GIFT: Guided Fine-Tuning and Transfer for Enhancing Instruction-Tuned Language Models

Zhiwen Ruan[1], **Yichao Du**[2], **Jianjie Zheng**[1], **Longyue Wang**[2], **Yun Chen**[3]
Peng Li[4], **Jinsong Su**[5], **Yang Liu**[4], **Guanhua Chen**[1*]
[1]Southern University of Science and Technology, [2]Alibaba Group
[3]Shanghai University of Finance and Economics, [4]Tsinghua University, [5]Xiamen University

Abstract

A promising paradigm for adapting instruction-tuned language models is to learn task-specific updates on a pretrained base model and subsequently merge them into the instruction-tuned model. However, existing approaches typically treat the instruction-tuned model as a passive target that is only involved at the final merging stage, without guiding the training process. We propose **GIFT** (**Gui**ded **F**ine-Tuning and **T**ransfer), a simple and efficient framework that incorporates guidance from the instruction model into task adaptation. GIFT fine-tunes a low-rank adapter on the pretrained base model using confidence signals derived from the instruction-tuned model. The learned adapter is then merged into the instruction-tuned model, yielding task-specialized models that preserve general instruction-following behavior. We evaluate GIFT on mathematical and knowledge-intensive benchmarks across multiple model families and scales. Results show that GIFT consistently outperforms direct fine-tuning and representative transfer-based baselines, while maintaining robust generalization and favorable test-time scaling behavior.

1 Introduction

Instruction-tuned open-source large language models (LLMs), such as Qwen and Llama, have demonstrated strong instruction-following and zero-shot generalization capabilities, driven by the scaling of model parameters and training data (AI@Meta, 2024; Team, 2024). These models are produced through expensive and carefully designed post-training pipelines, resulting in parameters that are finely balanced across general reasoning ability (Ruan et al., 2025a; Yang et al., 2026; Wang et al., 2026), instruction-following behavior, and robustness (Brown et al., 2020; Fedus et al., 2022; Achiam et al., 2023; Wang et al., 2025). Consequently, adapting instruction-tuned models to specific downstream tasks remains challenging (Ruan et al., 2025d), as naive fine-tuning can easily disrupt this balance and lead to instability or performance degradation (Wu et al., 2025a).

A common alternative is to perform supervised fine-tuning on the pretrained base model and then transfer the learned task-specific updates back to the instruction-tuned model (Fleshman and Durme, 2024; Lin et al., 2025; Cao et al., 2025). Representative methods such as Shadow-FT (Wu et al., 2025a) and Chat Vector (Huang et al., 2024) follow this paradigm by training adapters on the base model and merging them into the instruction model, while Re-Adapt and its variants further explore linear combinations of base parameters, instruction offsets, and task-adapted updates (Fleshman and Durme, 2024). Despite their effectiveness in certain settings, these approaches treat the instruction-tuned model as a passive target that only participates at the final merging stage, without influencing how task-specific knowledge is learned.

In this work, we argue that the instruction-tuned model can play a more active role in task adaptation by providing guidance during the learning process itself, rather than only participating in post-hoc merging. Based on this insight, we propose **GIFT** (**Gui**ded **F**ine-Tuning and **T**ransfer), a simple and efficient framework that incorporates signals from the instruction model into task adaptation. Specifically, GIFT fine-tunes a low-rank adapter on the pretrained base model using precomputed confidence signals from the instruction model. This guidance mechanism effectively redistributes the learning signal by suppressing updates from uncertain predictions and focusing optimization on regions consistent with the instruction model's alignment. Finally, the learned adapter is merged into the instruction-tuned model, yielding a task-specialized model that inherits new capabilities while maintaining robust merge compatibility.

We evaluate GIFT on a range of mathematical

* Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 7866–7878

and knowledge-intensive tasks, including mathematics and medical QA, across multiple model families and scales. Empirical results demonstrate that GIFT consistently outperforms direct fine-tuning and existing merge-based baselines, while maintaining generalization and instruction-following capabilities. On mathematical benchmarks (Math500, Minerva Math, OlympiadBench, AIME 2024, and AMC 2023), GIFT improves accuracy by an average of 5.2% over Llama3.1-8B-Instruct (AI@Meta, 2024), and on medical QA tasks (Pal et al., 2022; Jin et al., 2021), it achieves a 6.2% gain. Overall, GIFT provides a principled and practical framework for adapting instruction-tuned language models, demonstrating that task-specific capabilities can be effectively acquired through guided fine-tuning on the base model and transferred back to the instruction model. [1]

2 Related Work

2.1 Model Arithmetic

Foundational studies establish that the parameter space of neural networks supports arithmetic manipulation (Izmailov et al., 2019; Wortsman et al., 2022), a property formalized by Task Vectors (Ilharco et al., 2023) to merge or unlearn capabilities via vector addition. To mitigate performance degradation arising from parameter interference, recent techniques employ sparsification strategies. TIES-Merging (Yadav et al., 2023) eliminates redundancy by pruning low-magnitude updates and resolving sign conflicts, while DARE (Yu et al., 2024) stochastically drops redundant delta parameters followed by rescaling. It is important to note that these merging techniques address a fundamentally different problem from GIFT. TIES and DARE operate in a purely post-hoc manner, assuming task-specific updates are already learned and focusing on how to combine them with minimal interference. In contrast, GIFT targets the learning stage itself by shaping optimization on the base model using guidance from the instruction model. As a result, these methods are complementary rather than competing, and can be directly integrated into the GIFT pipeline if desired.

