LEARNING FROM MISTAKES: NEGATIVE REASONING SAMPLES ENHANCE OUT-OF-DOMAIN GENERALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

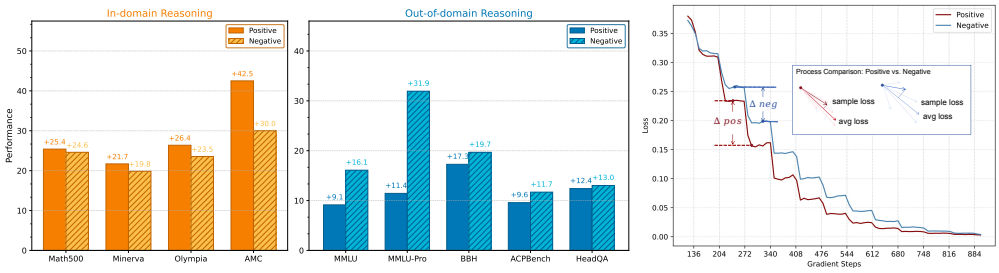
Supervised Fine-Tuning (SFT), which lays an important foundation of effective reasoning in LLMs, typically uses only correct Chain-of-Thought (CoT) data whose final answers match the ground truth, suffering from poor generalization due to overfitting and wasted data from discarding incorrect samples. Considering that incorrect samples contain implicit valid reasoning processes and diverse erroneous patterns, we investigate whether incorrect reasoning trajectories can serve as valuable supervision and surprisingly find that they substantially improve out-of-domain (OOD) generalization over correct-only training. To explain this, we performed an in-depth analysis through data, training, and inference, revealing 22 different patterns in incorrect chains, which yield two benefits: (1) *For training*, they produce a slower loss descent, indicating a broader optimization landscape that mitigates overfitting. (2) *For inference*, they raise model's policy entropy in the reasoning process by 35.67% over correct-only training (under on-policy strategy) and encourage exploration of alternative reasoning paths to improve generalization. Inspired by this, we propose **Gain-based LOss Weighting** (GLOW), an adaptive, sample-aware method that prompts models to identify underexplored patterns by rescaling sample loss weights based on inter-epoch progress. Theoretically, it converges to more generalizable solutions. Empirically, it outperforms full-data training across different model sizes and significantly improves the OOD performance of Qwen2.5-7B trained on math reasoning by 15.81% over positive-only training. Code is available at Github.

1 INTRODUCTION

Recent advances in large language models (LLMs), exemplified by GPT-5 (OpenAI, 2025), Gemini (Comanici et al., 2025), DeepSeek-R1 (DeepSeek-AI et al., 2025), and Qwen (Yang et al., 2025a), highlight the central role of Supervised Fine-Tuning (SFT) in modern training pipelines. By adapting base models with curated task-specific data, often enriched with Chain-of-Thougnt(CoT) annotations, SFT establishes the foundation for effective reasoning. Together with reinforcement learning (RL), which further optimizes outputs via preference-based feedback, SFT constitutes the standard two-stage paradigm underlying today's state-of-the-art LLMs.

Although SFT forms the foundation of current training pipelines, existing methods remain hindered by limitations that reduce both effectiveness and efficiency, most notably two key shortcomings (Luo et al., 2024a; Chu et al., 2025; Gupta et al., 2025; Deb et al., 2025): (1) **Poor Generalization**: models tend to overfit to domain-specific reasoning shortcuts present in the demonstrations rather than learning robust, transferable reasoning capabilities (Press et al., 2022; Han et al., 2025). This often leads to performance degradation on out-of-distribution (OOD) tasks(see Tables 1 and 2 for details). (2) **Data Inefficiency**: Dominant SFT relies on distilled reasoning paths and uses rejection sampling to select those leading to correct answers and formats. Discarding samples yielding incorrect answers not only wastes resources but also overlooks the correct reasoning paths potentially hidden therein (Hamdan & Yuret, 2025; Luo et al., 2024b; Li et al., 2025).

Owing to the above challenges, a natural question arises: *Are incorrect reasoning trajectories, often dismissed as noise, truly incapable of providing effective supervision?* Considering that errors encompass both valid reasoning processes and diverse erroneous patterns, these characteristics prompt



(a) (b)

Figure 1: (a) Qwen2.5-14B trained with positive samples shows limited transfer beyond math, whereas models trained on negative samples generalize more broadly across reasoning tasks. (b) Training loss curve on MMLU with Qwen2.5-32B.

us to inquire whether an SFT approach can not only improve data efficiency through utilizing all available samples but also benefit from this expanded exploration space to enhance generalization.

To investigate, we distill some data of mathematical reasoning problems and their corresponding trajectories from Qwen3-8B (Yang et al., 2025b) and split them based on whether the model's final answer matches the ground truth: correct solutions (positive) versus incorrect ones (negative). We then fine-tune a series of models, including Qwen2.5 and LLaMA (Dubey et al., 2024), on each subset separately. As illustrated in Figure 1a, the results surprisingly demonstrate that **models trained solely on negative samples outperform those trained on positive samples across many tasks, especially in the out-of-domain benchmarks.**.

To understand this, we analyze it stepwise from the perspectives of data, training, and inference. The negative samples can be divided into 9 major types and 22 diverse patterns (see Table 3 and Appendix A.3), with each type serving as a distinct environment. To perform well across these environments, the model needs to learn invariant reasoning patterns, fostering better generalization. Such diversity also brings two benefits: 1) For training: loss declines more slowly than positive-only training yet converges eventually (Figure 1b), demonstrating the optimizing process for diverse reasoning patterns instead of overfitting to limited patterns. 2) For inference, models trained on negatives exhibit higher policy entropy in reasoning trajectories, indicating more diverse path exploration and boosting cross-domain generalization. Collectively, the surprising advantage of negatives over positives reveals that previously overlooked negatives can encourage the model to conduct broader, more diverse exploration during optimization, yielding more efficient, generalizable reasoning strategies.

These observations provide insights into training more generalizable models with SFT. While relying solely on negative samples is a possible strategy, it remains a form of rejection sampling with low data efficiency. To address this, we introduce a dynamic mechanism called **G**ain-based **LO**ss **W**eighting (GLOW), which leverages the entire dataset without requiring prior negative sample selection. Specifically, during training, we measure a sample's value by its loss difference across consecutive epochs: a smaller difference implies minimal loss change between two optimization steps, indicating insufficient coverage of the sample's direction by other samples' optimization, and thus highlighting its greater uniqueness relative to other samples. We design a scaling function that adaptively emphasizes such samples by increasing their contribution to the loss. Theoretically, we show that this mechanism guides the model toward solutions with stronger generalization. Experimental results across models with different scales demonstrate its consistent improvements. In particular, on Qwen2.5-7B, GLOW achieves an average improvement of 2.14% over mixed-data training and gains of 5.51% in OOD scenarios compared to training with only positive samples.

In all, our core contributions can be summarized as follows:

- We provide the first systematic study demonstrating that negative reasoning samples constitute valuable supervision: fine-tuning on them improves out-of-distribution generalization. This offers a novel perspective on mitigating overfitting in SFT by exploiting these data.
- We provide a deep analysis of how negatives improve generalization from data, training, and inference perspectives, which reflects that negative samples can enable the model to conduct broader exploration of reasoning paths and directly strengthen generalization.

- We propose a novel GLOW mechanism that adaptively recognize and amplifies the contribution of samples with the highest training gain, measured by their loss reduction trajectory. This approach improves the utility of negative samples, enhances generalization, and offers a practical path toward more data-efficient SFT.

2 Related Works

Supervised Fine-Tuning for Reasoning SFT has emerged as a central approach for improving the reasoning capabilities of large language models (Wei et al., 2021; Ouyang et al., 2022). It adapts a general-purpose model to downstream tasks or desired behaviors by training on carefully curated datasets. To ensure data quality, rejection sampling (Ahn et al., 2024) is often employed as a filtering strategy that discards samples failing to meet predefined standards. Recent studies further show that SFT can transfer long CoT reasoning patterns from larger models to smaller ones (Shao et al., 2024a; Zheng et al., 2025; Yu et al., 2025b), thereby enhancing the reasoning performance of resource-efficient models. In addition, SFT provides a strong initialization for reinforcement learning by aligning models with human-preferred behaviors before optimization (Lewkowycz et al., 2022; Shao et al., 2024b). However, this reliance on heavily filtered data inevitably wastes data, as a large portion of available supervision is discarded.

Learning from Negative Data Learning from negative samples can be broadly categorized into prompt-based and finetuning-based approaches. Prompt-based methods exploit negative samples to guide model behavior. Gao & Das (2024) uses them to represent ambiguous preferences that models should avoid, while Alazraki et al. (2025) shows that incorporating a negative sample in the prompt is more effective than adding an extra positive one, and that providing incorrect rationales may even overconstrain the model. However, the effectiveness of such methods is limited by the reasoning and instruction following ability of the model. In contrast, finetuning-based approaches are more commonly employed to enhance reasoning ability or to provide a strong initialization for reinforcement learning (Guo et al., 2025). Some studies leverage teacher models to generate positive CoT trajectories from negative samples (Yu et al., 2025a; Pan et al., 2025; An et al., 2023), while others introduce prefixes to distinguish between positive and negative samples (Wang et al., 2024a; Tong et al., 2024).Nevertheless, most works treat negative samples as less valuable than positive ones. Although Li et al. (2025) emphasize the global reasoning structure over the final answer, they do not explicitly recognize negative samples as an independent supervision source.

