DISCO Balances the Scales: Adaptive Domain- and Difficulty-Aware Reinforcement Learning on Imbalanced Data

Yuhang Zhou [1*] **Jing Zhu** [3*] **Shengyi Qian** [3] **Zhuokai Zhao** [4] **Xiyao Wang** [1]
Xiaoyu Liu [1] **Ming Li** [1] **Paiheng Xu** [1] **Wei Ai** [1] **Furong Huang** [1,2]
[1] University of Maryland, College Park [2] Capital One
[3] University of Michigan, Ann Arbor [4] University of Chicago
{tonyzhou, xywang, xliu1231, paiheng, minglii, aiwei, furongh}@umd.edu
{jingzhuu, syqian}@umich.edu zhuokai@uchicago.edu

Abstract

Large Language Models (LLMs) are increasingly aligned with human preferences through Reinforcement Learning from Human Feedback (RLHF). Among RLHF methods, Group Relative Policy Optimization (GRPO) has gained attention for its simplicity and strong performance, notably eliminating the need for a learned value function. However, GRPO implicitly assumes a balanced domain distribution and uniform semantic alignment across groups—assumptions that rarely hold in real-world datasets. When applied to multi-domain, imbalanced data, GRPO disproportionately optimizes for dominant domains, neglecting underrepresented ones and resulting in poor generalization and fairness. We propose **D**omain-**I**nformed **S**elf-**C**onsistency Policy **O**ptimization (**DISCO**), a principled extension to GRPO that addresses inter-group imbalance with two key innovations. *Domain-aware reward scaling* counteracts frequency bias by reweighting optimization based on domain prevalence. *Difficulty-aware reward scaling* leverages prompt-level self-consistency to identify and prioritize uncertain prompts that offer greater learning value. Together, these strategies promote more equitable and effective policy learning across domains. Extensive experiments across multiple LLMs and skewed training distributions show that DISCO improves generalization, outperforms existing GRPO variants by 5% on Qwen3 models, and sets new state-of-the-art results on multi-domain alignment benchmarks.

1 Introduction

Aligning large language models (LLMs) with human preferences is a central challenge in modern AI systems [12, 20, 23]. Reinforcement Learning from Human Feedback (RLHF) has become the dominant approach for fine-tuning LLMs toward desirable behavior, enabling alignment with nuanced human intent [13, 16, 28, 14, 29, 22, 21]. Within this framework, Group Relative Policy Optimization (GRPO) [16] offers a promising alternative to value-based methods, simplifying training while achieving strong performance.

Despite its advantages, GRPO faces a significant challenge when applied to multi-domain datasets. While GRPO effectively removes the need for a value network and mitigates within-group variance, it implicitly assumes that prompt groups are sampled uniformly and that reward signals are semantically aligned across domains. However, this assumption often fails in practice. Real-world datasets are typically imbalanced, with a few dominant domains and many underrepresented ones [30]. Optimization gradients become skewed toward high-frequency domains, starving rare domains of

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Efficient Reasoning.

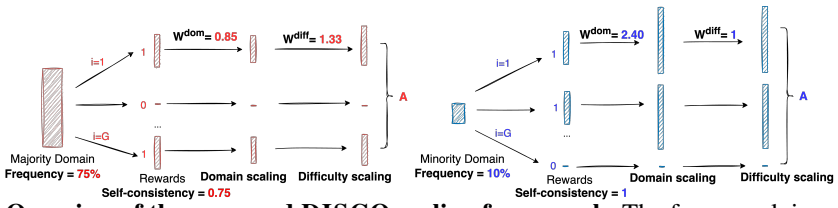

Figure 1: **Overview of the proposed DISCO scaling framework.** The framework is composed of two strategies to enhance GRPO's robustness: (1) domain-aware scaling, which reweights prompt groups based on domain frequency, and (2) difficulty-aware scaling, which encourages the model to focus more on uncertain samples based on self-consistency scores. w^{dom} and w^{diff} denote the domain and difficulty weight. G=Group size, A=Advantage.

