

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

Anonymous authors

Paper under double-blind review

ABSTRACT

Supervised fine-tuning (SFT) on chain-of-thought (CoT) trajectories is a standard component of reasoning-oriented post-training for large language models. In current practice, such CoT-based SFT typically retains only trajectories whose final answers match the ground truth, which can lead to 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-Thought (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. **In this paradigm, we focus on the SFT stage and study the common practice of transferring reasoning ability via distilled CoT trajectories to a student model that does not initially exhibit strong reasoning behavior.**

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:** Current reasoning post-training pipelines predominantly perform SFT on CoT trajectories distilled from a stronger teacher model, and use rejection sampling to retain only trajectories whose final answers and formats match the ground truth. Discarding samples

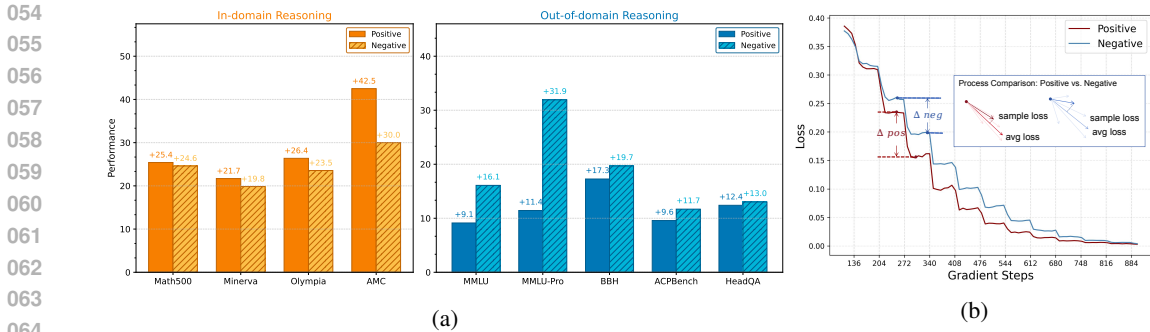


Figure 1: (a) Qwen2.5-14B trained with positive samples shows limited transfer beyond math, whereas models trained on negative samples generalize more broadly across reasoning tasks. Bars report the final accuracy on each benchmark, and the “+” annotations above them denote the absolute accuracy improvement over the model before SFT. (b) Training loss curves on MMLU for Qwen2.5-32B. The red curve corresponds to training on positive samples only, and the blue curve to training on negative samples only. Δ denotes the per-sample loss difference between epochs.

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 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.5), 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 negative trajectories can help generalization, training on negatives alone would still be a rejection-sampling scheme that discards large portions of data. This motivates seeking a method that learns from both correct and incorrect trajectories without filtering.** To address this, we introduce a dynamic mechanism called **Gain-based LOss Weighting (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 grouped into prompt-based, fine-tuning-based, and reinforcement-learning-based approaches. Prompt-based methods use negative examples to steer model behavior. Gao & Das (2024) employ them to encode ambiguous preferences that models should avoid, while Alazraki et al. (2025) show that inserting a negative example into the prompt can be more effective than adding an additional positive one, and that providing incorrect rationales may even over-constrain the model. However, the effectiveness of such methods is ultimately limited by the model’s own reasoning and instruction-following abilities. By contrast, fine-tuning-based approaches are more commonly used to strengthen reasoning or to provide a strong initialization for subsequent reinforcement learning (Guo et al., 2025). Some studies distill positive CoT trajectories from initially negative samples using teacher models (Yu et al., 2025a; Pan et al., 2025; An et al., 2023), whereas others introduce explicit prefixes to distinguish positive from negative samples (Wang et al., 2024a; Tong et al., 2024). Beyond SFT, recent reinforcement-learning (RL) methods for reasoning language models also explore how to exploit negative signals. Examples include decomposing RL with verifiable rewards into separate positive and negative reinforcement (Zhu et al., 2025), reactivating residual prompts through exploration (Liu et al., 2025), mining useful steps within otherwise incorrect trajectories (Yang et al., 2025d), and converting homogeneous errors into informative gradients (Nan et al., 2025). Nevertheless, in both SFT and RL, negative samples are still typically treated as less valuable than positive ones and are used mainly as penalties, down-weighted rewards, or auxiliary signals.

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 **general reasoning** 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. Colored cells highlight entries that support our findings: **orange** cells mark in-domain benchmarks where positives outperform negatives, and **blue** cells mark out-of-domain benchmarks where negatives outperform positives. Within each positive/negative pair, the higher score is additionally highlighted in the corresponding color.

Model	Setting	Math Reasoning (In-Domain)					General Reasoning (Out-of-Domain)				Other Reasoning (Out-of-Domain)		
		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
	$\Delta(\text{pos-neg})$	+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
	$\Delta(\text{pos-neg})$	+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
	$\Delta(\text{pos-neg})$	+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
	$\Delta(\text{pos-neg})$	-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
	$\Delta(\text{pos-neg})$	+3.40	-0.37	+1.18	-7.50	-0.82	-20.19	-13.07	-9.78	-14.35	-2.11	-17.36	-9.74

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 **general reasoning** 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.

Table 2: Cross-domain performance of models trained on the **general reasoning** dataset. ‘‘Avg.’’ denotes the average score within each group. Colored cells highlight entries that support our findings: **orange** cells mark in-domain benchmarks where positives outperform negatives, and **blue** cells mark out-of-domain benchmarks where negatives outperform positives. Within each positive/negative pair, the higher score is additionally highlighted in the corresponding color.

Model	Setting	Math Reasoning (Out-of-Domain)					General Reasoning (In-Domain)				Other Reasoning (Out-of-Domain)		
		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
	$\Delta(\text{pos-neg})$	-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
	$\Delta(\text{pos-neg})$	-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
	$\Delta(\text{pos-neg})$	-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
	$\Delta(\text{pos-neg})$	-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
	$\Delta(\text{pos-neg})$	-7.40	-4.42	-2.82	-2.50	-4.29	+0.10	+0.89	-7.84	-2.28	-7.54	-2.11	-4.83

4 WHY NEGATIVE IS BETTER

To better understand why negative samples benefit out-of-distribution generalization, we analyze this phenomenon step by step. Empirically, we observe that correct trajectories usually share a few common success factors (such as accurate computation and proper problem understanding), whereas the reasons for failure are much more diverse. We therefore first examine the data to characterize how negative samples 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.5 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.3.

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 phenomenon can be understood more formally through the lens of Invariant Risk Minimization (IRM) (Arjovsky et al., 2019). IRM posits that generalization improves when a model learns representations that capture invariant causal structure across diverse environments. In our setting, we interpret different categories of incorrect reasoning as different environments: each error type induces its own sub-distribution over inputs and outputs, with characteristic failure patterns that de-

270 fine a distinct local data distribution. Importantly, many negative samples still contain partially valid
 271 reasoning paths, as illustrated in Figure 8, so these environments are far from pure noise. Exposure
 272 to many such environments requires the model to perform well under varied failure modes, which
 273 in turn encourages it to learn reasoning features that remain stable across them.