Domain Generalization in LLMs Most fine-tuning studies prioritize improving reasoning within a single domain such as mathematics or code, while systematic treatment of cross-domain transfer remains limited. For example, Huan et al. (2025) study math data and show that SFT induces significant latent space and token rank shifts, which lead to forgetting of general capabilities. Wu et al. (2025) introduce two metrics, knowledge index and information gain, to disentangle knowledge from reasoning, finding that SFT on math provides little benefit in knowledge-intensive domains such as medicine. Similarly, Yang et al. (2025c) and Zhao et al. (2025) argue that SFT often constructs only superficial reasoning chains and fails to transfer effectively across domains. However, these studies are primarily diagnostic analyses: they do not propose concrete methods, nor do they investigate the problem from a data-centric perspective.

3 Negative Samples Enhance Out-of-Domain Reasoning

In this section, we describe the empirical phenomenon that motivates our study: fine-tuning on negative reasoning samples can enhance OOD generalization more effectively than fine-tuning on positive samples. We first detail the controlled experiments designed to validate this phenomenon and then present results that demonstrate its consistency across diverse benchmarks and model scales.

3.1 Data Construction and Training Setup

We use Qwen3-8B to distill responses from OpenMathReasoning (Moshkov et al., 2025) and the MMLU (Hendrycks et al., 2021b) training set as training data for mathematical and common sense tasks. Responses that matched the final answer are classified as positive, while others are defined

as negative. To ensure a fair comparison, we sample an equal number of positive and negative responses, each containing the complete reasoning format. We then use Qwen-2.5 series(3B, 7B, 14B, and 32B) model and Llama-3.1 8B for SFT training. For more detailed training configurations, please refer to the Appendix 3.1.

3.2 NEGATIVES SURPASS POSITIVES IN OUT-OF-DOMAIN

Table 1: Cross-domain performance of models trained on the **math reasoning** dataset. "Avg." denotes the average score within each group.

		Math Reasoning (In-Domain)					Common Sense (Out-of-Domain)				Other Reasoning (Out-of-Domain)		
Model	Setting	Math500	Minerva	Olympia	AMC	Avg.	MMLU	MMLU-Pro	BBH	Avg.	ACPBench	HeadQA	Avg.
Qwen2.5-3B	Base	52.60	21.32	22.52	32.50	32.24	31.88	12.54	27.75	24.06	23.31	33.15	28.23
	Full	60.80	26.10	23.26	35.00	36.29	64.13	38.66	52.29	51.69	32.68	62.69	47.69
	Positive	61.60	25.74	24.44	42.50	38.60	54.45	25.62	44.35	41.50	30.21	59.81	45.01
	Negative	58.60	23.53	24.15	42.50	37.20	64.09	39.20	53.87	52.39	33.06	63.13	48.10
	Δ(pos-neg)	+3.00	+2.21	+0.29	0.00	+1.38	-9.64	-13.58	-9.52	-10.91	-2.85	-3.32	-3.09
Qwen2.5-7B	Base	58.40	26.84	26.07	52.50	40.95	55.80	26.56	51.10	44.49	28.77	57.29	43.03
	Full	76.60	40.07	38.96	55.00	52.66	72.24	53.71	70.84	65.60	38.27	72.06	55.17
	Positive	78.00	36.76	41.78	57.50	53.51	61.03	32.70	60.58	51.44	33.38	68.60	50.99
	Negative	77.60	40.44	38.37	57.50	53.48	73.11	53.74	71.73	66.19	38.98	71.81	55.40
	Δ(pos-neg)	+0.40	-3.68	+3.41	0.00	+0.03	-12.08	-21.04	-11.15	-14.76	-5.60	-3.21	-4.41
Qwen2.5-14B	Base	62.60	26.84	27.56	40.00	39.25	64.68	35.77	59.27	53.24	37.04	68.75	52.90
	Full	86.80	47.79	52.30	82.50	67.35	81.56	67.63	80.90	76.70	48.13	81.44	64.79
	Positive	88.00	48.53	53.93	82.50	68.24	73.81	47.21	76.54	65.85	46.62	81.15	63.89
	Negative	87.20	46.69	51.11	70.00	63.75	80.77	67.70	78.95	75.81	48.73	81.77	65.25
	Δ(pos-neg)	+0.80	+1.84	+2.82	+12.50	+4.49	-6.96	-20.49	-2.41	-9.95	-2.11	-0.62	-1.37
Qwen2.5-32B	Base	63.20	34.19	26.52	35.00	39.73	68.34	39.80	58.65	55.60	38.63	68.45	53.54
	Full	92.20	52.57	57.19	85.00	71.74	85.22	73.10	83.53	80.62	50.67	84.90	67.79
	Positive	91.40	50.74	60.89	85.00	72.01	79.01	54.31	80.61	71.31	49.96	83.15	66.56
	Negative	92.20	50.74	58.37	95.00	74.08	85.47	73.53	84.51	81.17	51.80	85.27	68.54
	Δ(pos-neg)	-0.80	0.00	+2.52	-10.00	-2.07	-6.46	-19.22	-3.90	-9.86	-1.84	-2.12	-1.98
Llama3.1-8B	Base	2.80	1.10	0.44	0.00	1.09	66.49	0.47	2.33	23.10	5.18	2.30	3.74
	Full	41.20	18.01	14.67	15.00	22.22	62.48	36.88	55.12	51.49	32.96	65.90	49.43
	Positive	37.80	18.01	10.37	12.50	19.67	41.95	23.15	45.07	36.72	31.20	47.81	39.50
	Negative	34.40	18.38	9.19	20.00	20.49	62.14	36.22	54.85	51.07	33.31	65.17	49.24
	Δ(pos-neg)	+3.40	-0.37	+1.18	-7.50	-0.82	-20.19	-13.07	-9.78	-14.35	-2.11	-17.36	-9.74

Table 2: Cross-domain performance of models trained on the **common sense** dataset. "Avg." denotes the average score within each group.

		Math Reasoning (Out-of-Domain)					Common Sense (In-Domain)				Other Reasoning (Out-of-Domain)		
Model	Setting	Math500	Minerva	Olympia	AMC	Avg.	MMLU	MMLU-Pro	BBH	Avg.	ACPBench	HeadQA	Avg.
Qwen2.5-3B	Base	52.60	21.32	22.52	32.50	32.24	31.88	12.54	27.75	24.06	23.31	33.15	28.23
	Full	58.20	23.16	25.19	35.00	35.39	66.74	40.82	53.35	53.64	35.70	67.61	51.66
	Positive	59.20	27.21	25.04	30.00	35.36	67.88	42.56	52.84	54.43	34.93	67.69	51.31
	Negative	59.60	28.31	25.48	40.00	38.35	65.42	38.55	52.28	52.08	36.13	68.85	52.49
	Δ(pos-neg)	-0.40	-1.10	-0.44	-10.00	-2.99	+2.32	+4.01	+0.56	+2.30	-1.20	-1.16	-1.18
Qwen2.5-7B	Base	58.40	26.84	26.07	52.50	40.95	55.80	26.56	51.10	44.49	28.77	57.29	43.03
	Full	75.60	38.60	40.15	47.50	50.46	73.14	51.15	71.30	65.20	42.18	72.76	57.47
	Positive	74.40	37.50	39.85	50.00	50.44	73.42	53.22	68.23	64.96	40.32	74.25	57.29
	Negative	77.00	37.13	42.07	60.00	54.05	71.23	45.79	69.46	62.16	42.61	73.38	58.00
	Δ(pos-neg)	-2.60	+0.37	-2.22	-10.00	-3.61	+2.19	+7.43	-1.23	+2.80	-2.29	+0.87	-0.71
Qwen2.5-14B	Base	62.60	26.84	27.56	40.00	39.25	64.68	35.77	59.27	53.24	37.04	68.75	52.90
	Full	82.20	43.01	51.85	70.00	61.77	78.13	59.57	80.56	72.75	48.87	79.94	64.41
	Positive	80.20	42.28	50.96	72.50	61.49	80.09	65.26	80.21	75.19	48.56	80.53	64.55
	Negative	83.00	45.22	48.89	65.00	60.53	76.83	56.03	80.15	71.00	48.27	80.56	64.42
	Δ(pos-neg)	-2.80	-2.94	+2.07	+7.50	+0.96	+3.26	+9.23	+0.06	+4.18	+0.29	-0.03	+0.13
Qwen2.5-32B	Base	63.20	34.19	26.52	35.00	39.73	68.34	39.80	58.65	55.60	38.63	68.45	53.54
	Full	86.60	46.69	55.70	80.00	67.25	79.06	61.15	79.94	73.38	49.89	83.01	66.45
	Positive	85.20	46.69	56.15	75.00	65.76	81.97	68.54	81.60	77.37	50.35	82.90	66.63
	Negative	86.40	47.06	56.89	72.50	65.71	77.99	58.34	80.71	72.35	51.20	82.39	66.80
	Δ(pos-neg)	-1.20	-0.37	-0.74	+2.50	+0.05	+3.98	+10.20	+0.89	+5.02	-0.85	+0.51	-0.17
Llama3.1-8B	Base	2.80	1.10	0.44	0.00	1.09	66.49	0.47	2.33	23.10	5.18	2.30	3.74
	Full	20.00	15.81	6.52	2.50	11.21	66.49	40.56	53.73	53.59	36.06	69.55	52.81
	Positive	15.60	11.76	3.85	7.50	9.68	64.73	39.74	45.39	49.95	29.61	67.69	48.65
	Negative	23.00	16.18	6.67	10.00	13.96	64.63	38.85	53.23	52.24	37.15	69.80	53.48
	Δ(pos-neg)	-7.40	-4.42	-2.82	-2.50	-4.29	+0.10	+0.89	-7.84	-2.28	-7.54	-2.11	-4.83

As shown in Table 1 and Table 2, we surprisingly find that training on negative samples, although it yields smaller improvements than positive samples on in-domain performance, consistently consistently improves OOD generalization. Overall, models trained on negative math reasoning samples achieve an average improvement of 11.97% on common sense tasks and 4.11% on other reasoning tasks. Similarly, models trained on negative MMLU samples gain an average of 1.98% on mathematical reasoning and 1.35% on other reasoning benchmarks. Although mathematical problems

are generally more suitable for constructing reasoning-focused data, the same trend is observed for models trained on MMLU, indicating that the benefit of negative samples for OOD generalization is not limited to a specific domain. These observations motivate a deeper analysis into the underlying factors that make negative samples more effective for enhancing OOD reasoning performance.