	IMDB	GSM8K	Math	NQ	ARC	Avg.
Balanced	89.60	57.94	23.82	14.90	47.44	**46.74**
Math heavy	87.80	**58.98**	**24.38**	13.96	45.99	46.22
IMDB heavy	**90.90**	56.88	23.54	13.85	45.90	46.21
NQ heavy	89.40	56.22	23.36	**16.57**	46.58	46.43
ARC heavy	88.80	55.29	22.46	14.16	**48.29**	45.80

Table 1: Performance (Exact Match (EM) Accuracy %) of Qwen2.5-0.5B trained with GRPO ($G = 4$) under various domain-heavy training distributions.

learning signal. This results in models that generalize poorly to critical low-resource domains and amplify existing data biases. Data augmentation offers one workaround but introduces substantial overhead in generating high-quality synthetic prompts [19].

To address the inter-group imbalance in GRPO, we propose **D**omain-**I**nformed **S**elf-**C**onsistency Policy **O**ptimization (**DISCO**), an enhanced framework designed to promote equitable learning across imbalanced multi-domain datasets. DISCO introduces two complementary strategies to improve generalization under distributional skew: *domain-aware* and *difficulty-aware* reward scaling.

Domain-aware scaling reweights prompt groups inversely by their frequency, reducing over-optimization on dominant domains while amplifying learning signals from underrepresented ones. *Difficulty-aware scaling* leverages prompt-level self-consistency, an intrinsic signal in GRPO, to identify and upweight prompts where the policy exhibits high uncertainty or inconsistent responses. Since not all prompts are equally challenging, treating them uniformly can lead the policy to overfit on easy examples while neglecting harder, more informative ones. By prioritizing uncertain prompts, this strategy guides the model to focus its learning on cases that offer a greater signal for improvement.

By integrating these two forms of adaptive scaling, DISCO enables all domains, regardless of their prevalence, to meaningfully contribute to policy optimization. As a result, it effectively mitigates GRPO's inter-group imbalance and achieves state-of-the-art performance across diverse LLM architectures and training distributions.Our contributions are summarized as follows:

- **Systematic Analysis of GRPO under data imbalance:** We perform the first systematic analysis on GRPO's inherent vulnerability to dataset imbalance.
- **Strategic Framework:** Our novel DISCO framework introduces a powerful integration of domain-aware and difficulty-aware reward scaling strategies.
- **SoTA Performance:** Our comprehensive empirical evaluations confirm that DISCO improves existing GRPO algorithms by 5% on Qwen3 models across diverse benchmarks and sets new standards for generalization performance.

2 Domain- and Difficulty-Aware Scaling

2.1 Background and Motivation

While GRPO normalizes advantages locally within prompt groups, its global optimization trajectory can be unduly influenced by the frequency of domains in the training data, a vulnerability already highlighted in Section 1. To empirically demonstrate this limitation, we performed an experiment training Qwen2.5-0.5B [23] with GRPO using datasets featuring distinct domain compositions and

Dataset	Task Domain
IMDB [11]	Text Classification (TC)
GSM8K [2]	Mathematical Reasoning
MATH [6]	Mathematical Reasoning
Natural Questions (NQ) [9]	Open-domain QA
ARC [1]	Multi-step Reasoning QA

(a) Test datasets and their corresponding task domains. All evaluations are conducted using exact match (EM) accuracy.

Setup	Math	NQ	ARC	IMDB
Balanced	25%	25%	25%	25%
Math-heavy	75%	8.3%	8.3%	8.4%
NQ-heavy	8.3%	75%	8.3%	8.4%
ARC-heavy	8.3%	8.3%	75%	8.4%
IMDB-heavy	8.3%	8.3%	8.4%	75%

(b) Training Distribution by Domain (Proportions)

Table 2: Overview of evaluation and training datasets. "Heavy" settings allocate 75% of training prompts to a single domain, with the remaining 25% distributed equally among the others. For the math domain, we sample from the MetaMath dataset [24] for training, while for all other domains, we use the training portion of each corresponding evaluation benchmark.

present the performance in Table 1. Specifically, we trained multiple models, each on 4,000 examples. For each model, the training data composition was intentionally skewed by heavily weighting one domain while underrepresenting the others.

