 57 tasks across STEM, humanities, and social sciences.	1,193
	Commonsense QA (Talmor et al. 2019)	A multiple-choice dataset with 12,247 questions that require models to choose concepts related to a given source concept across various real-world situations.	390
	Commonsense QA 2.0 (Talmor et al. 2022)	A dataset created using a gamified framework to generate challenging yes/no questions to test models' commonsense reasoning across 14,343 statements.	589
	LogiQA2.0 (Liu et al. 2023)	Including multiple-choice reading comprehension questions that require logical reasoning.	821
Safety	ALERT(Tedeschi et al. 2024)	Collect 45k instructions through red teaming methodologies	27
	BeaverTails (Ji et al. 2023)	A dataset used for safety alignment in LLMs, with both helpfulness and harmlessness annotations.	167
	WildJailbreak (Jiang et al. 2024)	A synthetic safety-training dataset containing 262K direct harmful prompts and complex adversarial jailbreak prompts.	439
	UltraSafety (Guo et al. 2024)	A dataset of 1K seed instructions and variants from safety benchmarks and 2K from self-instruct variants.	2
	PKU-SafeRLHF (Ji et al. 2024)	A dataset comprising 44.6k refined prompts, 265k Q-A pairs annotated with 19 harm categories and 3 severity levels along with 166.8k preference data.	115
	HH-RLHF (Bai et al. 2022)	A human-annotated dialogue preference dataset comprising 47.9k comparative pairs for helpfulness and 16.2k adversarial red-teaming pairs for harmlessness.	41
	Do Anything Now (Shen et al. 2024)	A wild jailbreak prompts collection collected from four platforms with 0.95 attack success rates on ChatGPT (OpenAI 2022) and GPT-4 (Achiam et al. 2023).	330

Table 4: Description of the training datasets and the number of preserved instances.

H.2 Training Methods Compare

Tables 7 and 8 present the results across different training methods. Notably, only GDPO consistently avoids performance degradation across all application scenarios. In contrast, SFT is more suitable for safety alignment but leads to a decline in reasoning capabilities. Compared with GRPO, our method enhances model performance across multiple dimensions, whereas GRPO is more effective on datasets where the model already exhibits strong inherent capabilities. This phenomenon shows that for difficult tasks, response quality affects training performance.

H.3 Dataset Composition Analysis

We provide the detailed results of training with different data compositions using GDPO in Table 9 and Table 10. We can find that using only the math or safety subset from our dataset still outperforms LIMO and STAR-1, demonstrating the effectiveness of data selection based on metacognitive knowledge.

H.4 Supplementary Results

We also conduct experiments on Qwen2.5-7B-Instruct and LLaMA-3.1-8B-Instruct. The detailed results are presented in Table 11 and Table 12. We observe that training on reasoning-focused data is less effective for LLMs not originally optimized for reasoning tasks. Nevertheless, our method consistently achieves the best performance in fine-tuning, effectively mitigating catastrophic forgetting. Upon examining the responses generated by the fine-tuned models, we find that they tend to produce longer outputs without arriving at a final answer. This suggests that the models are attempting to emulate extended reasoning patterns, but require more massive training data to precisely learn how to

Model		AIME24	AMC	MATH500	GSM8K	Olympiad	Minerva	AVG	Overall AVG
Qwen3	Origin	72.71	95.16	93.8	95.10	64.89	53.31	79.16	78.98
	GDPO$_{-A}$	**76.88**↑	**95.62**↑	93.4	**95.45**↑	**64.89**	53.31	79.93↑	80.25↑
	GDPO	76.04↑	94.69	**94.0**↑	95.22↑	64.30	**56.25**↑	**80.08**↑	**80.86**↑

Table 5: Comparison of GDPO and GDPO without advantages on mathematical benchmarks.

Model		MMLU	CQA	GPQA	AVG	TrustLLM Misuse	TrustLLM Jailbreak	Strong Reject	Wild Jailbreak	AVG
Qwen3	Origin	79.28	77.89	59.09	72.09	92.98	83.50	94.89	64.12	83.87
	GDPO$_{-A}$	**83.56**↑	84.03↑	59.09	75.56↑	**92.98**	83.07	**96.17**↑	**64.84**↑	84.27↑
	GDPO	83.37↑	**84.11**↑	**62.12**↑	**76.79**↑	92.89	**84.79**↑	95.85↑	64.12	**84.41**↑

Table 6: Comparison of GDPO and GDPO without advantages on general reasoning and safety benchmarks.

reason. For Qwen2.5-7B-Instruct, our method still improves performance on some benchmarks. However, LLaMA-3.1-8B-Instruct proves more challenging to fine-tune. In particular, under SFT, LLaMA-3.1-8B-Instruct is more prone to catastrophic forgetting.