2.2 Enhanced Instruct Model

Recent research exploits the linear arithmetic property of model weights to enhance capabilities. REAdapt (Fleshman and Durme, 2024) and Chat Vector (Huang et al., 2024) pioneered this by extracting alignment vectors—defined as the difference between instruct and base weights—and grafting them onto domain-adapted or multilingual backbones to transfer instruction-following skills. Extending this to model evolution, ParamΔ(Cao et al., 2025) and Fine-tuning Transfer(Lin et al., 2025) demonstrate that these "diff vectors" can be propagated across model versions (e.g., Llama 3 to 3.1) to instantly replicate capabilities. Most recently, Shadow-FT (Wu et al., 2025a) addresses the instability of direct instruction tuning by learning updates on the base model and transferring them to the instruct version, preserving alignment while incorporating task knowledge.

While effective, these approaches treat the instruction model merely as a passive recipient for weight merging. In contrast, GIFT uses the instruction model to provide offline token-level confidence signals during base-model adaptation. By using these signals to weigh the base model's loss, GIFT produces adapters that are more compatible with the instruction model upon merging.

3 Method

3.1 Setup and Objective

Let $\mathcal{D} = \{(x, y)\}$ denote a supervised dataset. We denote the pretrained base model as $p_{\theta_B}(y \mid x)$ and the instruction-tuned model as $p_{\theta_I}(y \mid x)$. Task adaptation is performed by training a low-rank adapter ϕ on the base model.

Prior work, such as Shadow-FT (Wu et al., 2025a) and Chat Vector (Huang et al., 2024), suggests that updates can be transferred from the base model to the instruction model, as they share the same backbone and lie in a nearby region of parameter space:

$$\theta_I = \theta_B + \Delta_I. \tag{1}$$

Leveraging this property, task-specific updates learned on the base model (denoted $M(\phi)$) can be directly applied to the instruction model:

$$\theta'_I = \theta_I + M(\phi), \tag{2}$$

where $M(\cdot)$ represents the merge operator (e.g., LoRA merging).

However, existing approaches typically treat the instruction model θ_I as a passive target, only consulted at the final merging stage. As a result, the adapter ϕ is trained using standard objectives that

[1] Our code is publicly available at https://github.com/sustech-nlp/gift.

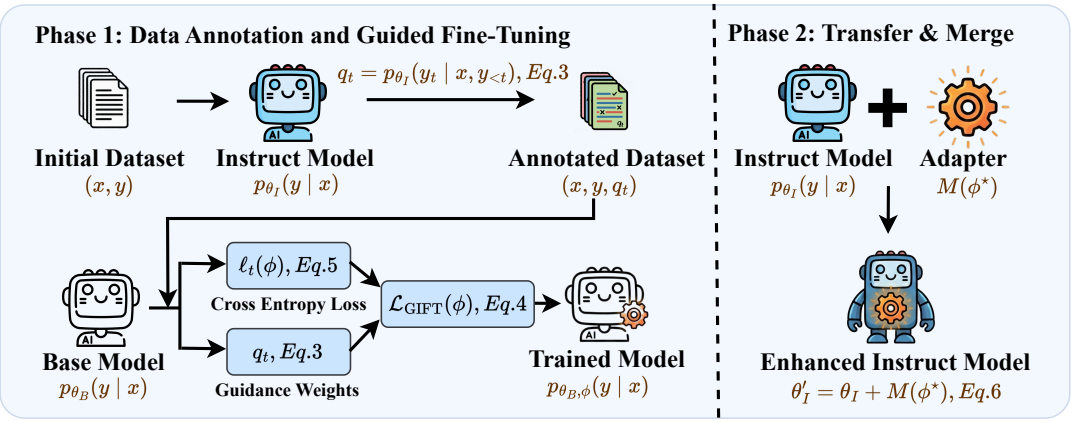

Figure 1: Overview of GIFT. **Phase 1: Data Annotation and Guided Fine-Tuning.** The instruction-tuned model is first used to annotate the initial training dataset with confidence scores, producing an annotated dataset. The base model is then fine-tuned with a low-rank adapter using these confidence-guided supervision signals. **Phase 2: Transfer and Merge.** After training, the learned adapter is merged into the instruction-tuned model, yielding a task-enhanced instruction model without directly fine-tuning it.

treat all tokens equally, overlooking the useful confidence information embedded in θ_I. We instead use supervision signals from θ_I to measure token importance during base-model adaptation, allowing us to distinguish between valuable task knowledge and potential noise. Our objective is to leverage this guidance to learn a higher-quality adapter ϕ, which, when merged via Eq. 2, produces a more effective adapted model θ'_I.