As shown in Table 1, domain-heavy training biases GRPO performance: models excel in the overrepresented domain but underperform on others compared to a balanced setup. The Math-heavy model performs best on math tasks, but its average score across all domains is lower than the balanced model's.These results confirm that GRPO lacks a mechanism for inter-group calibration, and its optimization may become biased toward more frequently sampled domains.

2.2 Domain-Aware Scaling

Vanilla GRPO's group-level normalization operates independently of a prompt group's originating domain frequency. In imbalanced datasets (Section 2.1), this means high-frequency domains disproportionately influence the aggregated optimization gradient, potentially marginalizing low-frequency domains. To mitigate this issue, we introduce a **domain-aware reward scaling** strategy. For each prompt group q from domain d, we apply a domain weight $w^{\text{dom}}(q)$ to rescale its rewards$r_i^{\text{scaled}} = r_i \cdot w^{\text{dom}}(q)$, and compute the group-level advantage as:

$$A_i = r_i^{\text{scaled}} - \bar{r}^{\text{scaled}}, \tag{1}$$

where $\bar{r}^{\text{scaled}} = \frac{1}{G}\sum_{j=1}^{G} r_j^{\text{scaled}}$ is the group mean.

Note that we do not apply standard deviation normalization, in order to preserve the absolute scaling effect of the domain weights. This design allows domain frequency to directly modulate the magnitude of the advantage signal, enabling rarer domains to exert a stronger influence during policy updates.

2.3 Difficulty-Aware Scaling

The GRPO algorithm provides a natural mechanism for estimating prompt-level difficulty.

To capture this, we define the self-consistency (SC) score for prompt q as: $\text{SC}(q) = \frac{1}{G}\sum_{i=1}^{G} r_i$, and define the difficulty weight as:$w^{\text{diff}}(q) = \frac{1}{\text{SC}(q)+\epsilon'}$, where ϵ' is a small constant to ensure numerical stability.

Note that when all generations in a group are incorrect (i.e., $\text{SC}(q) = 0$), w^{diff} becomes large. However, this does not lead to instability, as all advantages will be zero based on Equation 1, and thus no policy update occurs. This mechanism encourages the model to focus more on prompts it finds uncertain, while ignoring uniformly poor or trivially easy cases.

Combining both components, we compute the final scaled reward of DISCO as:

$$r_i^{\text{scaled}} = r_i \cdot w^{\text{dom}}(q) \cdot w^{\text{diff}}(q), \tag{2}$$

where $w^{\text{dom}}(q)$ and $w^{\text{diff}}(q)$ are the domain- and difficulty-based weights for prompt group q, respectively. The final advantage is then computed using Equation 1. This formulation retains the structure of GRPO while enhancing it with principled scaling to reflect domain imbalance and prompt difficulty.

3 Results

We evaluate DISCO impact through comparisons with baseline methods and targeted ablation studies. Details about experiment setups can be found in the appendix.

Our method outperforms baselines across different models and training distributions. To evaluate the effectiveness of DISCO, we compare DISCO against two key baselines: Naive GRPO, the original formulation without reward rescaling, and Dr. GRPO [10].

Focusing on Table 2, we first note that all GRPO variants evaluated consistently yield substantial improvements over their respective base models, confirming the effectiveness of GRPO-based alignment [4]. Turning to the comparison between the GRPO variants, we observe from the overall average performance ('Avg.' column) that **DISCO achieves the highest overall average score on 5 out of the 6 models**.

While Dr. GRPO occasionally improves over Naive GRPO, the consistent advantages of DISCO underscore the benefit of incorporating explicit domain and difficulty signals. These findings highlight the strength of our joint scaling approach in enhancing GRPO alignment, particularly in navigating performance trade-offs introduced by domain imbalance.