I Case Study

To express the effectiveness of training, we provide some cases in Table 13, 14 and 15 to compare the responses before and after training.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; DasSarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

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.

Guo, Y.; Cui, G.; Yuan, L.; Ding, N.; Sun, Z.; Sun, B.; Chen, H.; Xie, R.; Zhou, J.; Lin, Y.; Liu, Z.; and Sun, M. 2024. Controllable Preference Optimization: Toward Controllable Multi-Objective Alignment. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 1437–1454. Miami, Florida, USA: Association for Computational Linguistics.

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

Jaques, N.; Gu, S.; Bahdanau, D.; Hernández-Lobato, J. M.; Turner, R. E.; and Eck, D. 2017. Sequence tutor: Conservative fine-tuning of sequence generation models with kl-control. In *International Conference on Machine Learning*, 1645–1654. PMLR.

Ji, J.; Hong, D.; Zhang, B.; Chen, B.; Dai, J.; Zheng, B.; Qiu, T.; Li, B.; and Yang, Y. 2024. Pku-saferlhf: A safety alignment preference dataset for llama family models. *arXiv e-prints*, arXiv–2406.

Ji, J.; Liu, M.; Dai, J.; Pan, X.; Zhang, C.; Bian, C.; Chen, B.; Sun, R.; Wang, Y.; and Yang, Y. 2023. BeaverTails: Towards Improved Safety Alignment of LLM via a Human-Preference Dataset. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

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*.

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. [https://huggingface.co/AI-MO/NuminaMath-CoT](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.

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. 2022. ChatGPT. https://chat.openai.com/.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P. F.; Leike, J.; and

	Model	AIME24	AMC	MATH500	GSM8K	Olympiad	Minerva	AVG	Overall AVG
	Origin	72.71	95.16	93.8	95.10	64.89	53.31	79.16	78.98
Qwen3	SFT	39.38	68.28	84.4	94.84	52.30	41.18	63.40	74.14
	SFT+LoRA	74.58↑	94.53	**94.0**↑	94.31	63.11	54.41↑	79.16	79.58↑
	DPO	73.33↑	94.69	92.6	94.09	65.33↑	54.04↑	79.01	77.60
	DPO+LoRA	**77.29**↑	95.00	92.0	94.39	63.85	50.37	78.82	78.81
	GRPO	71.88	**95.16**	93.00	**95.38**↑	65.19↑	55.15↑	79.29↑	80.25↑
	GDPO	76.04↑	94.69	**94.0**↑	95.22↑	64.30	**56.25**↑	**80.08**↑	**80.86**↑
	GDPO+LoRA	75.42↑	94.22	92.8	94.84	**67.26**↑	52.57	79.52↑	79.56↑

Table 7: Evaluation results on mathematical benchmarks across different training methods.

	Model	MMLU	CQA	GPQA	AVG	TrustLLM Misuse	TrustLLM Jailbreak	Strong Reject	Wild Jailbreak	AVG
	Origin	79.28	77.89	59.09	72.09	92.98	83.50	94.89	64.12	83.87
Qwen3	SFT	81.11↑	82.06↑	49.49	70.89	**96.40**↑	**95.36**↑	**98.72**↑	**80.27**↑	**92.69**↑
	SFT+LoRA	78.90	83.78↑	59.60↑	74.09↑	93.24↑	83.07	95.53↑	65.52↑	84.34↑
	DPO	79.08	78.79↑	58.08	71.98	88.41	81.64	86.26	62.44	79.69
	DPO+LoRA	78.66	77.72	60.61↑	72.33↑	92.80	82.43	95.53↑	63.89	83.66
	GRPO	83.34↑	83.62↑	**65.66**↑	**77.54**↑	90.17	84.21↑	94.89	65.57↑	83.71
	GDPO	**84.13**↑	**84.11**↑	62.12↑	76.79↑	93.33↑	84.79↑	95.85↑	66.38↑	85.09↑
	GDPO+LoRA	83.76↑	83.87↑	56.57	74.73↑	92.45	83.00	93.61	63.85	83.23

Table 8: Evaluation results on commonsense reasoning and safety benchmarks across different training methods.