3.2 Guided Fine-Tuning

To train the adapter ϕ within the merge-based framework described above, prior methods typically rely on standard supervised fine-tuning (SFT) (Wu et al., 2025a; Huang et al., 2024). This conventional approach optimizes the negative log-likelihood on the base model:

$$\mathcal{L}_{\text{SFT}} = \mathbb{E}_{(x,y)\sim\mathcal{D}}\Big[\sum_{t=1}^{T} -\log p_{\theta_B,\phi}(y_t \mid x, y_{<t})\Big],$$

which implicitly treats all target tokens as equally important for task adaptation. However, natural language generation admits substantial flexibility. As observed in prior work (Ruan et al., 2025c), many tokens in reference responses are substitutable or stylistic, contributing little to core task reasoning. Forcing the model to equally fit such tokens, especially when they are weakly supported by the instruction model, can introduce unnecessary noise and degrade stability after merging.

To address this issue, GIFT explicitly models token importance using guidance from the instruction-tuned model. For each training example (x, y), we perform a single forward pass of the instruction model to compute

$$q_t = p_{\theta_I}(y_t \mid x, y_{<t}), \quad t = 1, \ldots, T, \tag{3}$$

where T denotes the target sequence length.

We interpret q_t as an importance score. While prior approaches estimate data quality using a model's own likelihood (Wu et al., 2025b; Zhu et al., 2026), the instruction-tuned model θ_I exhibits stronger alignment and task competence after post-training. As a result, confidence signals derived from θ_I provide a more reliable calibration of which tokens reflect essential task knowledge (See Section 4.3).

Since the dataset and instruction model are fixed, these importance scores are computed offline once per dataset, yielding augmented training tuples $(x, y, \mathbf{q})$, where $\mathbf{q} = [q_1, \ldots, q_T]$ is the sequence of confidence scores corresponding to each token in response y. During fine-tuning, we incorporate token importance into the optimization objective:

$$\mathcal{L}_{\text{GIFT}}(\phi) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\Big[\sum_{t=1}^{T} q_t\, \ell_t(\phi)\Big], \tag{4}$$

$$\ell_t(\phi) = -\log p_{\theta_B,\phi}(y_t \mid x, y_{<t}), \tag{5}$$

where q_t acts as an importance weight. In all experiments, we use q_t directly as computed in Eq. (3), without additional normalization, clipping, or temperature scaling. Tokens assigned low confidence by the instruction model contribute minimally to

the gradient, while high-confidence tokens dominate the optimization signal. This mechanism concentrates learning on instruction-consistent and task-critical tokens, guiding the adapter toward instruction-aligned updates.

Unlike knowledge distillation, GIFT does not minimize a divergence between teacher and student distributions. The instruction model is queried only once to produce scalar importance scores, which are then used to reweight the standard cross-entropy loss during base-model adapter training. This design preserves the simplicity and efficiency of supervised fine-tuning while injecting guidance from the instruction model into the adaptation process.

3.3 Transfer via Adapter Merge

After optimization, the learned adapter $\phi^\star$ is merged into the instruction model:

$$\theta_I' = \theta_I + M(\phi^\star). \tag{6}$$

Unlike post-hoc merging methods, the adapter learned by GIFT has been shaped throughout training by guidance. As a result, the merged model inherits task-specific improvements while preserving the original instruction-following behavior and robustness.

We adopt standard LoRA merging in our experiments to minimize confounding factors and isolate the effect of guided fine-tuning. More generally, **GIFT is formulated as a unified pipeline for adapting instruction-tuned models, rather than a standalone training objective or a specialized merging procedure**. Its main contribution lies in integrating instruction-aware learning on the base model with a transfer mechanism that injects the resulting task-specific updates into the instruction-tuned model.

4 Experiments

4.1 Experimental Setup

We evaluate **GIFT** on both mathematical reasoning and knowledge-intensive question answering tasks to study whether guided adapter transfer can improve task performance under limited supervision. To ensure fair comparison, all methods use identical LoRA hyperparameters and are evaluated under the same decoding and evaluation protocols. Additional results on newer model families are provided in Appendix B.

Models. We consider both pretrained base models and their corresponding instruction-tuned variants. For mathematical reasoning, experiments are conducted on four model families: Llama3.1-8B and Llama3.2-3B (AI@Meta, 2024), Qwen2.5-7B (Team, 2024), and DeepSeek-Math-7B (Shao et al., 2024). For knowledge-intensive evaluation, we follow prior work (Zhu et al., 2026) and focus on the Llama3.1-8B family to study domain adaptation.