Model	Math-heavy	IMDB-heavy	NQ-heavy	ARC-heavy	Avg.
		Qwen2.5-0.5B			
Base	42.80				
Naive GRPO	46.22	46.21	46.43	45.80	46.17
Dr GRPO	46.90	46.22	47.84	47.62	47.14
Ours	**47.92**	**47.67**	**47.91**	**47.70**	**47.80**
		Qwen2.5-1.5B			
Base	60.06				
Naive GRPO	**64.81**	64.91	65.15	65.02	**64.97**
Dr GRPO	64.76	64.75	64.72	65.16	64.85
Ours	64.12	**64.93**	**65.26**	**65.35**	64.91
		Qwen2-0.5B			
Base	43.18				
Naive GRPO	44.05	43.76	43.97	**45.35**	44.28
Dr GRPO	**45.48**	44.31	43.14	44.87	44.45
Ours	43.98	**44.83**	**44.50**	45.07	**44.60**
		LLaMA3.2-1B			
Base	22.31				
Naive GRPO	27.61	29.94	21.76	25.86	26.29
Dr GRPO	22.24	29.72	22.17	29.74	25.97
Ours	**28.97**	**30.02**	**22.19**	**30.56**	**27.94**
		Qwen1.5-MoE			
Base	45.57				
Naive GRPO	66.05	64.03	**66.04**	65.46	65.40
Dr GRPO	62.46	64.85	64.50	65.58	64.35
Ours	**66.37**	**65.59**	65.19	**66.00**	**65.79**
		Qwen3-0.6B			
Base	25.59				
Naive GRPO	41.99	34.83	34.72	34.31	36.46
Dr GRPO	42.08	33.23	36.76	32.91	36.25
Ours	**45.07**	**35.64**	**44.44**	**38.28**	**40.86**

Figure 2: Average accuracy under each domain-heavy training setup for different methods. Scores are averaged over five task-specific datasets per domain. Bold indicates the best-performing method.

4 Related Work

Reinforcement Learning from Human Feedback (RLHF) aligns Large Language Models (LLMs) with human preferences, commonly using Proximal Policy Optimization (PPO) [13, 15]. A popular successor, GRPO, simplifies this process by eliminating the need for a value function [16, 4]. Recent work has focused on improving GRPO's training dynamics by addressing challenges such as length bias [10, 27] and instability [25, 26, 17]. While these methods target training algorithms, our work is the first to examine GRPO's vulnerability to dataset imbalance. Our proposed strategies are complementary and can be integrated with existing GRPO variants.

Data Imbalance in NLP is a well-known challenge where skewed data distributions cause models to underperform on minority classes [7, 5]. Common mitigation strategies include data augmentation and loss adjustment [30, 18]. Our approach is analogous to loss adjustment, but we directly modify the rewards within the GRPO framework rather than reweighting the loss function. Unlike data augmentation, which can incur additional computational costs, our policy modifications are cost-neutral and can be used in conjunction with data-based techniques.

5 Conclusion

In this work, we investigate the vulnerability of GRPO to domain imbalance in multi-domain alignment and introduce DISCO that combines domain and difficulty-aware reward scaling to enable GRPO to better handle skewed training distributions.

References

[1] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*, 2018.

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

[3] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[4] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[5] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

[6] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

[7] Sophie Henning, William Beluch, Alexander Fraser, and Annemarie Friedrich. A survey of methods for addressing class imbalance in deep-learning based natural language processing. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 523–540, 2023.

[8] Jian Hu, Xibin Wu, Zilin Zhu, Xianyu, Weixun Wang, Dehao Zhang, and Yu Cao. Openrlhf: An easy-to-use, scalable and high-performance rlhf framework. *arXiv preprint arXiv:2405.11143*, 2024.

[9] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. *arXiv preprint arXiv:1906.00300*, 2019.

[10] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

[11] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150, 2011.

[12] OpenAI. Gpt-4 technical report, 2023.

[13] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[14] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

[15] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[16] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[17] Yi Su, Dian Yu, Linfeng Song, Juntao Li, Haitao Mi, Zhaopeng Tu, Min Zhang, and Dong Yu. Crossing the reward bridge: Expanding rl with verifiable rewards across diverse domains. *arXiv e-prints*, pages arXiv–2503, 2025.

[18] Yanmin Sun, Mohamed S Kamel, Andrew KC Wong, and Yang Wang. Cost-sensitive boosting for classification of imbalanced data. *Pattern recognition*, 40(12):3358–3378, 2007.

[19] Naama Tepper, Esther Goldbraich, Naama Zwerdling, George Kour, Ateret Anaby Tavor, and Boaz Carmeli. Balancing via generation for multi-class text classification improvement. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1440–1452, 2020.