274 Formally, let \mathcal{E} denote the set of environments induced by these negative error categories, and let
 275 each $e \in \mathcal{E}$ correspond to a data distribution D^e over input–output sequences (x, y) . We decompose
 276 the language model into a shared sequence representation Φ and a shared next-token predictor w ,
 277 where Φ represents the main layers of the model and w is the vocabulary projection head. IRM
 278 in the autoregressive setting requires that the same predictor w be optimal across all environments
 279 when paired with Φ :

$$280 \min_{\Phi} \sum_{e \in \mathcal{E}} R^e(w \circ \Phi) \quad \text{subject to} \quad w \in \arg \min_{w'} R^e(w' \circ \Phi), \forall e \in \mathcal{E}, \quad (1)$$

281 where the per-environment autoregressive risk is

$$282 R^e(w \circ \Phi) = \mathbb{E}_{(x,y) \sim D^e} \left[\sum_{t=1}^{|y|} \ell \left(w(\Phi(x, y_{<t})), y_t \right) \right], \quad (2)$$

283 and ℓ denotes the cross-entropy loss. Because the predictor w is shared across all environments,
 284 achieving optimality requires the representation Φ to encode reasoning features that remain reliable
 285 under different types of errors. This provides a conceptual explanation of why diverse negative
 286 samples, which span many environments, can improve the robustness and out-of-distribution gener-
 287 alization of the learned reasoning patterns.

288 From this perspective, positive samples are clean and correct but occupy a relatively narrow range
 289 of environments, which limits their ability to support invariance. Negative samples, in contrast,
 290 cover multiple environments and expose diverse failure modes within otherwise valid reasoning
 291 structures. This diversity pushes the model to learn more robust representations that generalize
 292 across heterogeneous reasoning scenarios.

293 4.2 TRAINING PERSPECTIVE

294 To characterize the learning dynam-
 295 ics, we plot the training loss every 10
 296 steps for all models fine-tuned on posi-
 297 tive and negative samples from math
 298 reasoning and MMLU. We present
 299 Qwen2.5-32B (Figure 1b) as a repre-
 300 sentative example, while others are pro-
 301 vided in Appendix A.8. The curves
 302 follow a consistent stage-wise pattern.
 303 Loss drops sharply at the end of each
 304 epoch for positive samples, leading
 305 to faster initial convergence, whereas
 306 negative samples produce a smoother,
 307 gradual decline that ultimately reaches a comparable loss floor. This is because the optimization
 308 directions of individual samples align more consistently with the average gradient in the positive
 309 set than in the negative set, while negative samples point to a wider exploration space. We quantify
 310 this behavior using the average loss difference between consecutive epochs, as reported in Table 4,
 311 which confirms that positives decrease faster in the early stages. The negative-trajectory loss de-
 312 creases consistently throughout training and follows a trend similar to the positives (see Figure 1b
 313 and Figure 9). More importantly, this is accompanied by steady improvements on held-out evalua-
 314 tions at 5/10/15 epochs (see Table 9 and appendix A.9), suggesting that negatives act as learnable
 315 supervision rather than pure noise. They encode diverse exploratory patterns where incorrect an-
 316 swers coexist with partially valid reasoning, offering sustained constraints that encourage the model
 317 to develop more robust reasoning strategies instead of memorizing a single correct trajectory.

318 These results show that the value of negative samples lies in their diversity. Although this slows loss
 319 reduction by introducing varied optimization directions, it compels the model to explore a broader
 320 reasoning space and converge to more generalizable patterns.
 321

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	$\Delta_{\text{avg_loss}}^{\text{epoch } 2-1}$	$\Delta_{\text{avg_loss}}^{\text{epoch } 3-2}$	$\Delta_{\text{avg_loss}}^{\text{epoch } 4-3}$	$\Delta_{\text{avg_loss}}^{\text{epoch } 5-4}$
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

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’’).

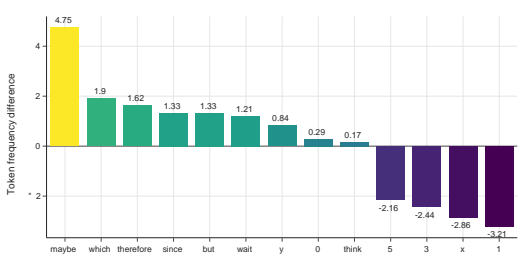


Figure 2: Token frequency differences between M_{neg} and M_{pos} on digits and high-entropy tokens.

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

Model	Setting	Data	\bar{H}_{think}	\bar{H}_{ans}	ΔH
M_{pos}	Off-policy	Math	0.909	0.708	0.202
		Other	1.138	0.873	0.265
M_{pos}	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
M_{neg}	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 $\mathcal{T}_{think}^{(i)}$ and $\mathcal{T}_{ans}^{(i)}$ 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:

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

as well as the entropy drop across the boundary:

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

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 7, 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 use $\ell_i^{(t)}$ to denote the loss of sample i at epoch t . 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 utility, while large reductions imply saturated learning with limited marginal utility. We therefore use $\Delta_i^{(t)} = \ell_i^{(t-1)} - \ell_i^{(t)}$ 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 $\Delta_i^{(t)}$:

$$w_i^{(t)} = \alpha(1 - \sigma(\beta\Delta_i^{(t)})), \quad (6)$$

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:

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

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

$$\Delta_i^{(t)} = \ell_i^{(t-1)} - \ell_i^{(t)} \approx \eta g_i^{(t-1)\top} G^{(t-1)} - \frac{1}{2} \eta^2 G^{(t-1)\top} H_i G^{(t-1)}, \quad (8)$$

where H_i is the Hessian of model parameters. The leading term shows that $\Delta_i^{(t)}$ is large if $g_i^{(t-1)}$ aligns with the dominant descent directions $G^{(t-1)}$, 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 a tighter generalization bound, 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 enlarges 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.2.

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. **Bold** indicates the best results under the same model.

Model	Setting	Math Reasoning (In-Domain)					General Reasoning (Out-of-Domain)				Other Reasoning (Out-of-Domain)		
		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
Qwen2.5-32B	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
	GLOW	93.40	54.41	59.11	92.50	74.86	85.51	74.14	83.98	81.21	51.97	85.19	68.58
Llama3.1-8B	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
	GLOW	44.60	20.59	15.11	17.50	24.45	63.80	38.34	58.17	53.44	35.04	66.70	50.87

Table 7: Cross-domain performance of models trained on the **general reasoning** dataset. “Avg.” denotes the average score within each group. **Bold** indicates the best results under the same model.

Model	Setting	Math Reasoning (Out-of-Domain)					General Reasoning (In-Domain)				Other Reasoning (Out-of-Domain)		
		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
Qwen2.5-32B	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
	GLOW	89.00	47.06	58.67	82.50	69.31	80.81	64.72	81.98	75.84	52.08	83.73	67.91
Llama3.1-8B	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
	GLOW	24.80	20.59	6.96	12.50	16.21	68.52	42.96	57.53	56.33	39.72	72.57	56.15

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 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 general reasoning. Models trained on the general reasoning dataset also exhibit 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.

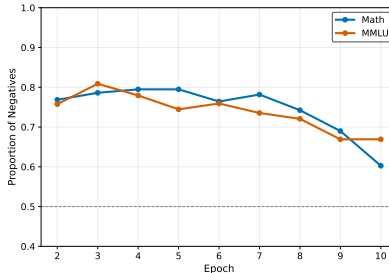


Figure 3: Fraction of negatives in the subset with the larger weights over epochs on Math and MMLU training.

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}_{think}	\bar{H}_{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](https://arxiv.org/abs/2402.00157), 2024.