Datasets. We evaluate **GIFT** on two representative domains: mathematical reasoning and knowledge-intensive question answering. For mathematical reasoning, we fine-tune models on 2,000 samples from NuminaMath-CoT (LI et al., 2024), and evaluate performance on a diverse set of benchmarks, including Math500 (Hendrycks et al., 2021), Minerva Math (Lewkowycz et al., 2022), OlympiadBench (He et al., 2024), AIME 2024 (American Institute of Mathematics, 2024), and AMC 2023 (Mathematical Association of America, 2023). These benchmarks span varying difficulty levels and reasoning styles, providing a comprehensive assessment of mathematical generalization under limited supervision. For knowledge-intensive evaluation, we fine-tune models on 10,000 samples from MedMCQA (Pal et al., 2022). Evaluation is conducted on MMLU-medical (Hendrycks et al., 2020), MedQA (Jin et al., 2021), and the MedMCQA test set, which collectively measure factual accuracy, domain knowledge coverage, and multiple-choice reasoning in the medical domain.

Baselines. We compare GIFT with several representative adaptation and transfer baselines that differ in where task-specific updates are learned and how they are merged into the instruction-tuned model.

- **Instruct Model**. The original instruction-tuned checkpoint released by the model developers serves as both a strong baseline and the merge target for all transfer-based methods.
- **Instruct-SFT**. Direct supervised fine-tuning of the instruction-tuned model.
- **Shadow-FT / Chat Vector** (Huang et al., 2024; Wu et al., 2025a). Both methods fine-tune the base model and then directly merge the learned weight updates or adapters into the instruction-tuned model.
- **Re-Adapt / Task Arithmetic** (Fleshman and Durme, 2024; Ilharco et al., 2023). Linear combi-

Methods	Math-500	Minerva Math	Olympiad Bench	AIME24	AMC23	Average
Llama3.1-8B-Instruct	37.4	16.8	9.8	1.9	18.0	16.8
w. SFT	23.0	7.7	4.9	0.6	8.3	8.9
w. Shadow-FT	40.1	18.0	11.3	1.9	18.6	18.0
w. Re-Adapt	26.7	9.3	5.5	0.4	9.7	10.3
w. LoRE-Adapt	20.0	6.2	4.1	0.2	7.2	7.5
w. **GIFT (Ours)**	**45.8**	**19.9**	**15.9**	**5.0**	**23.3**	**22.0**
Llama3.2-3B-Instruct	38.2	12.4	9.5	3.5	18.8	16.5
w. SFT	22.4	4.9	4.3	0.6	10.0	8.4
w. Shadow-FT	37.6	12.1	10.7	4.2	18.9	16.7
w. Re-Adapt	19.0	4.1	3.9	0.4	9.7	7.4
w. LoRE-Adapt	9.6	2.4	2.1	0.2	3.9	3.6
w. **GIFT (Ours)**	**42.5**	**14.3**	**12.7**	**5.4**	**22.3**	**19.5**
Qwen2.5-7B-Instruct	73.2	38.2	35.4	11.2	48.6	41.3
w. SFT	48.3	17.2	16.5	1.7	22.3	21.2
w. Shadow-FT	70.6	36.6	31.3	7.9	43.4	38.0
w. Re-Adapt	68.9	33.4	30.1	10.4	44.5	37.5
w. LoRE-Adapt	66.1	27.3	29.0	7.7	41.1	34.2
w. **GIFT (Ours)**	**75.0**	**38.4**	**36.1**	**12.7**	**52.5**	**42.9**
DeepSeek-Math-7B-Instruct	39.0	17.6	10.9	0.8	15.6	16.8
w. SFT	30.5	11.4	6.7	0.6	11.9	12.2
w. Shadow-FT	39.9	18.3	11.7	1.0	16.2	17.4
w. Re-Adapt	33.6	13.3	8.2	0.6	15.0	14.1
w. LoRE-Adapt	31.6	12.4	7.6	0.2	11.7	12.7
w. **GIFT (Ours)**	**42.6**	**19.6**	**13.5**	**1.9**	**20.9**	**19.7**

Table 1: Average@16 accuracy of four large language models on mathematical reasoning benchmarks. The best performance of each model across benchmarks is bold. GIFT consistently surpasses all compared baselines across four model families, improving average accuracy while preserving general instruction-following robustness.

nations of the base model, instruction vector, and task-adapted parameters, using coefficients of 0.5 as recommended (Fleshman and Durme, 2024; Ilharco et al., 2023).

• **LoRE-Adapt** (Fleshman and Durme, 2024). A low-rank variant of Re-Adapt that applies truncated SVD to the instruction offset, yielding a compact instruction adapter.