[20] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[21] Chaoqi Wang, Zhuokai Zhao, Yibo Jiang, Zhaorun Chen, Chen Zhu, Yuxin Chen, Jiayi Liu, Lizhu Zhang, Xiangjun Fan, Hao Ma, et al. Beyond reward hacking: Causal rewards for large language model alignment. *arXiv preprint arXiv:2501.09620*, 2025.

[22] Chaoqi Wang, Zhuokai Zhao, Chen Zhu, Karthik Abinav Sankararaman, Michal Valko, Xuefei Cao, Zhaorun Chen, Madian Khabsa, Yuxin Chen, Hao Ma, et al. Preference optimization with multi-sample comparisons. *arXiv preprint arXiv:2410.12138*, 2024.

[23] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[24] Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*, 2023.

[25] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, et al. Dapo: An open-source llm reinforcement learning system at scale, 2025. *URL https://arxiv. org/abs/2503.14476*.

[26] Kaichen Zhang, Yuzhong Hong, Junwei Bao, Hongfei Jiang, Yang Song, Dingqian Hong, and Hui Xiong. Gvpo: Group variance policy optimization for large language model post-training. *arXiv preprint arXiv:2504.19599*, 2025.

[27] Xiaojiang Zhang, Jinghui Wang, Zifei Cheng, Wenhao Zhuang, Zheng Lin, Minglei Zhang, Shaojie Wang, Yinghan Cui, Chao Wang, Junyi Peng, et al. Srpo: A cross-domain implementation of large-scale reinforcement learning on llm. *arXiv preprint arXiv:2504.14286*, 2025.

[28] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.

[29] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023.

[30] Yuhang Zhou, Jing Zhu, Paiheng Xu, Xiaoyu Liu, Xiyao Wang, Danai Koutra, Wei Ai, and Furong Huang. Multi-stage balanced distillation: Addressing long-tail challenges in sequence-level knowledge distillation. *Findings of EMNLP*, 2024.

A Domain Weight Variants

Domain Weight Variants. We explore three definitions of the domain weight based on domain frequency. Let p_d denote the proportion of prompt groups from domain d: $p_d = \frac{N_d}{\sum_{d'} N_{d'}}$, where N_d is the number of prompts from domain d.

We consider the following three scaling variants:

$$\textbf{v1 (log):} \quad w^{\text{dom}} = \log\left(1 + \frac{1}{p_d}\right) \tag{3}$$

$$\textbf{v2 (log-squared):} \quad w^{\text{dom}} = \left[\log\left(1 + \frac{1}{p_d}\right)\right]^2 \tag{4}$$

$$\textbf{v3 (inverse):} \quad w^{\text{dom}} = \frac{1}{p_d} \tag{5}$$

These variants are chosen to represent a spectrum of upweighting strengths. v1 (log) is hypothesized to provide a tempered yet significant boost to underrepresented domains; the logarithmic function

naturally moderates the impact of extreme p_d values, potentially enhancing training stability. v2 (log-squared) represents a more assertive non-linear scaling. v3 (inverse) offers the most direct and aggressive form of upweighting, making domain weights sharply inversely proportional to their frequency. This systematic variation from a more conservative (v1) to a highly aggressive (v3) approach allows us to study how different levels and types of domain correction affect training dynamics and model performance.

B Implementation Details

We train models using the `OpenRLHF` framework [8] with GRPO optimization. Each model is fine-tuned for 1 epoch. The rollout and training batch sizes are both set to 64, with micro-batch sizes of 8 and 4, respectively. We use a maximum prompt and generation length of 1024 tokens. The KL penalty is initialized at $1e{-3}$ and estimated using the K3 estimator. All models use a learning rate of $1e{-6}$.

All evaluations are conducted using zero-shot inference, with models generating answers without access to in-context examples. For each task, we apply a fixed, task-specific prompt template for both training and evaluation. The prompts are designed to clearly define the task instruction and input format.

C Experiment Setups

Dataset Setup. We evaluate across four task domains: IMDB (text classification), GSM8K and MATH (math problem solving), NATURAL QUESTIONS (open-domain QA), and ARC (reasoning QA). Our training dataset consists of 4,000 examples, as detailed in Table 2.