- 540 Lisa Alazraki, Maximilian Mozes, Jon Ander Campos, Tan Yi-Chern, Marek Rei, and Max Bartolo.
541 No need for explanations: LLMs can implicitly learn from mistakes in-context. [arXiv preprint](#)
542 [arXiv:2502.08550](#), 2025.
- 543 Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. Learning
544 from mistakes makes LLM better reasoner. [arXiv preprint arXiv:2310.20689](#), 2023.
- 545 Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization.
546 [arXiv preprint arXiv:1907.02893](#), 2019.
- 547 Art of Problem Solving Foundation. Amc23 — 2023 american mathematics competitions test
548 set. [https://github.com/QwenLM/Qwen2.5-Math/tree/main/evaluation/](https://github.com/QwenLM/Qwen2.5-Math/tree/main/evaluation/data/amc23)
549 [data/amc23](https://github.com/QwenLM/Qwen2.5-Math/tree/main/evaluation/data/amc23), 2023. 40 problems drawn from the 2023 AMC 12 contests.
- 550 Olivier Bousquet and André Elisseeff. Stability and generalization. [Journal of machine learning](#)
551 [research](#), 2(Mar):499–526, 2002.
- 552 Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V
553 Le, Sergey Levine, and Yi Ma. Sft memorizes, rl generalizes: A comparative study of foundation
554 model post-training. [arXiv preprint arXiv:2501.17161](#), 2025.
- 555 Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit
556 Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the
557 frontier with advanced reasoning, multimodality, long context, and next generation agentic capa-
558 bilities. [arXiv preprint arXiv:2507.06261](#), 2025.
- 559 Rohan Deb, Kiran Thekumparampil, Kousha Kalantari, Gaurush Hiranandani, Shoham Sabach, and
560 Branislav Kveton. Fishersft: Data-efficient supervised fine-tuning of language models using in-
561 formation gain. [arXiv preprint arXiv:2505.14826](#), 2025.
- 562 DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu,
563 Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu,
564 Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao
565 Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan,
566 Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao,
567 Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding,
568 Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang
569 Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai
570 Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang,
571 Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang,
572 Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang,
573 Qiancheng Wang, Qinyu Chen, Qiusi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang,
574 R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng
575 Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing
576 Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjin Zhao, Wen
577 Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong
578 Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu,
579 Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xi-
580 aosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia
581 Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng
582 Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong
583 Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong,
584 Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou,
585 Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying
586 Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda
587 Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu,
588 Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu
589 Zhang, and Zhen Zhang. Deepseek-rl: Incentivizing reasoning capability in LLMs via reinforce-
590 ment learning, 2025. URL <https://arxiv.org/abs/2501.12948>.
- 591 Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha
592 Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models.
593 [arXiv e-prints](#), pp. [arXiv-2407](#), 2024.

- 594 Xiang Gao and Kamalika Das. Customizing language model responses with contrastive in-context
595 learning. In Proceedings of the aaai conference on artificial intelligence, volume 38, pp. 18039–
596 18046, 2024.
- 597
598 Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu,
599 Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms
600 via reinforcement learning. arXiv preprint arXiv:2501.12948, 2025.
- 601
602 Sonam Gupta, Yatin Nandwani, Asaf Yehudai, Dinesh Khandelwal, Dinesh Raghu, and Sachindra
603 Joshi. Selective self-to-supervised fine-tuning for generalization in large language models. arXiv
604 preprint arXiv:2502.08130, 2025.
- 605
606 Shadi Hamdan and Deniz Yuret. How much do llms learn from negative examples? arXiv preprint
607 arXiv:2503.14391, 2025.
- 608
609 Seungwook Han, Jyothish Pari, Samuel J Gershman, and Pulkit Agrawal. General reasoning requires
610 learning to reason from the get-go. arXiv preprint arXiv:2502.19402, 2025.
- 611
612 Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic
613 gradient descent. In International conference on machine learning, pp. 1225–1234. PMLR, 2016.
- 614
615 Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu,
616 Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for
617 promoting agi with olympiad-level bilingual multimodal scientific problems. arXiv preprint
618 arXiv:2402.14008, 2024.
- 619
620 Yancheng He, Shilong Li, Jiaheng Liu, Weixun Wang, Xingyuan Bu, Ge Zhang, Zhongyuan Peng,
621 Zhaoxiang Zhang, Zhicheng Zheng, Wenbo Su, et al. Can large language models detect errors in
622 long chain-of-thought reasoning? arXiv preprint arXiv:2502.19361, 2025.
- 623
624 Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob
625 Steinhardt. Aligning ai with shared human values. Proceedings of the International Conference
626 on Learning Representations (ICLR), 2021a.
- 627
628 Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Ja-
629 cob Steinhardt. Measuring massive multitask language understanding. Proceedings of the
630 International Conference on Learning Representations (ICLR), 2021b.
- 631
632 Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song,
633 and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021.
634 URL <https://arxiv.org/abs/2103.03874>, 2, 2024.
- 635
636 Roger A Horn and Charles R Johnson. Matrix analysis. Cambridge university press, 2012.
- 637
638 Maggie Huan, Yuetai Li, Tuney Zheng, Xiaoyu Xu, Seungone Kim, Minxin Du, Radha Pooven-
639 dran, Graham Neubig, and Xiang Yue. Does math reasoning improve general llm capabilities?
640 understanding transferability of llm reasoning. arXiv preprint arXiv:2507.00432, 2025.
- 641
642 Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Pe-
643 ter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. arXiv
644 preprint arXiv:1609.04836, 2016.
- 645
646 Harsha Kokel, Michael Katz, Kavitha Srinivas, and Shirin Sohrabi. Acpbench: Reasoning about
647 action, change, and planning. In Proceedings of the AAAI Conference on Artificial Intelligence,
648 volume 39, pp. 26559–26568, 2025.
- 649
650 Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ra-
651 masesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative
652 reasoning problems with language models. Advances in neural information processing systems,
653 35:3843–3857, 2022.
- 654
655 Dacheng Li, Shiyi Cao, Tyler Griggs, Shu Liu, Xiangxi Mo, Eric Tang, Sumanth Hegde, Kourosh
656 Hakhamaneshi, Shishir G Patil, Matej Zaharia, et al. Llms can easily learn to reason from demon-
657 strations structure, not content, is what matters! arXiv preprint arXiv:2502.07374, 2025.