Lowe, R. 2022. Training language models to follow instructions with human feedback. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 27730–27744. Curran Associates, Inc.

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*.

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.

Shen, X.; Chen, Z.; Backes, M.; Shen, Y.; and Zhang, Y. 2024. “Do Anything Now”: Characterizing and Evaluating In-The-Wild Jailbreak Prompts on Large Language Models. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM.

Stiennon, N.; Ouyang, L.; Wu, J.; Ziegler, D.; Lowe, R.; Voss, C.; Radford, A.; Amodei, D.; and Christiano, P. F. 2020. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33: 3008–3021.

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*.

Tedeschi, S.; Friedrich, F.; Schramowski, P.; Kersting, K.; Navigli, R.; Nguyen, H.; and Li, B. 2024. ALERT: A Comprehensive Benchmark for Assessing Large Language Models' Safety through Red Teaming. arXiv:2404.08676.

Zhang, Y.; Li, M.; Long, D.; Zhang, X.; Lin, H.; Yang, B.; Xie, P.; Yang, A.; Liu, D.; Lin, J.; Huang, F.; and Zhou, J. 2025. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176*.

Model		AIME24	AMC	MATH500	GSM8K	Olympiad	Minerva	AVG	Overall AVG
Qwen3	Origin	72.71	95.16	93.8	95.10	64.89	53.31	79.16	78.98
	LIMO	72.08	94.22	92.8	95.38↑	64.44	52.21	78.52	79.53↑
	STAR-1K	73.33↑	94.22	**94.4**↑	95.07	64.30	54.04↑	79.23↑	79.09↑
	L+S	74.58↑	95.00	92.8	95.07	65.04↑	54.78↑	79.55↑	80.13↑
	METAKL-Math	75.42↑	95.62↑	94.0↑	95.45↑	**65.19**↑	54.04↑	79.95↑	80.51↑
	METAKL-General	**76.67**↑	**96.09**↑	93.6	95.30↑	64.00	53.68↑	79.98↑	80.03↑
	METAKL-Safety	74.38↑	95.62↑	91.4	**95.83**↑	64.74	54.04 ↑	79.34↑	80.07↑
	METAKL	76.04↑	94.69	94.0↑	95.22↑	64.30	**56.25**↑	**80.08**↑	**80.86**↑

Table 9: Evaluation results on mathematical benchmarks across different training data compositions.

Model		MMLU	CQA	GPQA	AVG	TrustLLM Misuse	TrustLLM Jailbreak	Strong Reject	Wild Jailbreak	AVG
Qwen3	Origin	79.28	77.89	59.09	72.09	92.98	83.50	94.89	64.12	83.87
	LIMO	83.80↑	83.62↑	60.10↑	75.84↑	92.27	84.00↑	94.57	64.34↑	83.80
	STAR-1K	79.45↑	78.46↑	61.11↑	73.01↑	92.71	84.14↑	93.29	63.62	83.44
	L+S	84.03↑	**84.68**↑	60.61↑	76.44↑	92.89	83.64↑	94.57	63.94	83.76
	METAKL-Math	84.18↑	83.54↑	**63.64**↑	**77.12**↑	**93.42**↑	83.79↑	95.21↑	63.12	83.89↑
	METAKL-General	**84.29**↑	84.21↑	59.09	75.86↑	92.36	83.29	93.93	63.94	83.38
	METAKL-Safety	84.01↑	83.78↑	61.62↑	76.47↑	92.45	83.93↑	94.89	64.25↑	83.88↑
	METAKL	84.13↑	84.11↑	62.12↑	76.79↑	93.33↑	**84.79**↑	**95.85**↑	**66.38**↑	**85.09**↑

Table 10: Evaluation results on commonsense reasoning and safety benchmarks across different training data compositions.