4 WHY NEGATIVE IS BETTER

To better understand why negative samples benefit out-of-distribution generalization, we analyze this phenomenon step by step. We first examine the properties of the data to assess how negatives introduce greater diversity. We then study the training dynamics to reveal how this diversity influences optimization. Finally, we analyze model behaviors during inference to show how these training effects translate into stronger generalization. This step-by-step analysis sheds light on the mechanism through which negatives improve OOD performance.

4.1 DATA PERSPECTIVE

Following (He et al., 2025), we categorize reasoning errors into 9 major types and 22 subtypes. For each negative sample in the OpenMathReasoning and MMLU training datasets, we employ Gemini-2.5-Pro (Comanici et al., 2025) to assign its error category (see Appendix A.3 for the prompt used). As shown in Table 3, the distribution of error types is highly diverse, covering a wide spectrum from logical errors to comprehension errors. Negative samples exhibit a richer variety of reasoning patterns, whereas positive data tend to follow more consistent trajectories. Detailed classification can be found in Appendix A.2.

Table 3: Error categorization in the negative OpenMathReasoning and MMLU samples.

Error Categories	**OpenMathReasoning**	**MMLU**
Calculation	27	9
Completeness	11	28
Evaluation System	2599	2024
Formal	57	123
Knowledge	27	199
Logical	195	4116
Programming	8	5
Understanding	435	1056
Special Cases	301	1137
Total	**3660**	**8697**

This diversity can be theoretically understood through the lens of Invariant Risk Minimization(IRM) (Arjovsky et al., 2019). IRM posits that generalization can be improved when a model learns representations that capture invariant causal features across diverse environments. In our setting, each category of negative reasoning errors can be viewed as an environment with its own patterns. By being exposed to a broad variety of such environments, the model is implicitly encouraged to discover reasoning strategies that are invariant across them. Formally, let $\mathcal{E}$ denote the set of environments induced by distinct negative error categories, and let each $e \in \mathcal{E}$ correspond to a data distribution D^e over input–output sequences (x, y). We decompose the language model into a shared sequence representation Φ and a shared next-token predictor w. The IRM objective in the autoregressive setting requires a single predictor w to be simultaneously optimal when composed with Φ in every environment:

$$\min_{\Phi} \sum_{e \in \mathcal{E}} R^e(w \circ \Phi) \quad \text{subject to} \quad w \in \arg\min_{w'} R^e(w' \circ \Phi),\ \forall e \in \mathcal{E}, \tag{1}$$

where the per-environment autoregressive risk is given by

$$R^e(w \circ \Phi) = \mathbb{E}_{(x,y)\sim D^e}\left[\sum_{t=1}^{|y|} \ell\Big(w\big(\Phi(x, y_{<t})\big),\ y_t\Big)\right], \tag{2}$$

and ℓ denotes the cross-entropy loss. By enforcing that the same next-token classifier is optimal across all environments when paired with Φ, **IRM encourages the representation to encode invariant causal structure so that the conditional next-token distributions, and consequently the reasoning strategies, remain stable across $\mathcal{E}$.**

From this perspective, positive samples are clean and correct but lack diversity, which limits their ability to support invariance. Negative samples, in contrast, span multiple environments and expose diverse failure modes within valid reasoning structures. This diversity drives the model to learn more robust representations that generalize across heterogeneous reasoning scenarios.

4.2 TRAINING PERSPECTIVE

To characterize the learning dynamics, we plot the training loss every 10 steps for all models fine-tuned on positive and negative samples from math reasoning and MMLU. We present Qwen2.5-32B (Figure 1b) as a representative example, while others are provided in Appendix A.7. The curves follow a consistent stage-wise pattern. Loss drops sharply at the end of each epoch for positive samples, leading to faster initial convergence, whereas negative samples produce a smoother, gradual decline that ultimately reaches a comparable loss floor. This is because the optimization directions of individual samples align more consistently with the average gradient in the positive set than in the negative set, while negative samples point to a wider exploration space. We quantify this behavior using the average loss difference between consecutive epochs, as reported in Table 4, which confirms that positives decrease faster in the early stages. The negative curve remains monotonic and closely tracks the positive curve, indicating that negatives provide meaningful learning signals rather than noise. They encode diverse exploratory patterns where incorrect answers coexist with partially valid reasoning, offering sustained constraints that encourage the model to develop more robust reasoning strategies instead of memorizing a single correct trajectory.

Table 4: Comparison of training dynamics of Qwen2.5-32B under positive and negative MMLU settings. Each value represents the **difference** between the per-epoch loss drops of the Positive (Δ_{pos}) and Negative (Δ_{neg}), i.e., $\Delta_{\text{pos}} - \Delta_{\text{neg}}$. Small decimal values are expected, and the interpretation relies on the relative difference.

Model	$\Delta^{\text{epoch } 2-1}_{\text{avg\_loss}}$	$\Delta^{\text{epoch } 3-2}_{\text{avg\_loss}}$	$\Delta^{\text{epoch } 4-3}_{\text{avg\_loss}}$	$\Delta^{\text{epoch } 5-4}_{\text{avg\_loss}}$
Qwen2.5-3B	0.014957	0.013486	0.015686	0.014000
Qwen2.5-7B	0.009729	0.022514	0.014172	0.001156
Qwen2.5-14B	0.008515	0.017786	0.011157	0.005472
Qwen2.5-32B	0.007143	0.018200	0.015557	0.003772
Llama3.1-8B	0.015586	0.023344	0.005571	0.004915

These results show that the value of negative samples lies in their diversity. Although this slows loss reduction by introducing varied optimization directions, it compels the model to explore a broader reasoning space and converge to more generalizable patterns.

4.3 Inference Perspective

After analyzing the properties of the training data and the characteristics of the optimization process, we further investigate what drives the superior out-of-distribution performance of models trained with negatives. To this end, we focus on policy entropy, which provides a principled measure of the uncertainty and exploration in model reasoning. We investigate how training on different types of trajectories shapes the entropy dynamics of model reasoning. We first analyze the policy entropy of the model. We use M_{pos} to denote the model trained on the positive subset of OpenMathReasoning, and M_{neg} for the one trained on the negative subset. To assess entropy in both in-domain and out-of-domain settings, we distill trajectories with reasoning trace and final answers from Qwen3-8B on a math set (denoted as "Math") and an OOD set (denoted as "Other").

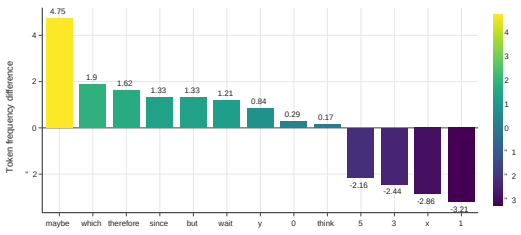


Figure 2: Token frequency differences between M_{neg} and M_{pos} on digits and high-entropy tokens.

Table 5: Policy entropy analysis on M_{pos} and M_{neg}.

Model	Setting	Data	$\bar{H}_{\text{think}}$	$\bar{H}_{\text{ans}}$	ΔH
M_{pos}	Off-policy	Math	0.909	0.708	0.202
		Other	1.138	0.873	0.265
	On-policy	Math	0.753	0.601	0.153
		Other	0.669	0.757	-0.088
M_{neg}	Off-policy	Math	1.212	0.883	0.329
		Other	1.427	0.992	0.435
	On-policy	Math	1.011	0.772	0.239
		Other	0.917	0.783	0.134

We mark the thinking span as the tokens between <think> and </think>, and the answer span as the tokens after </think>. For each prompt x, we compute the policy entropy within these spans under two rules: (i) **off-policy**: measuring under the teacher's reference trajectory; and (ii) **on-policy**: the model generates its own trajectory under a fixed decoding rule. Unless noted, entropy is computed from raw T=1 logits (no temperature rescaling), not excluding padding and special boundary tokens.