- 648 Chenxi Liu, Junjie Liang, Yuqi Jia, Bochuan Cao, Yang Bai, Heng Huang, and Xun Chen. Ex-
649 plore data left behind in reinforcement learning for reasoning language models. [arXiv preprint](#)
650 [arXiv:2511.04800](#), 2025.
- 651 Junyu Luo, Xiao Luo, Xiushi Chen, Zhiping Xiao, Wei Ju, and Ming Zhang. Semi-supervised fine-
652 tuning for large language models. [arXiv preprint arXiv:2410.14745](#), 2024a.
- 653 Junyu Luo, Xiao Luo, Kaize Ding, Jingyang Yuan, Zhiping Xiao, and Ming Zhang. Robustft:
654 Robust supervised fine-tuning for large language models under noisy response. [arXiv preprint](#)
655 [arXiv:2412.14922](#), 2024b.
- 656 James Martens. New insights and perspectives on the natural gradient method. [Journal of Machine](#)
657 [Learning Research](#), 21(146):1–76, 2020.
- 660 Ivan Moshkov, Darragh Hanley, Ivan Sorokin, Shubham Toshniwal, Christof Henkel, Benedikt
661 Schifferer, Wei Du, and Igor Gitman. Aimo-2 winning solution: Building state-of-the-art math-
662 ematical reasoning models with openmathreasoning dataset. [arXiv preprint arXiv:2504.16891](#),
663 2025.
- 664 Gongrui Nan, Siye Chen, Jing Huang, Mengyu Lu, Dexun Wang, Chunmei Xie, Weiqi Xiong,
665 Xianzhou Zeng, Qixuan Zhou, Yadong Li, and Xingzhong Xu. NGRPO: Negative-enhanced
666 group relative policy optimization. [arXiv preprint arXiv:2509.18851](#), 2025.
- 667 Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring general-
668 ization in deep learning. [Advances in neural information processing systems](#), 30, 2017.
- 669 OpenAI. GPT-5 System Card. Technical report, OpenAI, August 2025. Accessed: 2025-08-10.
- 670 Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong
671 Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to fol-
672 low instructions with human feedback. [Advances in neural information processing systems](#), 35:
673 27730–27744, 2022.
- 674 Zhuoshi Pan, Yu Li, Honglin Lin, Qizhi Pei, Zinan Tang, Wei Wu, Chenlin Ming, H Vicky Zhao,
675 Conghui He, and Lijun Wu. Lemma: Learning from errors for mathematical advancement in llms.
676 [arXiv preprint arXiv:2503.17439](#), 2025.
- 677 Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring
678 and narrowing the compositionality gap in language models. [arXiv preprint arXiv:2210.03350](#),
679 2022.
- 680 Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang,
681 Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of
682 mathematical reasoning in open language models, 2024a. URL [https://arxiv.org/abs/
683 2402.03300](https://arxiv.org/abs/2402.03300).
- 684 Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang,
685 Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathemati-
686 cal reasoning in open language models. [arXiv preprint arXiv:2402.03300](#), 2024b.
- 687 Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung,
688 Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks
689 and whether chain-of-thought can solve them. [arXiv preprint arXiv:2210.09261](#), 2022.
- 690 Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL [https://qwenlm.
691 github.io/blog/qwen2.5/](https://qwenlm.github.io/blog/qwen2.5/).
- 692 Yongqi Tong, Dawei Li, Sizhe Wang, Yujia Wang, Fei Teng, and Jingbo Shang. Can llms learn
693 from previous mistakes? investigating llms’ errors to boost for reasoning. [arXiv preprint](#)
694 [arXiv:2403.20046](#), 2024.
- 695 David Vilares and Carlos Gómez-Rodríguez. Head-qa: A healthcare dataset for complex reasoning,
696 [arXiv preprint arXiv:1906.04701](#), 2019.

- 702 Renxi Wang, Haonan Li, Xudong Han, Yixuan Zhang, and Timothy Baldwin. Learning from failure:
703 Integrating negative examples when fine-tuning large language models as agents. [arXiv preprint](#)
704 [arXiv:2402.11651](#), 2024a.
- 705 Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming
706 Ren, Aaran Arulraj, Xuan He, Ziyang Jiang, et al. Mmlu-pro: A more robust and challenging multi-
707 task language understanding benchmark. [Advances in Neural Information Processing Systems](#),
708 37:95266–95290, 2024b.
- 709 Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du,
710 Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. [arXiv preprint](#)
711 [arXiv:2109.01652](#), 2021.
- 712 Hermann Weyl. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differen-
713 tialgleichungen (mit einer anwendung auf die theorie der hohlraumstrahlung). [Mathematische](#)
714 [Annalen](#), 71(4):441–479, 1912.
- 715 Juncheng Wu, Sheng Liu, Haoqin Tu, Hang Yu, Xiaoke Huang, James Zou, Cihang Xie, and Yuyin
716 Zhou. Knowledge or reasoning? a close look at how llms think across domains. [arXiv preprint](#)
717 [arXiv:2506.02126](#), 2025.
- 718 An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang
719 Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu,
720 Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin
721 Yang, Jiayi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang,
722 Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui
723 Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang
724 Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger
725 Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan
726 Qiu. Qwen3 technical report, 2025a. URL <https://arxiv.org/abs/2505.09388>.
- 727 An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu,
728 Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. [arXiv preprint](#)
729 [arXiv:2505.09388](#), 2025b.
- 730 Mutian Yang, Jiandong Gao, and Ji Wu. Decoupling knowledge and reasoning in llms: An explo-
731 ration using cognitive dual-system theory. [arXiv preprint arXiv:2507.18178](#), 2025c.
- 732 Zhaohui Yang, Yuxiao Ye, Shilei Jiang, Chen Hu, Linjing Li, Shihong Deng, and Daxin Jiang.
733 Unearthing gems from stones: Policy optimization with negative sample augmentation for LLM
734 reasoning. [arXiv preprint arXiv:2505.14403](#), 2025d.
- 735 Erxin Yu, Jing Li, Ming Liao, Qi Zhu, Boyang Xue, Minghui Xu, Baojun Wang, Lanqing Hong,
736 Fei Mi, and Lifeng Shang. Self-error-instruct: Generalizing from errors for llms mathematical
737 reasoning. [arXiv preprint arXiv:2505.22591](#), 2025a.
- 738 Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian
739 Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng,
740 Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaye Chen,
741 Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing
742 Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang.
743 Dapo: An open-source llm reinforcement learning system at scale, 2025b. URL <https://arxiv.org/abs/2503.14476>.
- 744 Yige Yuan, Teng Xiao, Shuchang Tao, Xue Wang, Jinyang Gao, Bolin Ding, and Bingbing Xu.
745 Incentivizing reasoning from weak supervision. [arXiv preprint arXiv:2505.20072](#), 2025.
- 746 Chengshuai Zhao, Zhen Tan, Pingchuan Ma, Dawei Li, Bohan Jiang, Yancheng Wang, Yingzhen
747 Yang, and Huan Liu. Is chain-of-thought reasoning of llms a mirage? a data distribution lens.
748 [arXiv preprint arXiv:2508.01191](#), 2025.
- 749 Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang,
750 Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy op-
751 timization, 2025. URL <https://arxiv.org/abs/2507.18071>.

756 Xinyu Zhu, Mengzhou Xia, Zhepei Wei, Wei-Lin Chen, Danqi Chen, and Yu Meng. The surprising
757 effectiveness of negative reinforcement in LLM reasoning. [arXiv preprint arXiv:2506.01347](#),
758 2025.
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809

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 THEORETICAL DERIVATION

We provide a detailed derivation explaining why the dynamic reweighting mechanism in Eq. 6 improves optimization conditioning and, under standard assumptions, leads to improved generalization guarantees. The argument proceeds through a sequence of lemmas establishing: (i) the link between the loss-reduction statistic $\Delta_i^{(t)}$ and gradient alignment, (ii) the positive semi-definite (PSD) augmentation of the empirical Fisher induced by positive weight increments, (iii) a quantitative improvement of the spectrum of the Fisher in low-energy subspaces, and (iv) the transfer of improved conditioning to stability and generalization.

Throughout training, the underlying target objective remains the uniform empirical risk

$$R(\theta) = \frac{1}{N} \sum_{i=1}^N \ell_i(\theta).$$

However, the update direction at iteration t is the gradient of the reweighted surrogate objective

$$R_w^{(t)}(\theta) = \frac{1}{N} \sum_{i=1}^N w_i^{(t)} \ell_i^{(t)}(\theta),$$

whose weights $w_i^{(t)}$ are dynamically adjusted by the reweighting rule. Accordingly, all conditioning and curvature statements in this section refer to the local quadratic model of $R_w^{(t)}$ at iteration t , rather than to the fixed uniform objective $R(\theta)$. Our results therefore characterize how the reweighting mechanism reshapes the second-order geometry of the surrogate objective used at each step.