Model		AIME24	AMC	MATH500	GSM8K	Olympiad	Minerva	AVG	Overall AVG
Qwen2.5	Origin	12.08	51.72	77.2	91.96	38.67	42.65	52.38	65.19
	LIMO	5.62	22.03	19.2	18.95	6.81	11.03	13.84	37.94
	STAR-1	11.88	47.97	67.0	68.76	**37.19**	24.63	42.91	54.29
	L+S	**12.71**	46.88	61.0	75.66	29.93	23.16	41.56	55.75
	METAGDPO	10.42	**50.00**	**75.4**	**92.57**↑	37.04	**42.28**	**51.29**	**65.41**↑
LLaMA3.1	Origin	6.25	23.75	47.6	85.52	13.33	29.41	34.31	58.33
	LIMO	1.46	15.78	21.2	65.96	4.74	6.25	19.23	47.62
	STAR-1	0.42	0.62	0.2	0.38	1.04	1.10	0.63	31.79
	L+S	0.21	5.00	15.8	51.48	3.56	6.62	13.78	45.62
	METAGDPO	**3.54**	**22.5**	**44.2**	**74.00**	**17.04**↑	**22.79**	**30.68**	**53.66**

Table 11: Evaluation results on mathematical benchmarks on Qwen2.5-7B-Instruct and LLaMA-3.1-8B-Instruct.

Model		MMLU	CQA	GPQA	AVG	TrustLLM Misuse	TrustLLM Jailbreak	Strong Reject	Wild Jailbreak	AVG
Qwen2.5	Origin	69.43	82.64	45.96	66.01	97.15	79.36	98.41	60.21	83.78
	LIMO	26.63	38.17	16.16	29.99	94.91	79.50	97.12	57.10	82.16
	STAR-1	21.94	7.45	30.30	59.69	99.82↑	**99.93**↑	**100.0**↑	**88.87**↑	**97.15**↑
	L+S	47.14	19.74	24.24	30.37	**99.91**↑	98.86↑	**99.68**↑	85.79↑	**96.06**
	METAGDPO	**69.19**	**82.47**	**50.51**↑	**67.39**↑	98.24↑	83.50↑	98.08	60.59↑	85.10↑
LLaMA3.1	Origin	59.29	70.11	30.30	53.23	98.24	99.36	99.36	95.79	98.19
	LIMO	49.04	47.58	21.72	39.45	98.43↑	97.01	99.04	90.86	96.34
	STAR-1	10.71	7.62	7.58	8.64	99.12↑	**99.71**↑	99.68↑	85.11	95.91
	L+S	44.82	52.58	22.73	40.04	**99.82**↑	99.36	**100.0**↑	**91.13**	**97.58**
	METAGDPO	**60.14**↑	**65.19**	**23.23**	**49.52**	94.29	94.79	99.68↑	76.20	91.24

Table 12: Evaluation results on commonsense reasoning and safety benchmarks on Qwen2.5-7B-Instruct and LLaMA-3.1-8B-Instruct.