Formally, let $\mathcal{V}$ be the vocabulary and θ the model parameters. At step t the token-level policy entropy is

$$H_t(\theta \mid x, y_{<t}) \;=\; -\sum_{v\in\mathcal{V}} p_\theta(v \mid x, y_{<t}) \log p_\theta(v \mid x, y_{<t}) \,, \tag{3}$$

where $p_\theta(\cdot \mid x, y_{<t})$ is induced by pre-softmax logits. For each sample i, let $\mathcal{T}^{(i)}_{\text{think}}$ and $\mathcal{T}^{(i)}_{\text{ans}}$ denote the token within thinking and answer spans, respectively. The spans are determined by model's own generation (on-policy) or teacher's trajectory (off-policy). We report the average entropy per span:

$$\bar{H}^{(i)}_{\text{think}} \;=\; \frac{1}{\left|\mathcal{T}^{(i)}_{\text{think}}\right|} \sum_{t\in\mathcal{T}^{(i)}_{\text{think}}} H_t, \qquad \bar{H}^{(i)}_{\text{ans}} \;=\; \frac{1}{\left|\mathcal{T}^{(i)}_{\text{ans}}\right|} \sum_{t\in\mathcal{T}^{(i)}_{\text{ans}}} H_t, \tag{4}$$

as well as the entropy drop across the boundary:

$$\Delta H^{(i)} \;=\; \bar{H}^{(i)}_{\text{think}} - \bar{H}^{(i)}_{\text{ans}}, \tag{5}$$

Results in Table 5 show that models trained on negative trajectories sustain higher policy entropy in thinking span and exhibit a larger boundary gap, **indicating broader search followed by sharper commitment and aligning with their stronger cross-domain transfer.** Moreover, Off-policy are consistently higher than on-policy, since teacher-forcing trajectories push the model into low-confidence neighborhoods with diffuse distributions, while self-decoding remains confined to a few high-confidence modes that yield lower entropy. Under distribution shift, entropy increases for both spans. The positive-trained model degrades most and even flips the margin on on-policy OOD, indicating unstable calibration and over-specialization to in-domain templates. Overall, negative supervision induces a "high-entropy reasoning" profile that better predicts generalization.

We then analyze the distribution of high-entropy tokens in the trajectories generated by different models. Figure 2 shows the token frequency distribution difference per trajectory for M_{pos} and M_{neg}. Compared with M_{pos}, M_{neg} produces substantially more discourse and hesitation tokens such as "maybe," "wait," and "but," while emitting numerals less frequently, indicating that its trajectories devote more budget to exploratory connective reasoning than committing to numeric content. We also visualize the trajectories generated by the two models in Figure 6, showing that during inference, M_{neg} has a higher effective branching factor, enabling the model to maintain multiple continuations plausible and to explore more reasoning paths before committing to an answer.

5 BETTER LEVERAGING OF NEGATIVE

In this section, we move beyond the empirical finding that negatives improve out-of-distribution generalization. Relying solely on negatives is essentially a form of rejection sampling and does not make efficient use of the data, as sample quality cannot be determined simply by correctness. Our goal is to develop models that achieve strong performance on both in-domain and out-of-distribution settings with higher data efficiency. To this end, we focus on the training process as the most principled direction for improvement. Building on the analysis of training dynamics, we present a general mechanism, establish its theoretical foundation, and validate its effectiveness through experiments.

5.1 GAIN-BASED LOSS WEIGHTING

Negative trajectory supervision improves reasoning because it enlarges the model's effective training space. This is evidenced by three key observations: (1) compared to positive training, negative training yields similarly shaped learning curves but slower convergence at a fixed step budget, indicating that updates are less concentrated along a few dominant directions and thus avoid early collapse into limited reasoning patterns. (2) Analysis in Section 4.3 shows that models trained with negatives exhibit a higher policy entropy, thereby gaining a greater capacity for exploration. Taken together, our observations suggest a practical motivation: reweight the objective to amplify the loss contributions of under-explored samples, dynamically steering updates toward complementary directions and yielding progressively larger incremental gains.

We assess the learning progress of each sample by the reduction in its loss across consecutive epochs. Samples with small loss reductions correspond to patterns that remain insufficiently learned and offer

higher optimization value, while large reductions imply saturated learning with limited marginal utility. We therefore use $\Delta_i^{(t)} = \ell_i^{(t)} - \ell_i^{(t-1)}$ to identify under-learned samples and amplify their impact, ensuring that training prioritizes the regions where the model can still achieve the greatest gain. Specifically, the contribution of each sample is adjusted according to $\Delta_i^{(t)}$:

$$w_i^{(t)} = \alpha \cdot \sigma\Big(\beta \cdot \Delta_i^{(t)}\Big), \tag{6}$$

where $\sigma(\cdot)$ is the sigmoid function, and α, β are scaling hyperparameters. For the first epoch, we set $w_i^{(1)} = 1$ for all samples. The reweighted training objective becomes:

$$\mathcal{L}_{\text{GLOW}(\theta)} = \sum_{i=1}^{N} w_i^{(t)} \cdot \ell_i^{(t)}. \tag{7}$$

Theoretical View We provide a sketch of why the reweighted objective in Eq. 7 improves generalization. Consider one gradient update at step t, $\theta^{(t+1)} = \theta^{(t)} - \eta G^{(t)}$ with $G^{(t)} = \sum_i w_i^{(t)} g_i^{(t)}$. By the L-smoothness of the loss, a Taylor expansion gives

$$\Delta_i^{(t)} \;=\; \ell_i^{(t)} - \ell_i^{(t+1)} \;\approx\; \eta\, g_i^{(t)\top} G^{(t)} - \tfrac{1}{2}\eta^2 G^{(t)\top} H_i G^{(t)}, \tag{8}$$

where H_i is the local Hessian of model parameters. The leading term shows that $\Delta_i^{(t)}$ is large if $g_i^{(t)}$ aligns with the dominant descent directions $G^{(t)}$, and small otherwise. Hence, Eq. 6 adaptively increases the weight of samples whose gradients lie in less explored directions.

Let $F_w = \frac{1}{N}\sum_i w_i^{(t)} g_i^{(t)} g_i^{(t)\top}$ denote the empirical Fisher, which quantifies the extent of the model's directional exploration in parameter space. Increasing $w_i^{(t)}$ for small-$\Delta_i^{(t)}$ samples adds positive semi-definite increments $\delta w_i g_i g_i^\top$ along diverse directions. By Weyl's inequality (Weyl, 1912), this raises the smaller eigenvalues of F_w, improving its effective rank and conditioning (Horn & Johnson, 2012). Since F_w approximates the Hessian in standard settings (Martens, 2020), the optimization landscape becomes better conditioned, leading to more balanced descent across directions. Stability-based generalization bounds (Bousquet & Elisseeff, 2002; Hardt et al., 2016) then imply tighter generalization error, as flatter and more isotropic minima correlate with improved robustness (Keskar et al., 2016; Neyshabur et al., 2017).

In summary, the dynamic weighting in Eq. 6 systematically enlarge gradients from diverse, less-explored reasoning trajectories (often negatives), increases gradient diversity, and thus improves both optimization and generalization. For detailed proof, see appendix A.5

5.2 EXPERIMENTAL RESULTS

Building on the theoretical analysis, we empirically validate the effectiveness of GLOW in the SFT stage. All other experimental settings are the same as 3.1 and details are described in Appendix A.1.

Table 6: Cross-domain performance of models trained on the **math reasoning** dataset. "Avg." denotes the average score within each group.

		Math Reasoning (In-Domain)					Common Sense (Out-of-Domain)				Other Reasoning (Out-of-Domain)		
Model	Setting	Math500	Minerva	Olympia	AMC	Avg.	MMLU	MMLU-Pro	BBH	Avg.	ACPBench	HeadQA	Avg.
Qwen2.5-3B	Full	60.80	26.10	23.26	35.00	36.29	64.13	**38.66**	52.29	51.69	32.68	62.69	47.69
	GLOW	**62.80**	**27.21**	**24.30**	**42.50**	**39.20**	**64.49**	38.63	**53.20**	**52.11**	**33.66**	**63.38**	**48.52**
Qwen2.5-7B	Full	76.60	40.07	38.96	55.00	52.66	72.24	53.71	70.84	65.60	38.27	72.06	55.17
	GLOW	**79.60**	**40.07**	**41.04**	**60.00**	**55.18**	**73.99**	**55.77**	**71.99**	**67.25**	**39.19**	**72.50**	**55.85**
Qwen2.5-14B	Full	86.80	47.79	52.30	82.50	67.35	81.56	67.63	80.90	76.70	48.13	81.44	64.79
	GLOW	**87.80**	**52.21**	**52.44**	**82.50**	**68.74**	**82.53**	**68.70**	**81.65**	**77.63**	**49.51**	**82.35**	**65.93**

GLOW enhances cross-domain generalization without pre-selecting samples. We apply GLOW to the random shuffled mixture of positive and negative data and observe consistent improvements across domains and different scales of models. For simplicity, we only report results for

Table 7: Cross-domain performance of models trained on the **common sense** dataset. "Avg." denotes the average score within each group.