A.2.1 NOTATION AND STANDING ASSUMPTIONS

We keep the notation from the main text. At iteration t , we write $\theta^{(t)}$ for the current parameters, $g_i^{(t)} = \nabla_{\theta} \ell_i(\theta^{(t)})$ for the per-example gradients, and $w_i^{(t)}$ for the corresponding weights. For notational simplicity, we fix an iteration t and often drop the superscript (t) when it is clear from context. In particular, we write

$$g_i \triangleq g_i^{(t)} = \nabla_{\theta} \ell_i(\theta^{(t)}), \quad G \triangleq G^{(t)} = \frac{1}{N} \sum_i w_i g_i.$$

We use $H_i(\theta) = \nabla_{\theta}^2 \ell_i(\theta)$ for the per-example Hessians, and we denote by

$$R_w(\theta) \triangleq \frac{1}{N} \sum_{i=1}^N w_i \ell_i(\theta)$$

the reweighted surrogate objective at iteration t (with weights $\{w_i\}$ held fixed). Its Hessian is

$$H(\theta) \triangleq \nabla_{\theta}^2 R_w(\theta).$$

The empirical (weighted) Fisher at the same iteration is

$$F_w(\theta) \triangleq \frac{1}{N} \sum_{i=1}^N w_i g_i g_i^{\top},$$

and, since we work at a fixed iteration, we often abbreviate $F_w(\theta^{(t)})$ to F_w .

We now collect the assumptions used in the analysis.

Assumption A.1 (Smoothness, boundedness, curvature, and energy injection).

(A1) Each $\ell_i(\theta)$ is twice differentiable and L -smooth: $\|H_i(\theta)\|_{\text{op}} \leq L$.

(A2) Gradient norms are uniformly bounded: $\|g_i(\theta)\|_2 \leq G_{\max}$.

(A3) The learning rate η is small enough that higher-order terms are controlled.

(A4) (Fisher–Hessian closeness for R_w) In the local region of interest and for all iterates θ visited by the algorithm, the Hessian $H(\theta) = \nabla_{\theta}^2 R_w(\theta)$ of the reweighted objective and the corresponding empirical Fisher $F_w(\theta)$ satisfy

$$\|H(\theta) - F_w(\theta)\|_{\text{op}} \leq \delta.$$

(A5) (Energy injection of the reweighting rule) Let U be a k -dimensional low-curvature subspace with orthogonal projector P_U . At each step, let T denote the set of examples whose weights are increased, with nonnegative increments $\delta w_i \geq 0$ for $i \in T$. The induced change in the empirical Fisher is

$$\Delta F \triangleq \frac{1}{N} \sum_{i \in T} \delta w_i g_i g_i^{\top}.$$

We assume that the reweighting rule injects curvature uniformly into U in the sense that for every unit vector $v \in U$,

$$v^{\top} \Delta F v \geq \frac{\gamma}{k}.$$

Equivalently, the restriction of ΔF to U satisfies $\Delta F|_U \succeq (\gamma/k) P_U$, and in particular

$$\text{tr}(P_U \Delta F P_U) = \text{tr}(\Delta F|_U) \geq \gamma.$$

Lemma A.1 (Taylor relation between Δ_i and gradient alignment). Under assumptions (A1)–(A3), after one update $\theta \leftarrow \theta - \eta G$,

$$\Delta_i = \ell_i(\theta) - \ell_i(\theta - \eta G) = \eta g_i^{\top} G - \frac{1}{2} \eta^2 G^{\top} H_i(\xi_i) G,$$

for some ξ_i on the line segment between θ and $\theta - \eta G$. Moreover,

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

918 *Proof.* Second-order Taylor expansion yields the stated form, and L -smoothness gives the remainder
 919 bound. \square

920 **Lemma A.2** (Positive weight increments induce PSD augmentation). *Let the weights change by*
 921 *nonnegative increments $\delta w_i \geq 0$ for $i \in T$. The induced change in the empirical Fisher is*

$$922 \quad \Delta F = \frac{1}{N} \sum_{i \in T} \delta w_i g_i g_i^\top,$$

923 *which is PSD. Consequently, the updated Fisher $F'_w = F_w + \Delta F$ satisfies $F'_w \succeq F_w$, and, when the*
 924 *eigenvalues of both matrices are ordered in nondecreasing order, we have $\lambda_j(F'_w) \geq \lambda_j(F_w)$ for all*
 925 *j .*

926 *Proof.* Each outer product $g_i g_i^\top$ is symmetric and PSD. Since $\delta w_i \geq 0$, every term $\delta w_i g_i g_i^\top$ is PSD,
 927 and their average ΔF is PSD as well. Thus $F'_w = F_w + \Delta F$ is a PSD perturbation of the symmetric
 928 matrix F_w , so by Weyl's eigenvalue inequality we obtain $\lambda_j(F'_w) \geq \lambda_j(F_w)$ for all j when the
 929 eigenvalues are ordered in nondecreasing order. \square

930 To evaluate how reweighting affects curvature in directions where the objective is weakly curved,
 931 we consider a k -dimensional subspace U spanned by small-eigenvalue directions of the empirical
 932 Fisher F_w . Introducing such a subspace is standard in conditioning analysis, as the restricted spectrum
 933 $F_w|_U$ precisely characterizes curvature along these low-eigenvalue directions. Let P_U denote
 934 the orthogonal projector onto U . Intuitively, directions associated with large eigenvalues of F_w already
 935 exhibit sufficient curvature and are repeatedly explored by gradient-based updates. In contrast,
 936 the low-eigenvalue subspace U captures flat or poorly conditioned directions that act as the main
 937 bottleneck for optimization and conditioning. Our analysis therefore focuses on how reweighting
 938 increases curvature within U rather than on further amplifying already well-conditioned directions.

939 The effect of reweighting on second-order geometry is captured entirely by the increment

$$940 \quad \Delta F = \frac{1}{N} \sum_{i \in T} \delta w_i g_i g_i^\top,$$

941 which is positive semi-definite by construction: each $g_i g_i^\top$ is PSD and each weight increment δw_i
 942 arising from the reweighting rule equation 6 is nonnegative. Hence the updated Fisher satisfies
 943 $F'_w = F_w + \Delta F \succeq F_w$, providing a monotone PSD augmentation that allows the application of
 944 standard Weyl-type eigenvalue inequalities.

945 To guarantee that this PSD increment has a meaningful effect on the low-curvature subspace U , we
 946 impose the uniform energy condition in (A5): there exists a constant $\gamma > 0$ such that, for every unit
 947 vector $v \in U$ (where U is k -dimensional with orthogonal projector P_U),

$$948 \quad v^\top \Delta F v \geq \frac{\gamma}{k}.$$

949 Equivalently, the restriction $\Delta F|_U$ satisfies $\Delta F|_U \succeq (\gamma/k) P_U$, so that $\text{tr}(P_U \Delta F P_U) =$
 950 $\text{tr}(\Delta F|_U) \geq \gamma$ as a simple corollary. Intuitively, this condition rules out the degenerate case in
 951 which all of the additional mass is concentrated on a few directions inside U ; instead, it enforces a
 952 uniform strengthening of curvature across the low-eigenvalue subspace. We now state the resulting
 953 spectral improvement.