Training Details. All methods are trained using the AdamW optimizer for 1 epoch with a maximum sequence length of 2048 tokens. To ensure a controlled comparison, we use an identical LoRA configuration across all experiments: global batch size of 256, learning rate 2×10^{-4}, LoRA rank $r = 64$, LoRA scaling $\alpha = 128$, LoRA dropout 0.05, and warmup ratio 0.1. Unless otherwise specified, we keep the remaining optimization settings fixed across all experiments. For mathematical reasoning benchmarks, following previous works (Wu et al., 2025b), we use stochastic decoding with temperature 1.0 and maximum generation length 4096. We sample 16 responses per problem and report Average@16, i.e., the average accuracy across these 16 stochastic samples. This setting maintains comparable accuracy while also reflecting the model's exploration ability under test-time sampling. For medical, we use standard multiple-choice accuracy.

Methods	MedQA	MMLU-medical	MedMCQA	Average
Llama3.1-8B-Instruct	55.2	75.1	57.4	62.6
w. SFT	50.0	64.0	57.9	57.3
w. Shadow-FT	65.6	73.8	55.9	65.1
w. Re-Adapt	63.6	71.4	54.2	63.1
w. LoRE-Adapt	62.6	70.1	54.3	62.4
w. GIFT (Ours)	**68.3**	**77.7**	**60.2**	**68.8**

Table 2: Results on medical QA benchmarks. GIFT improves knowledge acquisition while maintaining the general capabilities of the instruction model.

4.2 Main Results

From Table 1 and Table 2, we draw the following observations.

GIFT achieves consistent and substantial performance gains across models and benchmarks. Across five instruction-tuned backbones and all evaluated tasks, GIFT consistently improves average accuracy over both the original instruction models and prior adaptation methods. On mathematical reasoning, GIFT improves the average score by +5.2 on Llama3.1-8B, +3.0 on Llama3.2-3B, +2.9 on DeepSeek-Math-7B, and +1.6 on Qwen2.5-7B, demonstrating stable gains across model scales. In contrast, direct fine-tuning of instruction models (Instruct-SFT) leads to substantial degradation, particularly under limited supervision and stochas-

Ablation	Math-500	Minerva Math	Olympiad Bench	AIME24	AMC23	Average
Instruct Model	73.2	38.2	35.4	11.2	48.6	41.3
SFT+Merge	70.6	36.6	31.3	7.9	43.4	38.0
GIFT-BaseT	73.5	37.3	34.6	12.1	47.7	41.0
GIFT (ours)	75.0	38.4	36.1	12.7	52.5	42.9

Table 3: Ablation on Qwen2.5-7B. We compare the original instruction-tuned model, standard SFT with merge, a GIFT variant that uses the base model as teacher (GIFT-BaseT), and the full GIFT method that uses the instruction model as teacher.

tic decoding with temperature 1.0, which amplifies distributional shifts in instruction-tuned checkpoints. By learning task-specific updates on the base model and transferring them under instruction guidance, GIFT yields reliable improvements across all benchmarks.

GIFT remains effective even on strong instruction-tuned models. Improving highly optimized models such as Qwen2.5-7B-Instruct is particularly challenging. While Shadow-FT provide moderate gains on relatively weaker backbones (e.g., Llama and DeepSeek-Math), their performance degrades on Qwen2.5-7B-Instruct. In contrast, GIFT consistently delivers further improvements on this strong baseline, indicating that instruction-guided fine-tuning can introduce task-specific knowledge without disrupting existing capabilities.

GIFT generalizes across model families and domains. Beyond mathematical reasoning, GIFT also transfers effectively to knowledge-intensive medical QA. As shown in Table 2, GIFT improves the average score on Llama3.1-8B-Instruct from 62.6 to 68.8 (+6.2), achieving the best performance across MedQA, MMLU-medical, and MedMCQA. This result demonstrates that confidence-guided adaptation is not limited to reasoning-centric tasks, but also benefits fact-heavy domains that require accurate knowledge acquisition and structured decision-making.

4.3 Ablation

Table 3 presents an ablation study on Qwen2.5-7B, a strong and highly optimized instruction-tuned model on which further improvements are known to be challenging. The goal of this study is to isolate the contribution of guidance from the instruction model in the proposed GIFT framework.

We consider three representative adaptation settings. (1) **SFT+Merge** corresponds to the Shadow-FT setting described in Section 4.1, where a low-rank adapter is trained on the pretrained base model using standard supervised fine-tuning and then merged into the instruction-tuned model. (2) **GIFT-BaseT** adopts the same guided optimization framework as GIFT, but uses the base model itself as the teacher, allowing us to control for the effect of reweighting without instruction-aligned guidance. (3) **GIFT** uses the instruction-tuned model as the teacher during adapter training, and merges the resulting adapter back into the instruction model.