		Math Reasoning (Out-of-Domain)					Common Sense (In-Domain)				Other Reasoning (Out-of-Domain)		
Model	Setting	Math500	Minerva	Olympia	AMC	Avg.	MMLU	MMLU-Pro	BBH	Avg.	ACPBench	HeadQA	Avg.
Qwen2.5-3B	Full	58.20	23.16	25.19	35.00	35.39	66.74	40.82	**53.35**	53.64	35.70	67.61	51.66
	GLOW	**61.40**	**29.41**	**25.78**	**40.00**	**39.15**	**67.09**	**41.27**	52.61	**53.66**	**36.20**	**69.15**	**52.68**
Qwen2.5-7B	Full	75.60	38.60	40.15	47.50	50.46	73.14	**51.15**	71.30	65.20	42.18	72.76	57.47
	GLOW	**78.20**	**41.18**	**43.70**	**60.00**	**55.77**	**74.51**	51.13	**71.99**	**65.88**	**43.56**	**75.35**	**59.46**
Qwen2.5-14B	Full	82.20	43.01	51.85	70.00	61.77	78.13	59.57	80.56	72.75	48.87	79.94	64.41
	GLOW	**85.00**	**48.09**	**54.22**	**70.00**	**64.33**	**79.97**	**62.78**	**82.32**	**75.02**	**50.95**	**82.20**	**66.58**

full and GLOW. For training results using standard SFT on positive-only and negative-only samples, please refer to Table 1 and Table 2. As shown in Table 6, GLOW surpasses standard SFT in-domain across all math-trained models and attains the best average on out-of-domain tasks. On Qwen2.5-7B it reaches 55.18 in-domain and 67.25 out-of-domain, while remaining competitive on commonsense. Commonsense-trained models also show clear overall gains. Table 7 further reports that on Qwen2.5-14B, GLOW lifts out-of-domain math from 61.77 to 64.33 and out-of-domain reasoning from 64.41 to 66.58. These results indicate stronger data use from leveraging all samples and consistent improvements in both settings.

GLOW typically assigns higher weights to negatives. As shown in Figure 3, we train Qwen2.5-3B on math and MMLU tasks using GLOW. We use only questions and direct answers (for correctness checking) from these datasets, with all responses distilled from Qwen3-8B. The figure shows the fraction of negatives among examples receiving larger weights at each epoch. During Math and MMLU training, this fraction stays above 50% for most epochs, reaches about 75% to 80% early in training, and then decreases as learning progresses, but stays near 50%. This occurs because GLOW assigns larger weights to examples with stagnant loss reduction, a condition more common among negative samples. As a result, training places greater emphasis on unresolved reasoning rather than easy positives.

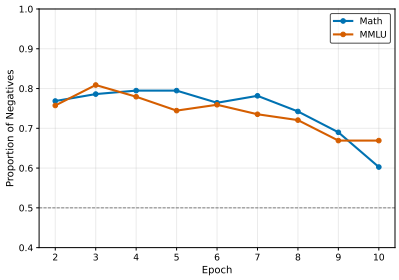


Figure 3: Fraction of negatives in the subset with the larger weights over epochs on Math and MMLU training.

GLOW enhances reasoning exploration while maintaining answer decisiveness. As shown in Table 8, applying GLOW consistently increases the average entropy during the thinking phase across all settings. For instance, think entropy rises from 0.36 to 0.71 on Math-to-Math and from 0.96 to 1.44 on MMLU-to-Other. In contrast, answer entropy changes modestly and even decreases out-of-domain. Taken together, these effects show that GLOW promotes broader exploration in reasoning while preserving answer decisiveness, which benefits generalization.

Table 8: Policy entropy changes with and without GLOW under various settings.

Setting	**Train**	**Test**	$\bar{H}_{\text{think}}$	$\bar{H}_{\text{ans}}$	ΔH
Full	Math	Math	0.36	0.22	0.14
		Other	1.24	1.38	-0.14
	MMLU	Math	0.54	0.34	0.20
		Other	0.96	0.98	-0.02
GLOW	Math	Math	0.71	0.35	0.36
		Other	1.52	1.30	0.22
	MMLU	Math	0.89	0.52	0.37
		Other	1.44	1.21	0.23

6 Conclusion

We show that negative reasoning trajectories can improve SFT generalization, mitigating the out-of-domain weakness of conventional training. Through analyses of data, training, and inference, we explain why negatives improve OOD generalization. Building on these insights, we introduce Gain-based LOss Weighting (GLOW), an adaptive, sample-aware scheme that up-weights underexplored examples by rescaling losses according to inter-epoch progress. Experiments demonstrate more data-efficient learning and consistent generalization gains across models and tasks.

ETHICS STATEMENT

This work does not involve human subjects, sensitive personal data, or potentially harmful applications. The datasets used in our experiments are derived from publicly available resources and follow their respective licenses. We do not foresee ethical risks or violations associated with our methodology or findings.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of our model selection, training objectives, and experimental setups in the main paper. Hyperparameters, dataset composition, and additional implementation details are included in the appendix. To further facilitate reproducibility, we will release our code through the URL referenced in the abstract.

REFERENCES

Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. Large language models for mathematical reasoning: Progresses and challenges. arXiv preprint arXiv:2402.00157, 2024.

Lisa Alazraki, Maximilian Mozes, Jon Ander Campos, Tan Yi-Chern, Marek Rei, and Max Bartolo. No need for explanations: Llms can implicitly learn from mistakes in-context. arXiv preprint arXiv:2502.08550, 2025.

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. Learning from mistakes makes llm better reasoner. arXiv preprint arXiv:2310.20689, 2023.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. arXiv preprint arXiv:1907.02893, 2019.

Art of Problem Solving Foundation. Amc23 — 2023 american mathematics competitions test set. `https://github.com/QwenLM/Qwen2.5-Math/tree/main/evaluation/data/amc23`, 2023. 40 problems drawn from the 2023 AMC 12 contests.

Olivier Bousquet and André Elisseeff. Stability and generalization. Journal of machine learning research, 2(Mar):499–526, 2002.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. arXiv preprint arXiv:2501.17161, 2025.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261, 2025.

Rohan Deb, Kiran Thekumparampil, Kousha Kalantari, Gaurush Hiranandani, Shoham Sabach, and Branislav Kveton. Fishersft: Data-efficient supervised fine-tuning of language models using information gain. arXiv preprint arXiv:2505.14826, 2025.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang,

Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. arXiv e-prints, pp. arXiv–2407, 2024.

Xiang Gao and Kamalika Das. Customizing language model responses with contrastive in-context learning. In Proceedings of the aaai conference on artificial intelligence, volume 38, pp. 18039–18046, 2024.

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.

Sonam Gupta, Yatin Nandwani, Asaf Yehudai, Dinesh Khandelwal, Dinesh Raghu, and Sachindra Joshi. Selective self-to-supervised fine-tuning for generalization in large language models. arXiv preprint arXiv:2502.08130, 2025.

Shadi Hamdan and Deniz Yuret. How much do llms learn from negative examples? arXiv preprint arXiv:2503.14391, 2025.

Seungwook Han, Jyothish Pari, Samuel J Gershman, and Pulkit Agrawal. General reasoning requires learning to reason from the get-go. arXiv preprint arXiv:2502.19402, 2025.

Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In International conference on machine learning, pp. 1225–1234. PMLR, 2016.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. arXiv preprint arXiv:2402.14008, 2024.

Yancheng He, Shilong Li, Jiaheng Liu, Weixun Wang, Xingyuan Bu, Ge Zhang, Zhongyuan Peng, Zhaoxiang Zhang, Zhicheng Zheng, Wenbo Su, et al. Can large language models detect errors in long chain-of-thought reasoning? arXiv preprint arXiv:2502.19361, 2025.

Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. Aligning ai with shared human values. Proceedings of the International Conference on Learning Representations (ICLR), 2021a.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. Proceedings of the International Conference on Learning Representations (ICLR), 2021b.

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, 2021. URL https://arxiv. org/abs/2103.03874, 2, 2024.

Roger A Horn and Charles R Johnson. Matrix analysis. Cambridge university press, 2012.

Maggie Huan, Yuetai Li, Tuney Zheng, Xiaoyu Xu, Seungone Kim, Minxin Du, Radha Poovendran, Graham Neubig, and Xiang Yue. Does math reasoning improve general llm capabilities? understanding transferability of llm reasoning. arXiv preprint arXiv:2507.00432, 2025.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. arXiv preprint arXiv:1609.04836, 2016.

Harsha Kokel, Michael Katz, Kavitha Srinivas, and Shirin Sohrabi. Acpbench: Reasoning about action, change, and planning. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 39, pp. 26559–26568, 2025.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. Advances in neural information processing systems, 35:3843–3857, 2022.

Dacheng Li, Shiyi Cao, Tyler Griggs, Shu Liu, Xiangxi Mo, Eric Tang, Sumanth Hegde, Kourosh Hakhamaneshi, Shishir G Patil, Matei Zaharia, et al. Llms can easily learn to reason from demonstrations structure, not content, is what matters! arXiv preprint arXiv:2502.07374, 2025.

Junyu Luo, Xiao Luo, Xiusi Chen, Zhiping Xiao, Wei Ju, and Ming Zhang. Semi-supervised fine-tuning for large language models. arXiv preprint arXiv:2410.14745, 2024a.

Junyu Luo, Xiao Luo, Kaize Ding, Jingyang Yuan, Zhiping Xiao, and Ming Zhang. Robustft: Robust supervised fine-tuning for large language models under noisy response. arXiv preprint arXiv:2412.14922, 2024b.