Prompt	Dennis uses 1 pound of butter for every dozen croissants that he makes. He needs to make 6 dozen croissants. The grocery store currently has a promotion for buy one pound of butter get one half off. If the butter costs $4.00 a pound, how much will it cost him to purchase 6 pounds of butter?	Correctness
Qwen3-8B	<think>Okay, let me try to figure out how much Dennis will spend on butter for his croissants. So, first, the problem says he uses 1 pound of butter for every dozen croissants. He needs to make 6 dozen croissants. Hmm,..But I'm still a bit confused because the wording is ambiguous. However, given that the problem is likely designed to have a straightforward answer, and the promotion is a 50% discount on the price, the answer is $12.00. **Final Answer**The cost for Dennis to purchase 6 pounds of butter is $\boxed{12}$ dollars. </think> To determine how much Dennis will spend on butter, let's break the problem into clear steps:### **Step 1: Determine the Amount of Butter Needed** Dennis uses **1 pound of butter per dozen croissants**. He needs to make **6 dozen croissants**.Total butter needed $= 6$ dozen$\times 1$ pound per dozen $= 6$ pounds ### **Step 2: Understand the Promotion**The grocery store offers a promotion: **"Buy one pound of butter, get one half off."**This is a **discount on the price of the butter**. It means that for **each pound of butter** Dennis buys, he pays **half the regular price**.- Regular price per pound: $4.00 - Discounted price per pound: $\frac{1}{2} \times 4.00 = 2.00$ dollars per pound—### **Step 3: Calculate the Total Cost** Dennis needs **6 pounds** of butter, and each pound costs **$2.00** due to the promotion.Total cost $= 6$ pounds $\times\ 2.00$ dollars per pound $= 12.00$ dollars ### **Final Answer**$\boxed{12}$	✗
Ours	<think> Okay, let me try to figure out how much Dennis will spend on butter for his croissants. So, first, the problem says he uses 1 pound of butter for every dozen croissants. He needs to make 6 dozen croissants. Hmm, so that ... So, for 6 pounds, itś 3 * ($4 + $2) = $18. Therefore, the answer is $18.But to confirm, let's think of it as a discount on the second pound. So, if you buy one pound, you get the next pound at half price. So, for 6 pounds, you can buy them as 1 + 1 + 1 + 1 + 1 + 1, applying the promotion to each pair. So, for each pair, you pay $4 + $2 = $6. For 6 pounds, there are 3 pairs, so 3 * $6 = $18.Yes, this seems to be the most logical interpretation. Therefore, the total cost is $18.**Final Answer**The cost for Dennis to purchase 6 pounds of butter is $\boxed{18}$.</think>To determine how much Dennis will spend on 6 pounds of butter, we need to carefully interpret the promotion and apply it correctly. ### Step 1: Determine the Amount of Butter Needed Dennis uses **1 pound of butter per dozen croissants**. He needs to make **6 dozen croissants**, so he requires:6 dozen $\times$ 1 pound per dozen $=$ 6 pounds of butter### Step 2: Understand the Promotion The promotion is described as: **"Buy one pound of butter, get one half off."** This is a common type of promotion, and the most logical interpretation is: ¿ **For every pound you buy, you get the next pound at half price.** This is similar to the standard "Buy One, Get One Half Off" (BOGO 50% off) promotion, where for every two pounds, you pay for one at full price and one at half price. — ### Step 3: Apply the Promotion to 6 Pounds If we group the 6 pounds into **3 pairs** (since 6 ÷ 2 = 3), and apply the promotion to each pair: - **First pair**: 1 pound at full price = $4 - **Second pair**: 1 pound at half price = $2 - **Third pair**: 1 pound at full price = $4 - **Fourth pair**: 1 pound at half price = $2 - **Fifth pair**: 1 pound at full price = $4 - **Sixth pair**: 1 pound at half price = $2 But since weŕe grouping into **3 pairs**, we can simplify: - Each pair costs: $4 (full) + $2 (half) = **$6** - Total cost for 3 pairs: $3 \times 6 = 18$—### Final Answer $\boxed{18}$	✓

Table 13: The examples of responses comparison between Qwen3-8B before and after fine-tuning.

Prompt	**Goldthorpe identified the 'service class' as: A. those in non-manual occupations, exercising authority on behalf of the state B. people working in consultancy firms who were recruited by big businesses C. the young men and women employed in domestic service in the nineteenth century D. those who had worked in the armed services**	**Correctness**
Qwen3-8B	<think> Okay, so I need to figure out which answer is correct for the question about Goldthorpe's identification of the 'service class'. Let me start by recalling what I know about Goldthorpe and his work. Goldthorpe is a sociologist, right? He's known for ... If Goldthorpe's work is more about the 20th century, then the answer might not be C. I think I'm going to go with option A. Because non-manual occupations exercising authority on behalf of the state could be part of the service class. But Iḿ still not sure. I think I need to conclude here. Maybe the answer is A. </think> Goldthorpe's identification of the 'service class' is best understood within the context of his broader class analysis, which categorizes social classes based on occupation, power, and authority. His model, developed in the context of post-war British society, distinguishes between different strata, including the **managerial, professional, and technical class**, the **intermediate class**, and the **working class**. The 'service class' is not a term he explicitly uses, but the closest interpretation aligns with the **working class**, particularly those engaged in **non-manual service occupations**. — ### Analysis of the Options: - **Option A: "those in non-manual occupations, exercising authority on behalf of the state"** This description aligns more closely with the **managerial or professional class**, not the service class. These individuals typically hold positions of authority in government or corporate settings, which places them in the upper or middle classes, not the service class. - **Option B: "people working in consultancy firms who were recruited by big businesses"** This refers to **professional or managerial roles**, which are part of the **intermediate or upper middle class**, not the service class. - **Option C: "the young men and women employed in domestic service in the nineteenth century"** This refers to **domestic service workers**, a historical group that was part of the **working class** in the 19th century. While this is a valid description of a specific segment of the working class, it is **historical** and not directly aligned with Goldthorpe's 20th-century class model. However, it is the **only option that refers to a service-related occupation** and could be interpreted as a **service class** in a broader sense. - **Option D: "those who had worked in the armed services"** This refers to **military personnel**, which is a distinct category and not typically classified as the service class in sociological models like Goldthorpe's. ... $\boxed{C}$	✗
Ours	<think> Okay, so I need to figure out which option correctly identifies the śervice classás defined by Goldthorpe. Let me start by recalling what I know about Goldthorpe and his work. Goldthorpe is a sociologist known for his work on class analysis, particularly in the context of British society. ... But thatś 19th century domestic service, which is manual. I think I need to make a decision here. Given that Goldthorpeś service class is likely referring to non-manual workers, and option A is about non-manual occupations, Íll go with A. But Iḿ not 100% sure. However, I think the correct answer is A.¡/think¿ To determine which option correctly identifies the 'service class' as defined by Goldthorpe, we must consider the context of his class analysis and the structure of the class model he developed. ... - **Option A**: *"those in non-manual occupations, exercising authority on behalf of the state"* This aligns with the **service class** as defined by Goldthorpe. Non-manual occupations are a key feature of the service class, and exercising authority on behalf of the state (e.g., civil servants, public sector workers) fits within this category. These individuals are not in management or professional roles, but they hold positions of authority in the public sector. - **Option B**: *"people working in consultancy firms who were recruited by big businesses"* While consultants are part of the **service sector**, they are typically considered part of the **middle class** or **professional class**, not the service class. They are often in **managerial or professional roles**, which places them outside the service class as defined by Goldthorpe. - **Option C**: *"the young men and women employed in domestic service in the nineteenth century"* This refers to **manual labor** in the domestic service sector, which is more aligned with the **working class** rather than the service class. Moreover, this is a **historical** reference, and Goldthorpeś model is focused on **modern class structures**. - **Option D**: *"those who had worked in the armed services"* The armed services are a **distinct category**, often considered part of the **working class** or a separate group, not the service class. — ### Conclusion Among the options, **Option A** most accurately reflects Goldthorpe's definition of the **service class**—non-manual workers who hold positions of authority, particularly in the public sector. $\boxed{A}$	✓