As shown in Table 3, directly merging an adapter trained via standard SFT on the base model (**SFT+Merge**) leads to a clear performance degradation compared to the original instruction-tuned model, with an average drop of 3.3 points. This result indicates that unguided task adaptation can introduce parameter updates that are incompatible with the instruction-tuned representation space, even when merged post hoc. Using the base model as the teacher (**GIFT-BaseT**) largely recovers the instruction baseline, achieving an average score comparable to the original instruction model (41.0 vs. 41.3), but fails to yield consistent improvements across benchmarks. This suggests that reweighting alone is insufficient when the guidance signal does not reflect instruction-aligned behavior. In contrast, the full **GIFT** method, which leverages guidance from the instruction-tuned model, consistently improves performance across all benchmarks, achieving a +1.6 average gain over the instruction baseline. This comparison highlights that the effectiveness of GIFT crucially depends on instruction-aligned guidance during fine-tuning, rather than merely reweighting tokens or transferring task-specific updates.

Overall, these results confirm that incorporating the instruction-tuned model during the fine-tuning stage is essential for learning adapters that are both merge-compatible and performance-improving.

5 Analyses

5.1 Test-Time Scaling

Inference-time strategies such as Best-of-N (BoN) sampling (Charniak and Johnson, 2005; Stiennon

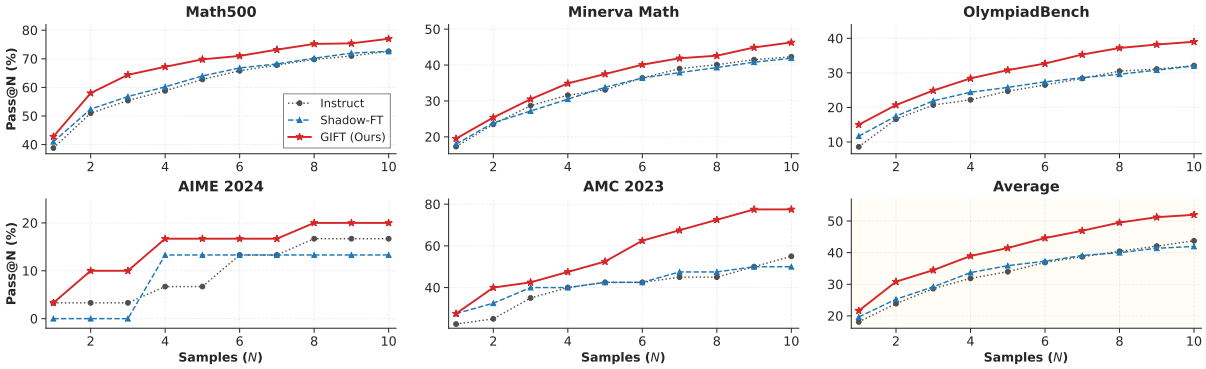

Figure 2: Pass@N performance under test-time scaling on Llama3.1-8B across five mathematical benchmarks and their average. GIFT consistently outperforms the instruction baseline and Shadow-FT at all sampling budgets.

et al., 2020), which generate multiple candidate outputs for each query, are widely adopted to enhance LLM reasoning (Welleck et al., 2024; Snell et al., 2025; Ruan et al., 2025b). The effectiveness of these methods relies on whether the generated candidates can sufficiently explore the solution space.

To quantify this effect, we adopt the **Pass@N** metric, which can be regarded as a special case of BoN (Chen et al., 2021). For a question q, let $\{o_i\}_{i=1}^{N}$ denote N outputs and $\mathcal{F}(o_i)$ a verifier returning 1 if o_i is correct and 0 otherwise. Then,

$$\text{Pass@}N(q) = \mathbb{I}\Big[\exists\, i \in \{1,\ldots,N\} \text{ s.t. } \mathcal{F}(o_i) = 1\Big]. \quad (7)$$

This metric evaluates whether at least one correct solution is found among N attempts.

Figure 2 reports the Pass@N performance of the instruction baseline, Shadow-FT, and GIFT on Llama3.1-8B across five mathematical benchmarks and their average. Across all tasks, **GIFT consistently outperforms both baselines at every sampling budget**. The performance gap is already present at $N = 1$ and remains stable as N increases, indicating that the gains introduced by GIFT are not restricted to single-sample accuracy. On more challenging benchmarks such as OlympiadBench and AIME 2024, GIFT exhibits stronger scaling behavior, accumulating correct solutions more effectively as additional samples are generated. In contrast, Shadow-FT shows limited or saturated improvements under larger sampling budgets. Overall, these results suggest that guided fine-tuning improves not only base accuracy but also the effectiveness with which models exploit test-time sampling, producing adaptations that are more compatible with inference-time scaling.