James Martens. New insights and perspectives on the natural gradient method. Journal of Machine Learning Research, 21(146):1–76, 2020.

Ivan Moshkov, Darragh Hanley, Ivan Sorokin, Shubham Toshniwal, Christof Henkel, Benedikt Schifferer, Wei Du, and Igor Gitman. Aimo-2 winning solution: Building state-of-the-art mathematical reasoning models with openmathreasoning dataset. arXiv preprint arXiv:2504.16891, 2025.

Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring generalization in deep learning. Advances in neural information processing systems, 30, 2017.

OpenAI. GPT-5 System Card. Technical report, OpenAI, August 2025. Accessed: 2025-08-10.

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.

Zhuoshi Pan, Yu Li, Honglin Lin, Qizhi Pei, Zinan Tang, Wei Wu, Chenlin Ming, H Vicky Zhao, Conghui He, and Lijun Wu. Lemma: Learning from errors for mathematical advancement in llms. arXiv preprint arXiv:2503.17439, 2025.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. arXiv preprint arXiv:2210.03350, 2022.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024a. URL https://arxiv.org/abs/2402.03300.

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

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. arXiv preprint arXiv:2210.09261, 2022.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

Yongqi Tong, Dawei Li, Sizhe Wang, Yujia Wang, Fei Teng, and Jingbo Shang. Can llms learn from previous mistakes? investigating llms' errors to boost for reasoning. arXiv preprint arXiv:2403.20046, 2024.

David Vilares and Carlos Gómez-Rodríguez. Head-qa: A healthcare dataset for complex reasoning. arXiv preprint arXiv:1906.04701, 2019.

Renxi Wang, Haonan Li, Xudong Han, Yixuan Zhang, and Timothy Baldwin. Learning from failure: Integrating negative examples when fine-tuning large language models as agents. arXiv preprint arXiv:2402.11651, 2024a.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. Advances in Neural Information Processing Systems, 37:95266–95290, 2024b.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. arXiv preprint arXiv:2109.01652, 2021.

Hermann Weyl. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differentialgleichungen (mit einer anwendung auf die theorie der hohlraumstrahlung). Mathematische Annalen, 71(4):441–479, 1912.

Juncheng Wu, Sheng Liu, Haoqin Tu, Hang Yu, Xiaoke Huang, James Zou, Cihang Xie, and Yuyin Zhou. Knowledge or reasoning? a close look at how llms think across domains. arXiv preprint arXiv:2506.02126, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. arXiv preprint arXiv:2505.09388, 2025b.

Mutian Yang, Jiandong Gao, and Ji Wu. Decoupling knowledge and reasoning in llms: An exploration using cognitive dual-system theory. arXiv preprint arXiv:2507.18178, 2025c.

Erxin Yu, Jing Li, Ming Liao, Qi Zhu, Boyang Xue, Minghui Xu, Baojun Wang, Lanqing Hong, Fei Mi, and Lifeng Shang. Self-error-instruct: Generalizing from errors for llms mathematical reasoning. arXiv preprint arXiv:2505.22591, 2025a.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025b. URL https://arxiv.org/abs/2503.14476.

Yige Yuan, Teng Xiao, Shuchang Tao, Xue Wang, Jinyang Gao, Bolin Ding, and Bingbing Xu. Incentivizing reasoning from weak supervision. arXiv preprint arXiv:2505.20072, 2025.

Chengshuai Zhao, Zhen Tan, Pingchuan Ma, Dawei Li, Bohan Jiang, Yancheng Wang, Yingzhen Yang, and Huan Liu. Is chain-of-thought reasoning of llms a mirage? a data distribution lens. arXiv preprint arXiv:2508.01191, 2025.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL https://arxiv.org/abs/2507.18071.

A Appendix

A.1 Experiments Setup

Distillation data curation We conduct experiments on mathematical reasoning and common sense, using Qwen3-8B (Yang et al., 2025b) to distill reasoning trajectories. For mathematics, we collect data from OpenMathReasoning (Moshkov et al., 2025), and for common sense from MMLU (Hendrycks et al., 2021b;a). Each trajectory is labeled as positive if the final answer matches the ground truth and negative otherwise. To ensure that all samples preserve complete reasoning structures and differ only in correctness, we discard instances exceeding 8,192 tokens. We then sample positive and negative data in a 1:1 ratio, resulting in 7.2k instances for mathematics and 17.4k for common sense.

Training Details We conduct experiments on the Qwen2.5 series (3B, 7B, 14B, 32B) (Team, 2024) and LLaMA-3.1-8B(Dubey et al., 2024). All models are fine-tuned for 20 epochs with a batch size of 128, using a cosine learning rate scheduler with 10% warm-up steps and a maximum learning rate of 5×10^{-5}. We set the training length to 20 epochs, as the loss does not converge earlier and benchmark performance continues to improve up to this point.

Evaluation Details Following Huan et al. (2025); Yuan et al. (2025), we evaluate models on three categories of benchmarks:(1) **mathematical reasoning**: MATH500 (Hendrycks et al., 2024), OlympiaBench (He et al., 2024), MinervaMath (Lewkowycz et al., 2022), and the competition-level AMC2023 (Art of Problem Solving Foundation, 2023); (2) **common sense reasoning**: MMLU, MMLU-Pro (Wang et al., 2024b), and BBH (Suzgun et al., 2022); (3) **other OOD reasoning**: ACPBench (Kokel et al., 2025) for planning, and HeadQA (Vilares & Gómez-Rodríguez, 2019) for medicine. Model performance is measured by accuracy. Evaluation uses the codebase from (Yuan et al., 2025), with sampling temperature 0.6, top-p 0.95, one sample per input, and max generation length 32,768 tokens.

We define in-domain and out-of-domain (OOD) evaluation based on the training data distribution. For models fine-tuned on mathematical reasoning tasks, in-domain evaluation uses mathematical problems while OOD evaluation employs other task categories. Conversely, models trained on MMLU are evaluated in-domain on commonsense tasks and OOD on the remaining domains. We compare three training strategies: using only positive samples, only negative samples, and a balanced combination of both.

A.2 Detailed taxonomy of negative training samples

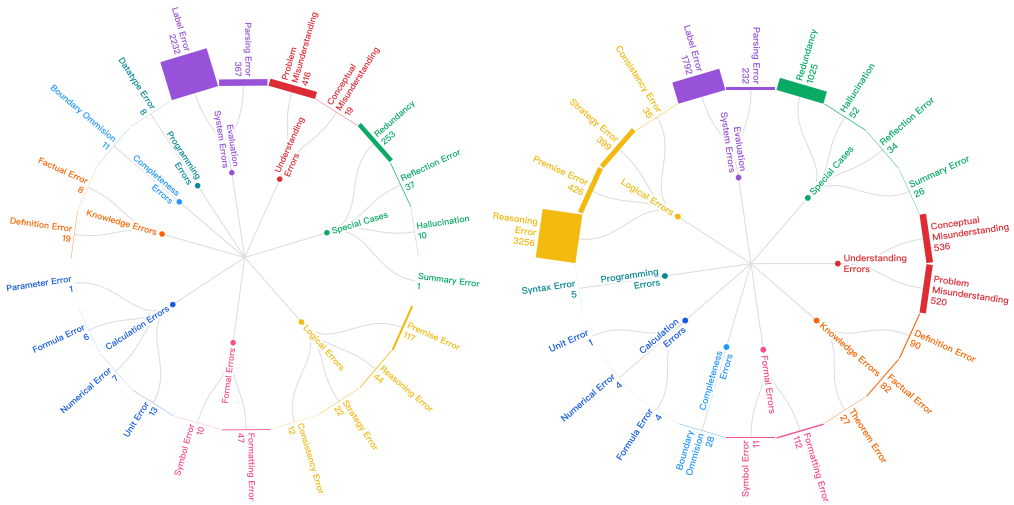
We provide statistics on the detailed categorization of negative samples in our training dataset. As shown in Figure 4a and Figure 4b, the error types of samples from OpenMathReasoning and MMLU that are not selected by reject sampling can be grouped into nine major categories and twenty-four subcategories. Although the distribution across categories is imbalanced, the errors still exhibit a broad coverage, ensuring a comprehensive representation of error types.

A.3 Prompt for Categorize Negative Samples

We design a structured prompt to categorize each erroneous reasoning trajectory into a fine-grained error class. The classification framework contains 9 primary categories and 22 sub-categories. The full classification schema and the prompt used for categorization are shown in Figure 5.

A.4 Case Study

As discussed in Section 4.3, negative trajectories exhibit higher entropy than positives ones on certain reasoning tokens and transition words. For illustration, we select one case and highlight the high-entropy segments. The results show that negatives contain substantially more such reasoning-related high-entropy fragments than positives.



(a) Error distribution in OpenMathReasoning. (b) Error distribution in MMLU.

Figure 4: Detailed categorization of negative samples in OpenMathReasoning and MMLU.

A.5 Detailed theoretical derivation

We provide a detailed derivation justifying why the dynamic reweighting mechanism in Eq. 6 improves optimization conditioning and, under standard assumptions, leads to tighter generalization bounds. The proof is organized as a sequence of lemmas linking (i) the loss reduction statistic $\Delta_i^{(t)}$ to gradient alignment, (ii) the weight update to a positive semi-definite augmentation of the empirical Fisher, (iii) the augmentation to a quantitative improvement of spectral properties of the Fisher (and hence of the local Hessian under a mild approximation), and (iv) improved spectral properties to better stability and generalization.