Baseline Methods. We compare our method against the following baselines: (1) **Base Model**, the pretrained model without any fine-tuning; (2) **Naive GRPO**, which applies the original group-relative optimization without any reward reweighting; and (3) **Dr. GRPO** [10], which removes the length and standard deviation normalization. While other GRPO variants address issues like length bias or training instability [25, 27, 26], they are not included as baselines as our work targets GRPO's vulnerability to domain imbalance. In addition, we conduct ablation studies to isolate the contributions of domain-aware and difficulty-aware components.

Model Setup. We evaluate our method across a diverse set of language models. These include **Qwen2.5-0.5B**, **Qwen2.5-1.5B**, **Qwen2-0.5B**, **Qwen3-0.6B**, and **Qwen1.5-MoE-A2.7B** (14B total parameters, 2.7B activated) [23], as well as **LLaMA3.2-1B** [3]. This selection spans both dense and mixture-of-experts (MoE) architectures and covers a range of model capacities from 0.5B to 14B parameters. For the GRPO group size, we use $G = 2, 4, 8, 16$ depending on the experiment. Additional implementation details are provided in Appendix B.

Table 3 presents the prompt templates used across the five datasets.

Performance Breakdown by Dataset. To illustrate how our method navigates domain trade-offs, we break down Qwen3-0.6B performance by evaluation dataset under each domain-heavy setting (Figure 3). This case study compares Naive GRPO, Dr. GRPO, and DISCO on the individual datasets.

The results for Qwen3-0.6B reveal a clear pattern where DISCO significantly boosts minority domain performance, sometimes involving a trade-off with majority domain scores. On majority domains (those comprising 75% of training data), DISCO performance varies compared to Naive GRPO. For instance, when trained NQ-heavy, DISCO improves performance on the NQ dataset (12.30% vs 11.47%). However, when trained Math-heavy, it scores lower on both MATH (41.68% vs 43.40%) and GSM8K (55.20% vs 58.91%). Similarly, under ARC-heavy training, the ARC score is slightly lower (48.29% vs 49.91%).

In contrast, DISCO demonstrates substantial and consistent improvements on tail domains. The most dramatic gains are seen in the NQ-heavy setting: the IMDB score jumps to 82.60% (from 57.90% for Naive GRPO), GSM8K increases to 42.72% (from 34.86%), and MATH rises to 35.70% (from

Dataset	Prompt Template
MATH	`Below is a math problem. Provide a detailed, step-by-step solution. ### Problem: {problem} ### Answer:`
GSM8K	Same as MATH
IMDB	`Below is a movie review. Determine the sentiment of the review as Positive or Negative. ### Review: {review} ### Answer:`
NQ	`Below is a question that requires a concise and accurate answer. Provide a detailed explanation before concluding with the correct answer. ### Question: {question} ### Answer:`
ARC	`Below is a question with multiple-choice answers. Choose the correct option based on your reasoning. ### Question: {question} ### Choices: A. {choice_A} B. {choice_B} ... ### Answer:`

Table 3: Prompt templates used for zero-shot inference. Each template is applied consistently during both training and evaluation.