Model	MMLU	IFEval
Qwen2.5-7B-Instruct	68.7	71.2
w. Shadow-FT	68.6	71.9
w. GIFT	68.8	72.1
Llama3.1-8B-Instruct	63.2	73.8
w. Shadow-FT	63.7	73.6
w. GIFT	63.7	74.7

Table 4: Impact of merging task-adapted adapters on general knowledge (MMLU) and instruction-following (IFEval) tasks.

5.2 Impact on General Capabilities

Beyond task-specific improvements, we examine whether merging task-adapted adapters affects the general capabilities of instruction-tuned models. We evaluate merged models on two complementary benchmarks: MMLU (Hendrycks et al., 2020), which measures broad multi-domain knowledge, and IFEval (Zhou et al., 2023), which evaluates instruction-following robustness under strict prompt-level constraints.

Table 4 reports results on Qwen2.5-7B-Instruct and Llama3.1-8B-Instruct. Across both backbones, GIFT preserves the original instruction model's performance on MMLU, with results matching or slightly exceeding the instruction baseline. This indicates that guided fine-tuning and merging do not compromise general knowledge coverage.

On IFEval, GIFT maintains or slightly improves prompt-level strict accuracy after merging, compared to both the instruction baseline and Shadow-FT. Across both backbones, no degradation in instruction-following robustness is observed.

Overall, these results indicate that merging GIFT-trained adapters does not compromise the general

capabilities of instruction-tuned models. The preserved performance on MMLU and IFEval suggests that GIFT enables task adaptation while remaining largely neutral to the model's general knowledge and instruction-following behavior.

5.3 Generalization to Instruction Tasks

To further examine whether GIFT generalizes beyond math and medical QA, we construct an additional instruction-following evaluation from summarization-style tasks in Super-NaturalInstructions (Wang et al., 2022). We use the official default split, filter tasks by keywords related to summarization, sample 2,000 training examples, and evaluate on 100 test instances per task over 8 tasks (800 examples in total). Table 5 shows that GIFT improves both exact match and RougeL over the original instruction model and Shadow-FT. These results suggest that guided base-to-instruct transfer remains effective on a qualitatively different task family involving multi-token generation.

Model	EM	RougeL
Llama3.1-8B-Instruct	10.75	37.38
w. Shadow-FT	9.50	37.65
w. GIFT	**12.00**	**40.28**

Table 5: Results on summarization-style tasks from Super-NaturalInstructions.

5.4 Model Size Scaling

Model Size	Instruct	Shadow-FT	GIFT
Qwen2.5-0.5B	7.9	2.8	**8.3**
Qwen2.5-1.5B	18.3	9.8	**22.0**
Qwen2.5-3B	30.4	24.5	**32.8**
Qwen2.5-7B	41.3	38.0	**42.9**
Qwen2.5-14B	46.9	44.9	**48.1**
Qwen2.5-32B	50.6	48.2	**51.2**

Table 6: Average accuracy across five mathematical reasoning benchmarks for the Qwen2.5 family across different scales.

We further examine whether the gains of GIFT persist across different model scales. To this end, we conduct a model size scaling study on the Qwen2.5 family, which spans from 0.5B to 32B parameters under a unified architecture and training recipe. This setting allows us to analyze how guided adapter transfer interacts with model capacity in a controlled manner.

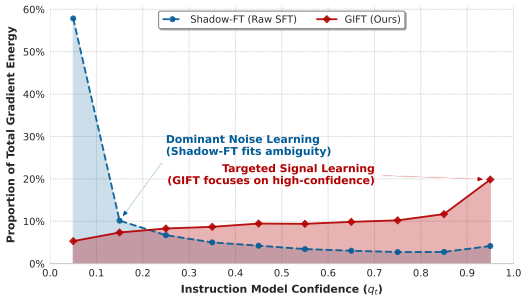

Figure 3: Training signal distribution across instruction-model confidence levels, where cross-entropy losses in GIFT are weighted by confidence q_t.

Table 6 reports the average accuracy across five mathematical reasoning benchmarks for each model size; full per-benchmark results are provided in Table 8. Across all scales, GIFT consistently outperforms both the instruction-tuned baseline and Shadow-FT, demonstrating stable improvements from small (0.5B) to large (32B) models.

Notably, the performance gap between GIFT and the baselines does not diminish as model size increases. While all methods benefit from larger model capacity, guided adapter transfer provides complementary gains that persist even for strong instruction-tuned models such as Qwen2.5-14B and Qwen2.5-32B. In contrast, Shadow-FT exhibits weaker or inconsistent improvements, particularly for smaller and mid-sized models, where unguided merging is more susceptible to instability.

Overall, these results indicate that GIFT scales favorably with model size and remains effective across a wide range of capacities, reinforcing its applicability as a general-purpose adaptation strategy for instruction-tuned language models.

5.5 Learning Signal Redistribution

We analyze how guided fine-tuning redistributes training signal across tokens with different instruction-model confidence levels. Following common practice, we use cross-entropy loss as a proxy for learning signal magnitude, where higher loss typically induces stronger parameter updates.