A.5.1 Notation and standing assumptions

We keep the notation from the main text. In particular $g_i^{(t)} = \nabla_\theta \ell_i(\theta^{(t)})$ is the gradient of the per-sample loss, $G^{(t)} = \sum_i w_i^{(t)} g_i^{(t)}$ is the weighted aggregate update direction, and $H_i(\theta) = \nabla_\theta^2 \ell_i(\theta)$ is the per-sample Hessian. We will drop the time superscript when it is clear from context.

We make the following standard (explicit) assumptions, which will be referenced in the lemmas below:

- (A1) Each $\ell_i(\theta)$ is twice differentiable and L-smooth: for all θ, $\|H_i(\theta)\|_{\mathrm{op}} \le L$.
- (A2) Gradient norms are uniformly bounded: $\|g_i(\theta)\|_2 \le G_{\max}$ for all i, θ.
- (A3) The learning rate η is small enough that second-order terms are controllable; concrete bounds will be given where needed.
- (A4) The empirical (weighted) Fisher is defined as
$$F_w \;=\; \frac{1}{N}\sum_{i=1}^{N} w_i\, g_i g_i^\top.$$
- (A5) (Fisher–Hessian closeness) In the local region of interest we have
$$\|\nabla^2 R(\theta) - F_w\|_{\mathrm{op}} \le \delta,$$
where $R(\theta) = \frac{1}{N}\sum_i \ell_i(\theta)$ and $\delta \ge 0$ is (assumed) small.

Lemma 1 (Taylor relation between Δ_i and gradient alignment)

Under (A1)–(A3), a single gradient update $\theta \leftarrow \theta - \eta G$ satisfies, for each sample i,
$$\Delta_i \;=\; \ell_i(\theta) - \ell_i(\theta - \eta G) = \eta\, g_i^\top G - \frac{1}{2}\eta^2\, G^\top H_i(\xi_i) G, \tag{9}$$

Prompt for Categorizing Negative Samples

```
You are an expert AI assistant tasked with identifying the single,
most specific error category from the list below.

Error Category List:
- Primary_category: Understanding Errors
  - sub_category: Problem Misunderstanding, Conceptual
      Misunderstanding
- Primary_category: Knowledge Errors
  - sub_category: Factual Error, Theorem Error, Definition Error
- Primary_category: Logical Errors
  - sub_category: Strategy Error, Reasoning Error, Premise Error,
      Consistency Error
- Primary_category: Calculation Errors
  - sub_category: Numerical Error, Formula Error, Parameter Error,
      Unit Error
- Primary_category: Programming Errors
  - sub_category: Syntax Error, Function Error, Data Type Error
- Primary_category: Formal Errors
  - sub_category: Symbol Error, Formatting Error
- Primary_category: Completeness Errors
  - sub_category: Boundary Omission
- Primary_category: Special Cases
  - sub_category: Reflection Error, Summary Error, Hallucination,
      Redundancy
- Primary_category: Evaluation System Errors
  - sub_category: Incorrect Ground Truth, Correct Answer Parsing
      Error

Data for Analysis:
- Question: {question}
- Ground Truth Answer: {groundtruth}
- Model's Reasoning Process (to be analyzed): {model_reasoning}

CRITICAL INSTRUCTION:
Analyze the provided reasoning process. Your response MUST be ONLY
    a single,
raw JSON object with the keys "sub_category" and "analysis". Do not
     include any
other text, explanations, apologies, or markdown formatting.

Example of a perfect response:
{
  "sub_category": "Premise Error",
  "analysis": "The model incorrectly assumed that all bicycles use
      plastic
            squares for identification, which is a flawed premise
                not
            supported by the question's context."
}
```

Figure 5: Prompt used for categorizing negative reasoning samples into predefined error subcategories.

Example from MATH-500

Problem: Denali and Nate work for a dog walking business and are paid for each dog they walk. Denali is responsible for 16 dogs and Nate is responsible for 12 dogs. Under the company's new policy, they will be assigned or unassigned new dogs in groups of x dogs. The ratio of Denali's pay to Nate's pay would be the same if Denali started walking $4x$ more dogs and Nate stayed at 12 dogs or if x of Nate's dogs were reassigned to Denali. Find x if $x \neq 0$. **Answer**: $\boxed{5}$

M_{pos} Thinking Trajectories

Okay, let me try to figure out this problem...
So, setting up the equation...
Let me compute both sides...

...would be the same under two different scenarios. Let me parse that...

So, solution is x = 0...

But the problem says ...

Let me check again...

x = 0 is correct...
The final answer is $\boxed{0}$.

M_{neg} Thinking Trajectories

Okay, let me try to figure out this problem...
The key is that...
Now, we can solve this...
..., so the answer is x = 3.
Wait, let me check...
it is wrong...

Alternatively, maybe there's a different way to interpret the problem?...
Maybe I missed something...

Alternatively, maybe there's a different interpretation? Let me check...
Wait, no. The problem says...
..., so the answer is 5.
But, Let me check ...
Therefore, x = 5 is correct...
The final answer is $\boxed{5}$.

Figure 6: Case study of thinking trajectories for M_{pos} and M_{neg} on the same question.

for some ξ_i on the line segment between θ and $\theta - \eta G$. Consequently, using $\|H_i\|_{\mathrm{op}} \le L$,

$$\left|\Delta_i - \eta\, g_i^\top G\right| \le \frac{1}{2} L \eta^2 \|G\|_2^2. \tag{10}$$

Proof. The equality equation 9 is a direct application of second-order Taylor expansion with integral/mean-value form of the remainder. The inequality equation 10 follows from $\left|G^\top H_i(\xi_i) G\right| \le \|H_i(\xi_i)\|_{\mathrm{op}} \|G\|_2^2 \le L\|G\|_2^2$. □

The leading term $\eta\, g_i^\top G$ indicates that Δ_i is large when g_i aligns with the aggregate direction G; the error is $O(\eta^2)$ and is controlled by smoothness.

LEMMA 2 (WEIGHT INCREMENT INDUCES PSD AUGMENTATION OF F_w)

Let the weights change by nonnegative increments $\delta w_i \ge 0$ for $i \in T \subseteq \{1, \dots, N\}$ (others unchanged). Then the resulting change in empirical Fisher is

$$\Delta F \;=\; \frac{1}{N} \sum_{i \in T} \delta w_i\, g_i g_i^\top,$$

which is positive semi-definite. Consequently, for the updated Fisher $F'_w = F_w + \Delta F$, Weyl's inequality implies that every eigenvalue of F'_w is at least the corresponding eigenvalue of F_w.

Proof. Each outer product $g_i g_i^\top$ is PSD and scaled by $\delta w_i \ge 0$, therefore the sum is PSD. The eigenvalue monotonicity under PSD additions is the classical Weyl inequality for symmetric matrices. □

LEMMA 3 (QUANTITATIVE IMPROVEMENT OF SMALL-EIGENVALUE SUBSPACE)

Let U be a k-dimensional subspace of $\mathbb{R}^d$ (with orthogonal projector P_U). Suppose the weight increments satisfy the energy condition

$$\mathrm{tr}\big(P_U \Delta F P_U\big) = \frac{1}{N} \sum_{i \in T} \delta w_i\, \|P_U g_i\|_2^2 \;\ge\; \gamma, \tag{11}$$

for some $\gamma > 0$. Denote by $\{\lambda_j(F_w|_U)\}_{j=1}^k$ the eigenvalues of F_w restricted to U (arranged in nonincreasing order). Then after augmentation the average eigenvalue on U increases by at least γ/k, and in particular the minimal eigenvalue on U satisfies

$$\lambda_{\min}\big(F'_w|_U\big) \;\ge\; \lambda_{\min}\big(F_w|_U\big) + \frac{\gamma}{k}.$$

Proof. Let $\{\mu_j\}_{j=1}^k$ be eigenvalues of $P_U F_w P_U$ and $\{\mu'_j\}$ those of $P_U F'_w P_U$. By linearity of trace,

$$\sum_{j=1}^{k} (\mu'_j - \mu_j) = \mathrm{tr}\big(P_U \Delta F P_U\big) \ge \gamma.$$

Hence the average eigenvalue increase is at least γ/k. Since the eigenvalues are ordered, the minimal eigenvalue increases by at least the average increase:

$$\mu'_k \ge \mu_k + \frac{\gamma}{k}.$$

This is the claimed bound. □

Remark. The condition equation 11 requires that the incremented weights inject nontrivial energy into the previously low-energy subspace U; without such a condition one cannot guarantee a rise of small eigenvalues (only non-decrease).

LEMMA 4 (TRANSFER FROM FISHER TO HESSIAN)

Under assumption (A5) (Fisher–Hessian closeness) we have, for the local Hessian $H(\theta) = \nabla^2 R(\theta)$ and the updated Fisher F'_w, that each eigenvalue of $H(\theta)$ increases by at least the corresponding increase in the Fisher eigenvalues minus δ; more precisely, if $\lambda_{\min}(F'_w|_U) - \lambda_{\min}(F_w|_U) \geq \Delta\lambda_F$ then

$$\lambda_{\min}(H'|_U) \;\geq\; \lambda_{\min}(H|_U) + \Delta\lambda_F - 2\delta,$$

where H' denotes the local Hessian after the small parameter change induced by the reweighted update and the factor 2δ accounts for the approximation error before and after the update.