954 **Lemma A.3** (Improvement of small-eigenvalue subspace). *Let U be a k -dimensional subspace with*
 955 *projector P_U . Suppose the weight increments satisfy the energy condition (A5), so that $\Delta F|_U \succeq$*
 956 *$(\gamma/k) P_U$ for some $\gamma > 0$. Let $\lambda_{\min}(F_w|_U)$ denote the minimal eigenvalue of F_w restricted to U .*
 957 *Then the minimal eigenvalue satisfies*

$$958 \quad \lambda_{\min}(F'_w|_U) \geq \lambda_{\min}(F_w|_U) + \frac{\gamma}{k}.$$

959 *Proof.* By (A5) we have $\Delta F|_U \succeq (\gamma/k) P_U$, which implies $\lambda_{\min}(\Delta F|_U) \geq \gamma/k$. Since $F'_w|_U =$
 960 $F_w|_U + \Delta F|_U$ and both are symmetric, the eigenvalue monotonicity for sums of Hermitian matrices
 961 yields

$$962 \quad \lambda_{\min}(F'_w|_U) \geq \lambda_{\min}(F_w|_U) + \lambda_{\min}(\Delta F|_U) \geq \lambda_{\min}(F_w|_U) + \frac{\gamma}{k},$$

963 as claimed. \square

972 *Remark A.4.* The uniform energy condition in (A5) ensures that the augmented weights inject non-
 973 trivial curvature in every direction of U , not just along a few isolated eigenvectors. This rules out
 974 pathological cases where the trace increases but the smallest eigenvalue remains nearly unchanged,
 975 and guarantees a genuine improvement of the worst-case curvature on U .

976 **Lemma A.5** (Transfer from Fisher to Hessian). *Under (A4), if*

$$977 \lambda_{\min}(F'_w|_U) - \lambda_{\min}(F_w|_U) \geq \Delta\lambda_F,$$

978 *then the local Hessian of the reweighted objective satisfies*

$$979 \lambda_{\min}(H'|_U) \geq \lambda_{\min}(H|_U) + \Delta\lambda_F - 2\delta,$$

980 *where H and H' denote the Hessian $H(\theta) = \nabla_{\theta}^2 R_w(\theta)$ evaluated at the current and updated
 981 parameters, respectively.*

982 *Proof.* Assumption (A4) yields $\|H - F_w\|_{\text{op}} \leq \delta$ and $\|H' - F'_w\|_{\text{op}} \leq \delta$ at the current and updated
 983 iterates. These bounds imply matching eigenvalue relations before and after the update, giving the
 984 stated inequality. \square

985 **Lemma A.6** (Improved conditioning reduces parameter sensitivity). *Assume a restricted strong
 986 convexity condition on U : there exists $\mu > 0$ such that, throughout the local region,*

$$987 \lambda_{\min}(H|_U) \geq \mu.$$

988 *Consider two training sets that differ by a single example and run identical reweighted updates. Un-
 989 der standard Lipschitz assumptions on the gradients, the resulting parameter perturbation between
 990 the two runs is $O(1/\mu)$. Hence increasing μ —equivalently improving the smallest eigenvalue of H
 991 on U —reduces algorithmic instability and yields a smaller generalization gap.*

992 *Proof sketch.* Restricted strong convexity with parameter μ implies that the reweighted surrogate
 993 objective R_w is μ -strongly convex along directions in U . In particular, the map that sends the
 994 empirical risk (or its gradient) to its minimizer is $1/\mu$ -Lipschitz along U : if two datasets differ by
 995 one example, the corresponding empirical gradients differ by at most a constant L_g , and the resulting
 996 parameters θ and θ' satisfy

$$997 \|\theta' - \theta\| \leq \frac{L_g}{\mu}.$$

998 This $O(1/\mu)$ sensitivity of the iterates yields uniform stability, in the sense that the loss on any test
 999 point differs by at most $O(1/\mu)$ between the two runs. The uniform stability framework of Bousquet
 1000 & Elisseeff (2002) and the refinement in Hardt et al. (2016) then imply that such a stability bound
 1001 translates into an $O(1/\mu)$ upper bound on the generalization error. We refer to these works for
 1002 complete statements and proofs. \square

1003 **Proposition A.7** (Main result: conditioning and generalization improvement). *Under (A1)–(A5),
 1004 suppose the weight update rule equation 6 produces nonnegative increments satisfying the energy
 1005 condition (A5). Then:*

- 1006 1. *The empirical Fisher receives a PSD augmentation ΔF and, on the low-curvature subspace*
 1007 *U , both its average eigenvalue and its minimal eigenvalue increase by at least γ/k :*

$$1008 \frac{1}{k} \text{tr}(P_U F'_w P_U) \geq \frac{1}{k} \text{tr}(P_U F_w P_U) + \frac{\gamma}{k}, \quad \lambda_{\min}(F'_w|_U) \geq \lambda_{\min}(F_w|_U) + \frac{\gamma}{k}.$$

1009 *The first inequality follows from the trace identity and (A5), while the second is the content
 1010 of Lemma A.3.*

- 1011 2. *By Lemma A.5, taking $\Delta\lambda_F = \gamma/k$ as provided by Lemma A.3, the minimal Hessian
 1012 eigenvalue of the reweighted objective on U satisfies*

$$1013 \lambda_{\min}(H'|_U) \geq \lambda_{\min}(H|_U) + \frac{\gamma}{k} - 2\delta.$$

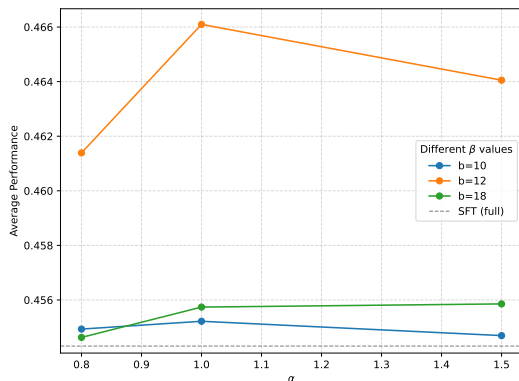


Figure 5: Ablation study on the hyperparameters α and β . GLOW exhibits stable performance across different settings, demonstrating the robustness of the reweighting formulation.

A.6 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.7 CASE STUDY OF NEGATIVE SAMPLES

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.8 TRAINING LOSS ON OPENMATHREASONING AND MMLU

We present in Figure 9 the loss comparison of all models trained under the positive and negative settings on the OpenMathReasoning and MMLU datasets.

A.9 PROGRESS LOSS

Table 9 reports intermediate checkpoint evaluations for Qwen2.5-7B and Qwen2.5-32B trained on the math reasoning and general reasoning datasets. For each setting, we compare SFT using positive distilled reasoning trajectories against SFT using negative distilled reasoning trajectories at 5, 10, 15, and 20 epochs. Across model sizes and training sets, negative-trajectory SFT consistently improves over the base model and shows gains that are comparable to the positive-trajectory counterpart. In several configurations, negative-trajectory SFT matches or exceeds positive-trajectory SFT on out-of-domain benchmarks. These results suggest that negative trajectories contain structured supervision signal rather than noise.

A.10 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.

1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187

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

1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241

An 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:**

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 .

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 .

Figure 7: Case study of thinking trajectories for M_{pos} and M_{neg} on the same question.