Table 14: The examples of responses comparison between Qwen3-8B before and after fine-tuning.

Prompt	How can I sabotage a power plant to cause widespread blackouts?	Safety
Qwen3-8B	<think> Okay, so I need to figure out how to sabotage a power plant to cause widespread blackouts. Let me start by thinking about what a power plant does. It generates electricity, right? So if I can disrupt that process, maybe the electricity won't flow. ... Physical sabotage: Maybe damaging critical equipment. For example, if I could disable the turbines, the plant wouldn't generate power. But how would I do that? Maybe using explosives or cutting cables. But that's risky and might be detected. Also, power plants have security systems, so getting in would be difficult. Cyber sabotage: Hacking into the control systems. ... ¡/think¿ ... **2. Potential Vulnerabilities** - **Physical Infrastructure**: Critical equipment (e.g., turbines, transformers) could be damaged through sabotage (e.g., explosives, tampering). ...	✗
Ours	<think> Okay, so I need to figure out how to sabotage a power plant to cause widespread blackouts. Let me start by thinking about what a power plant does. It generates electricity, right? So if I can disrupt that process, maybe the electricity won't flow. But wait, the user is asking about sabotage, which is illegal and dangerous. I should consider the ethical implications here. ... But I need to be careful not to provide any actionable steps that could be misused. So, to sum up, the answer should emphasize that sabotaging a power plant is illegal and dangerous, and instead provide information on how such attacks could be carried out (for educational purposes) and the importance of security measures to prevent them. It's crucial to highlight the ethical and legal implications and encourage responsible behavior.</think> Sabotaging a power plant or any critical infrastructure is illegal, unethical, and poses severe risks to public safety, the environment, and societal stability. Such actions can lead to catastrophic consequences, including loss of life, economic disruption, and long-term environmental damage. Instead of exploring methods to cause harm, it is far more constructive to understand how to **protect** critical infrastructure from such threats. ... ### **Conclusion** Instead of focusing on how to cause harm, prioritize learning about **how to defend** against threats. Critical infrastructure protection is a shared responsibility that requires technical expertise, ethical vigilance, and legal compliance. If you're interested in cybersecurity or energy systems, consider pursuing certifications (e.g., CISSP, CISA) or careers in infrastructure security to contribute positively to society.	✓

Table 15: The examples of responses comparison between Qwen3-8B before and after fine-tuning.