Proof sketch. By assumption $\|H - F_w\|_{\text{op}} \leq \delta$ and (after update) $\|H' - F'_w\|_{\text{op}} \leq \delta$. Then

$$\lambda_{\min}(H'|_U) \geq \lambda_{\min}(F'_w|_U) - \delta \quad \text{and} \quad \lambda_{\min}(H|_U) \leq \lambda_{\min}(F_w|_U) + \delta,$$

hence

$$\lambda_{\min}(H'|_U) - \lambda_{\min}(H|_U) \geq \big(\lambda_{\min}(F'_w|_U) - \lambda_{\min}(F_w|_U)\big) - 2\delta.$$

This gives the stated inequality. □

LEMMA 5 (IMPROVED CONDITIONING REDUCES PARAMETER SENSITIVITY UNDER LOCAL STRONG CONVEXITY)

Assume in the local region that $R(\theta)$ satisfies the (restricted) Polyak–Lojasiewicz (PL) or local strong convexity condition on U: there exists $\mu > 0$ such that for any θ in the region the Hessian restricted to U obeys $\lambda_{\min}(H|_U) \geq \mu$. Consider two training sets that differ by one example and run identical reweighted updates; then (under standard Lipschitz assumptions) the parameter perturbation induced by this data change is controlled by $1/\mu$. Therefore increasing μ (equivalently increasing the minimal eigenvalue on U) reduces the algorithmic instability and yields a smaller generalization gap Bousquet & Elisseeff (2002); Hardt et al. (2016).

Proof sketch. In strongly convex settings the mapping from gradients to parameter updates is Lipschitz with constant proportional to $1/\mu$; small perturbations in the empirical risk result in parameter differences bounded by a constant times $1/\mu$. Standard uniform stability results then transfer this parameter sensitivity bound into an $O(1/\mu)$-type improvement in the generalization error bound; see Bousquet & Elisseeff (2002); Hardt et al. (2016) for formal statements. □

MAIN PROPOSITION

Under assumptions (A1)–(A5), if the weight update rule equation 6 yields nonnegative increments δw_i that satisfy the energy condition equation 11 for some subspace U of dimension k, then:

1. The empirical Fisher F_w receives a PSD augmentation ΔF and the average eigenvalue on U increases by at least γ/k (Lemma A.5.1).
2. If Fisher approximates Hessian within δ, the minimal eigenvalue of the local Hessian on U increases by at least $\gamma/k - 2\delta$ (Lemma A.5.1).
3. If the local loss satisfies a strong-convexity-type condition on U with parameter μ, then replacing μ by $\mu' = \mu + \gamma/k - 2\delta$ reduces parameter sensitivity and, by standard stability-to-generalization results, yields a quantitatively tighter generalization bound (Lemma A.5.1).

In summary, the proposed reweighting mechanism systematically amplifies underrepresented gradients associated with small Δ_i, which enriches the spectrum of the empirical second-moment matrix F_w. This, in turn, improves the conditioning of the local Hessian H, leading to more stable optimization dynamics and enhanced generalization.

A.6 HYPERPARAMETER SENSITIVITY OF GLOW

As shown in Figure 7, GLOW yields modest improvements over the full-SFT reference in most configurations. Varying α between 0.8 and 1.5 leads to small changes, and $\beta = 12$ is generally stronger than $\beta = 10$ or $\beta = 18$ at matched α. These results suggest incremental gains with moderate hyperparameter choices in our setup.

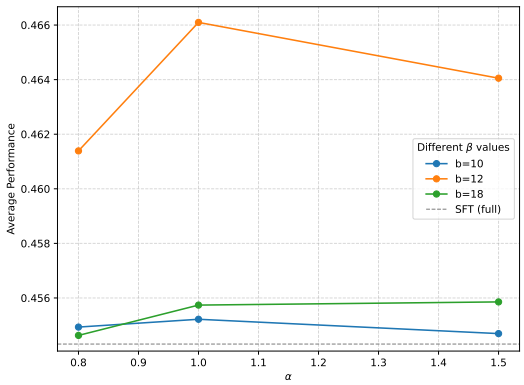


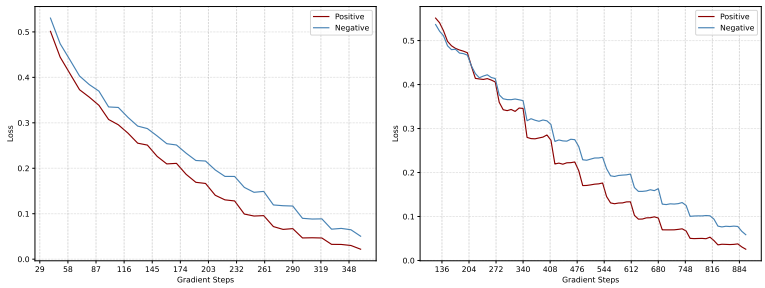
Figure 7: Ablation study on the hyperparameters α and β. GLOW exhibits stable performance across different settings, demonstrating the robustness of the reweighting formulation.

A.7 Training Loss on OpenMathReasoning and MMLU

We present in Figure 8 the loss comparison of all models trained under the positive and negative settings on the OpenMathReasoning and MMLU datasets.

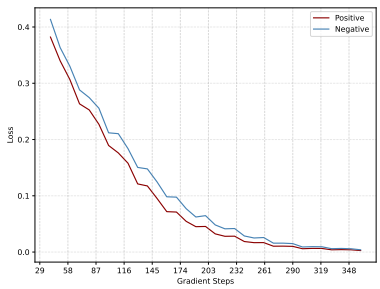
A.8 The Use of Large Language Models (LLMs)

We used LLMs only for copy-editing and minor stylistic polishing (grammar, phrasing, and LaTeX formatting). All suggestions were manually reviewed and edited by the authors. The authors take full responsibility for the manuscript's contents.

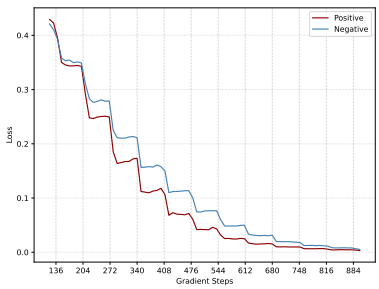


(a) Qwen2.5-3B on OpenMathReasoning

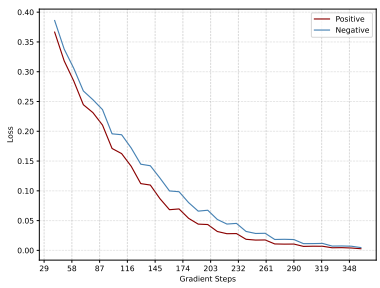
(b) Qwen2.5-3B on MMLU



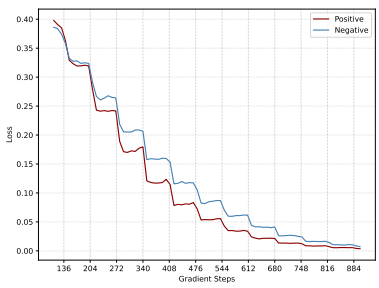
(c) Qwen2.5-7B on OpenMathReasoning



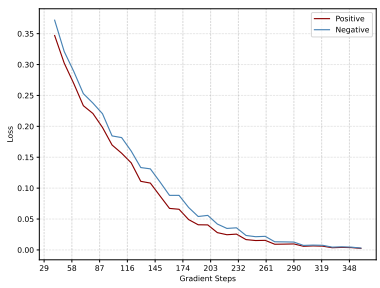
(d) Qwen2.5-7B on MMLU



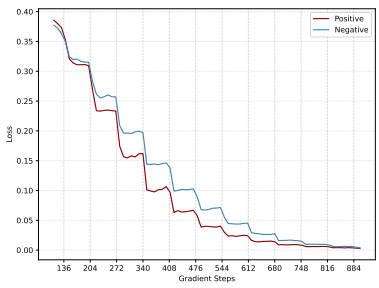
(e) Qwen2.5-14B on OpenMathReasoning



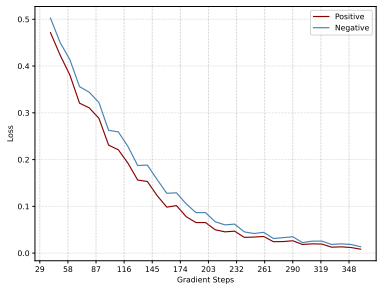
(f) Qwen2.5-14B on MMLU



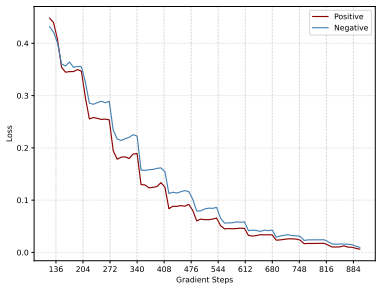
(g) Qwen2.5-32B on OpenMathReasoning



(h) Qwen2.5-32B on MMLU



(i) Llama3.1-8B on OpenMathReasoning



(j) Llama3.1-8B on MMLU

Figure 8: Training loss of Qwen2.5 models and Llama3.1-8B on OpenMathReasoning (left) and MMLU (right). Losses drop across epochs, with the positive setting converging faster than the negative.