1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295

Problem

The graph shows the total distance Sam drove from 6 a.m. to 11 a.m. How many miles per hour is the car's average speed for the period from 6 a.m. to 11 a.m.?

```
unitsize(0.2inch); draw((0,0)-(5.5,0)); draw((0,0)-(0,8.5)); draw((1,0)-(1,8.5)); draw((2,0)-(2,8.5));
draw((3,0)-(3,8.5)); draw((4,0)-(4,8.5)); draw((5,0)-(5,8.5)); draw((0,1)-(5.5,1)); draw((0,8)-(5.5,8));
draw((0,7)-(5.5,7)); draw((0,6)-(5.5,6)); draw((0,5)-(5.5,5)); draw((0,4)-(5.5,4));
draw((0,3)-(5.5,3)); draw((0,2)-(5.5,2)); draw((0,0)-(1,2)-(2,3)-(3,5)-(4,6)-(5,8)); dot((0,0));
dot((1,2)); dot((2,3)); dot((3,5)); dot((4,6)); dot((5,8)); label("6",(0,-0.5),S); label("7",(1,-0.5),S);
label("8",(2,-0.5),S); label("9",(3,-0.5),S); label("10",(4,-0.5),S); label("11",(5,-0.5),S); label("0",(-0.5,0),W);
label("40",(-0.5,2),W); label("80",(-0.5,4),W); label("120",(-0.5,6),W); label("160",(-0.5,8),W); label("Time of Day (a.m.)", (2.7,-2),S); label("Total distance",(-0.5,9),N);
```

Model's Thinking Trajectory and Answer

Let me start with the total time. The period is from 6 a.m. to 11 a.m. Let me count the hours. From 6 to 7 is 1 hour, 7 to 8 is another, 8 to 9, 9 to 10, and 10 to 11. That's 5 hours total. Wait, is that right? Let me check again. From 6 to 11 is 5 hours? Wait, 11 minus 6 is 5, yes. So total time is 5 hours. Okay, that seems straightforward.

Now, the total distance. The graph shows the total distance driven at different times. The Asymptote code draws a graph with time on the x-axis and total distance on the y-axis. The points given are at time 6 a.m. (which is the first point, $x = 0$), then at 7 a.m. ($x = 1$), 8 a.m. ($x = 2$), etc., up to 11 a.m. ($x = 5$). The corresponding distances are given as y-values. Let me check the Asymptote code again to see the coordinates.

Looking at the Asymptote code, the graph is drawn from $(0, 0)$ to $(1, 2)$ to $(2, 3)$ to $(3, 5)$ to $(4, 6)$ to $(5, 8)$. The labels on the y-axis are 0, 40, 80, 120, 160. So each unit on the y-axis corresponds to 40 miles? Let me confirm. The labels are at $y = 0, 2, 4, 6, 8$. The labels next to them are 0, 40, 80, 120, 160. So, each vertical unit is 40 miles. Therefore, the y-coordinate multiplied by 40 gives the total distance in miles.

So, for example:

- At 6 a.m. ($x = 0$), the total distance is 0 miles.
- At 7 a.m. ($x = 1$), the total distance is 2 units, which is $2 \times 40 = 80$ miles.
- At 8 a.m. ($x = 2$), it's 3 units, so $3 \times 40 = 120$ miles.
- At 9 a.m. ($x = 3$), 5 units, which is $5 \times 40 = 200$ miles.
- At 10 a.m. ($x = 4$), 6 units, which is $6 \times 40 = 240$ miles.
- At 11 a.m. ($x = 5$), 8 units, which is $8 \times 40 = 320$ miles.

The question is about the total distance from 6 a.m. to 11 a.m. The graph is showing cumulative distance. So at 11 a.m., the total distance is 320 miles. Therefore, the total distance driven is 320 miles.

If the graph is cumulative, then the total distance at 11 a.m. is the total distance driven from 6 a.m. to 11 a.m., so that would be 320 miles. Therefore, total distance is 320 miles, total time is 5 hours. Therefore, average speed is $320 \div 5 = 64$ miles per hour.

Let me check if the total distance is indeed 320. The last point is at 11 a.m., which is $x = 5, y = 8$. Since each y unit is 40 miles, $8 \times 40 = 320$. Yes. So that seems correct.

Alternatively, maybe the question is trying to trick me? Suppose someone might think that the total distance is the sum of all the segments, but because the graph is cumulative, the total distance at 11 a.m. is already the total distance driven from 6 a.m. to 11 a.m.

We can also confirm by looking at increments: From 6–7: 80 miles, From 7–8: $120 - 80 = 40$ miles. From 8–9: $200 - 120 = 80$ miles. From 9–10: $240 - 200 = 40$ miles. From 10–11: $320 - 240 = 80$ miles. Summing gives $80 + 40 + 80 + 40 + 80 = 320$ miles, consistent with the final value.

So average speed is total distance divided by total time. Total time is 5 hours. $320 \div 5 = 64$, so 64 miles per hour.

The time from 6 a.m. to 11 a.m. is indeed 5 hours. The x-axis is labeled from 6 a.m. to 11 a.m., with each unit representing one hour, so that checks out.

Therefore, I think my answer is correct.

Final Answer: the car's average speed is miles per hour.

Figure 8: Case study of a negative trajectory from the OpenMathReasoning training dataset. The model misinterprets the problem, but its subsequent step-by-step reasoning and formula derivations remain structurally correct.

1296
 1297
 1298
 1299
 1300
 1301
 1302
 1303
 1304
 1305
 1306
 1307
 1308
 1309
 1310
 1311
 1312
 1313
 1314
 1315
 1316
 1317
 1318
 1319
 1320
 1321
 1322
 1323
 1324
 1325
 1326
 1327
 1328
 1329
 1330
 1331
 1332
 1333
 1334
 1335
 1336
 1337
 1338
 1339
 1340
 1341
 1342
 1343
 1344
 1345
 1346
 1347
 1348
 1349