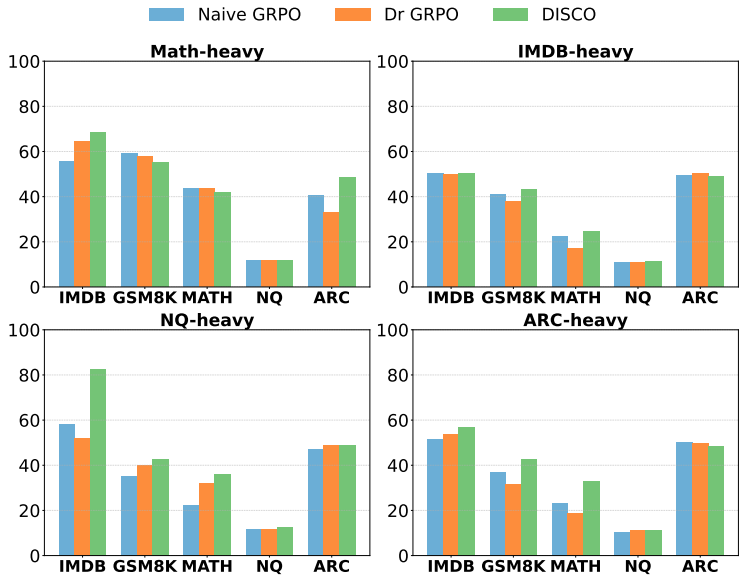

Figure 3: Qwen3-0.6B performance (EM Accuracy %) breakdown by the dataset under four domain-heavy training conditions (one condition per panel/subplot). Bar groups show results on individual datasets.

22.34%). Significant minority domain recovery is also evident in other settings, such as the gains on GSM8K (+2.28%) and MATH (+2.34%) under IMDB-heavy training.

These Qwen3-0.6B results show our joint scaling strategy effectively counters dominant domain overfitting, leading to significant recovery on minority domains, sometimes at the cost of peak head-domain performance. This yields a more balanced, generalized performance profile across diverse tasks, driven by the synergy between domain-aware reweighting and difficulty-aware scaling.

C.1 Ablation Study of Scaling Components

Combining domain- and difficulty-aware scaling delivers the strongest overall performance. To understand the individual contributions of our proposed scaling strategies, we conduct an ablation study selectively applying either the domain-aware weight ('Domain only') or the difficulty-aware weight ('Diff only'), comparing against the full version ('DISCO') and the Naive GRPO baseline. Results are reported in Table 4.

We observe varied effects from the individual components across models. For instance, on Qwen2.5-0.5B, neither difficulty-aware scaling alone (46.05%) nor domain-aware scaling alone (45.91%)

Model	Math-heavy	IMDB-heavy	NQ-heavy	ARC-heavy	Avg.
		Qwen2.5 0.5B			
Naive GRPO	46.22	46.21	46.43	45.80	46.17
Diff only	46.87	46.21	45.03	46.07	46.05
Domain only	45.94	46.33	45.28	46.10	45.91
DISCO	**47.92**	**47.67**	**47.91**	**47.70**	**47.80**
		Qwen2.5 1.5B			
Naive GRPO	64.81	64.91	65.15	65.02	64.97
Diff only	64.45	64.40	64.67	**65.39**	64.73
Domain only	**65.39**	64.73	64.65	65.18	**64.99**
DISCO	64.12	**64.93**	**65.26**	65.35	64.92
		LLaMA3.2 1B			
Naive GRPO	27.61	29.94	21.76	25.86	26.29
Diff only	28.67	29.87	**22.46**	30.52	27.88
Domain only	26.42	29.61	21.87	30.01	26.98
DISCO	**28.97**	**30.02**	22.19	**30.56**	**27.94**

Table 4: Ablation study comparing Naive GRPO with variants using only difficulty-aware scaling ('Diff only'), only domain-aware log-scaling ('Domain only', v1 weights), and both ('DISCO'). Best result among the three variants in each numerical column is bolded.

improved upon Naive GRPO (46.17%) on average. However, combining both in DISCO yields a significant boost to 47.80%. Conversely, on LLaMA3.2-1B, both 'Diff only' (27.88%) and 'Domain only' (26.98%) offer improvements over Naive GRPO (26.29%), and DISCO achieves the highest overall score (27.94%).

Interestingly, on the larger Qwen2.5-1.5B model, using 'Domain only' scaling achieves the highest average score (64.99%), slightly surpassing both Naive GRPO (64.97%) and DISCO (64.92%). This suggests domain re-weighting alone can be particularly effective for this model configuration. Despite 'Domain only' having the best average here, DISCO (64.92%) achieves competitive overall performance and secures the best scores among the variants under the IMDB-heavy and NQ-heavy conditions specifically.

These results reveal a complementary relationship between the two scaling components. Relying on one component alone proves insufficient: 'Domain only' scaling neglects sample difficulty variance within domains, while 'Diff only' scaling ignores global domain imbalance, both leading to inconsistent performance. In contrast, their joint use in DISCO consistently provides a more robust and generally higher-performing configuration across different models and domain imbalances, validating our combined approach.