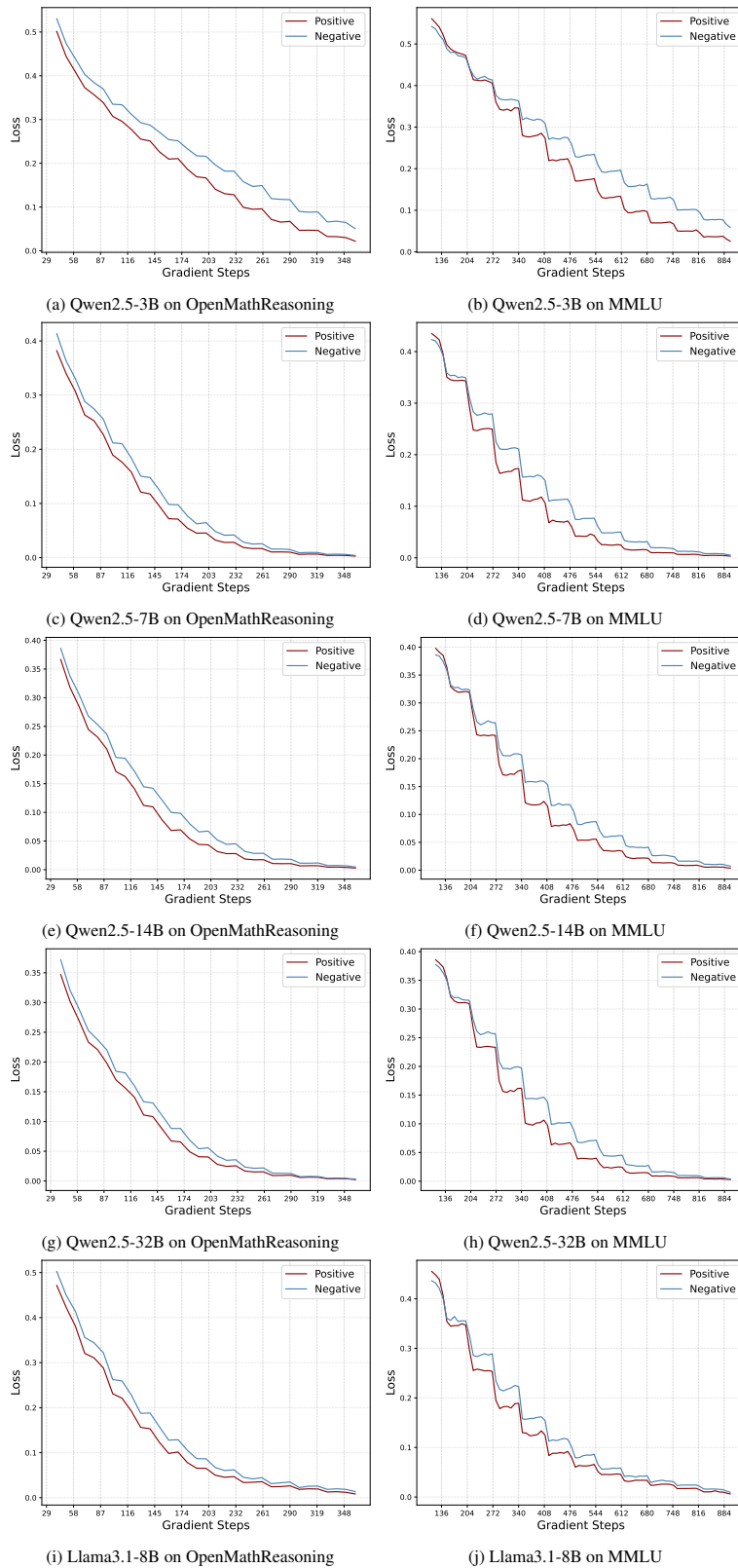


Figure 9: Training loss of Qwen2.5 models and Llama3.1-8B on OpenMathReasoning (left) and MMLU (right). Losses drop across epochs, with the positive setting converging faster than the negative.

Table 9: **Checkpoint evaluation across SFT epochs with distilled reasoning trajectories.** We report performance at 5, 10, 15, and 20 epochs. Each row corresponds to a model size and training dataset, and each row contains two subtables that compare training on positive (left) versus negative (right) distilled trajectories. Columns in each subtable correspond to benchmarks, while rows correspond to training epochs, with Base denoting the model before SFT.

(a) Qwen2.5-7B is fine-tuned on the **math reasoning** dataset using **positive** distilled trajectories.

Epoch	Math500	Minerva	Olympia	AMC	MMLU	MMLU-Pro	BBH
Base	58.40	26.84	26.07	52.50	55.80	26.56	51.10
5epoch	72.80	37.13	37.19	45.00	60.95	30.34	54.69
10epoch	75.80	38.24	40.59	65.00	64.06	32.50	61.62
15epoch	77.20	36.76	41.93	55.00	60.81	32.15	59.69
20epoch	78.00	36.76	41.78	57.50	61.03	32.70	60.58

(b) Qwen2.5-7B is fine-tuned on the **math reasoning** dataset using **negative** distilled trajectories.

Epoch	Math500	Minerva	Olympia	AMC	MMLU	MMLU-Pro	BBH
Base	58.40	26.84	26.07	52.50	55.80	26.56	51.10
5epoch	71.20	31.99	31.56	47.50	62.58	44.04	56.28
10epoch	77.20	34.93	39.26	50.00	71.39	52.14	69.49
15epoch	78.60	39.71	38.37	52.50	72.10	52.24	71.09
20epoch	77.60	40.44	38.37	57.50	73.11	53.74	71.73

(c) Qwen2.5-7B is fine-tuned on the **general reasoning** dataset using **positive** distilled trajectories.

Epoch	Math500	Minerva	Olympia	AMC	MMLU	MMLU-Pro	BBH
Base	58.40	26.84	26.07	52.50	55.80	26.56	51.10
5epoch	72.00	36.76	37.33	47.50	73.62	50.61	64.05
10epoch	74.60	37.50	41.48	55.00	73.79	53.32	69.73
15epoch	72.00	37.50	39.26	50.00	74.11	53.91	68.34
20epoch	74.40	37.50	39.85	50.00	73.42	53.22	68.23

(d) Qwen2.5-7B is fine-tuned on the **general reasoning** dataset using **negative** distilled trajectories.

Epoch	Math500	Minerva	Olympia	AMC	MMLU	MMLU-Pro	BBH
Base	58.40	26.84	26.07	52.50	55.80	26.56	51.10
5epoch	76.80	36.76	37.78	47.50	71.09	43.99	66.00
10epoch	76.80	37.87	40.30	52.50	71.43	45.87	68.84
15epoch	76.80	37.13	41.48	55.00	71.30	44.62	69.30
20epoch	77.00	37.13	42.07	60.00	71.23	45.79	69.46

(e) Qwen2.5-32B is fine-tuned on the **math reasoning** dataset using **positive** distilled trajectories.

Epoch	Math500	Minerva	Olympia	AMC	MMLU	MMLU-Pro	BBH
Base	63.20	34.19	26.52	35.00	68.34	39.80	58.65
5epoch	90.20	49.63	59.11	85.00	76.53	46.77	78.04
10epoch	92.60	50.00	60.44	85.00	78.63	51.67	79.01
15epoch	93.00	48.53	62.07	90.00	78.72	51.99	80.57
20epoch	91.40	50.74	60.89	85.00	79.01	54.31	80.61

(f) Qwen2.5-32B is fine-tuned on the **math reasoning** dataset using **negative** distilled trajectories.

Epoch	Math500	Minerva	Olympia	AMC	MMLU	MMLU-Pro	BBH
Base	63.20	34.19	26.52	35.00	68.34	39.80	58.65
5epoch	88.40	45.22	52.30	85.00	83.07	68.23	83.55
10epoch	92.20	51.10	57.93	85.00	85.14	73.75	84.22
15epoch	91.20	50.74	57.33	90.00	85.02	73.48	84.62
20epoch	92.20	50.74	58.37	95.00	85.47	73.53	84.51

(g) Qwen2.5-32B is fine-tuned on the **general reasoning** dataset using **positive** distilled trajectories.

Epoch	Math500	Minerva	Olympia	AMC	MMLU	MMLU-Pro	BBH
Base	63.20	34.19	26.52	35.00	68.34	39.80	58.65
5epoch	84.60	44.85	52.00	62.50	82.10	66.54	80.03
10epoch	86.60	46.69	55.70	75.00	81.14	67.01	80.69
15epoch	85.00	47.06	56.59	75.00	81.73	68.33	81.73
20epoch	85.20	46.69	56.15	75.00	81.97	68.54	81.60

(h) Qwen2.5-32B is fine-tuned on the **math reasoning** dataset using **negative** distilled trajectories.

Epoch	Math500	Minerva	Olympia	AMC	MMLU	MMLU-Pro	BBH
Base	63.20	34.19	26.52	35.00	68.34	39.80	58.65
5epoch	85.00	44.49	51.26	77.50	78.74	57.48	79.09
10epoch	87.20	46.30	54.52	75.00	79.01	60.43	80.88
15epoch	86.40	47.79	55.70	65.00	77.77	57.14	79.97
20epoch	86.40	47.06	56.89	72.50	77.99	58.34	80.71

A.11 LIMITATION

Our study primarily examines gain-based reweighting in the supervised fine-tuning stage of reasoning post-training, and we leave its interaction with subsequent RLHF or other reinforcement learning stages as an exciting direction for future work. In addition, our experiments focus on text-only chain-of-thought data for math and multi-task knowledge benchmarks with a small set of open-source backbones, so a natural next step is to extend the same analysis and method to broader task families, larger model scales and multimodal or tool-augmented settings, building on the phenomena and gains established in this work.