Our analysis covers over 500k target tokens from the NuminaMath-CoT training set using Llama3.1-8B. For each token, we record its cross-entropy loss ℓ_t and instruction-model confidence q_t, and group tokens into bins from low to high confidence. We compare Shadow-FT (Wu et al., 2025a), where the learning signal is proportional to ℓ_t, with **GIFT**, where it is proportional to $q_t \cdot \ell_t$.

Figure 3 shows the normalized contribution

Teacher Model	Time	Peak Mem.	File Size
Llama3.1-8B-Instruct	2m11s	$<$ 22GB	11.91MB
Qwen2.5-7B-Instruct	2m9s	$<$ 22GB	11.75MB

Table 7: Offline annotation cost for computing token-level confidence scores on 2,000 NuminaMath-CoT samples. The raw 2K file size is 2.9MB.

of each confidence region. Under Shadow-FT, 79.7% of the learning signal originates from low-confidence tokens, which also exhibit the highest average loss, indicating that standard fine-tuning emphasizes uncertain or ambiguous regions. In contrast, GIFT substantially reshapes this distribution. The contribution from low-confidence tokens drops to 29.6%, while medium- and high-confidence regions receive a much larger share. Notably, high-confidence tokens contribute only 6.9% of the signal under Shadow-FT but 31.5% under GIFT, despite having lower raw losses.

Overall, GIFT selectively redistributes learning signals rather than uniformly amplifying gradients. By suppressing updates from instruction-uncertain tokens and prioritizing instruction-consistent regions, GIFT steers optimization toward more compatible update directions, providing an empirical explanation for its improved stability and merge compatibility.

5.6 Offline Annotation Cost

Since GIFT requires a one-time offline pass over the training set to compute q_t, we additionally profile this preprocessing step. Table 7 summarizes the profiling results for the main mathematical setup with 2,000 NuminaMath-CoT samples on a single RTX 4090 24GB GPU using batch size 1. These results support the claim that the annotation stage is lightweight in our setting, although larger datasets would still incur additional preprocessing cost.

6 Conclusion

We introduced **GIFT**, a guided fine-tuning and transfer framework for adapting instruction-tuned language models under limited supervision. By leveraging effective confidence signals from the instruction model to guide adapter training on the base model, GIFT enables more stable and effective acquisition of task-specific knowledge. Extensive experiments on mathematical reasoning and knowledge-intensive benchmarks show that GIFT consistently outperforms direct fine-tuning and existing merge-based methods, while preserving the general instruction-following behavior of instruction-tuned models.

Limitations

GIFT requires an offline annotation step to compute confidence scores for the training dataset. Although this process is lightweight and performed only once, it introduces additional preprocessing cost for very large datasets. Exploring more efficient or on-the-fly approximations of guidance from the instruction model is a promising avenue for future research. While we focus on standard adapter merging for clarity, future work could explore combining GIFT with advanced merging techniques such as Fisher-weighted averaging or TIES. We leave this direction as future work, as it is orthogonal to the core contribution of guided fine-tuning.

Acknowledgements

This project was supported by National Natural Science Foundation of China (No. 62306132), Guangdong Basic and Applied Basic Research Foundation (No. 2025A1515011564), Natural Science Foundation of Shanghai (No. 25ZR1402136). We thank the anonymous reviewers for their insightful feedback on this work.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI@Meta. 2024. Llama 3 model card.

American Institute of Mathematics. 2024. Aime 2024 competition mathematical problems.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sheng Cao, Mingrui Wu, Karthik Prasad, Yuandong Tian, and Zechun Liu. 2025. Paramδ for direct mixing: Post-train large language model at zero cost. In *The Thirteenth International Conference on Learning Representations*.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative

reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 173–180, Ann Arbor, Michigan. Association for Computational Linguistics.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, and etc. 2021. Evaluating large language models trained on code. *Preprint*, arXiv:2107.03374.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

William Fleshman and Benjamin Van Durme. 2024. Readapt: Reverse engineered adaptation of large language models. *Preprint*, arXiv:2405.15007.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3828–3850.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Shih-Cheng Huang, Pin-Zu Li, Yu-chi Hsu, Kuang-Ming Chen, Yu Tung Lin, Shih-Kai Hsiao, Richard Tsai, and Hung-yi Lee. 2024. Chat vector: A simple approach to equip LLMs with instruction following and model alignment in new languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10943–10959, Bangkok, Thailand. Association for Computational Linguistics.

Weiyang Huang, Xuefeng Bai, Kehai Chen, Xinyang Chen, Yibin Chen, Weili Guan, and Min Zhang. 2026. Sat: Balancing reasoning accuracy and efficiency with stepwise adaptive thinking. *Preprint*, arXiv:2604.07922.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. 2019. Averaging weights leads to wider optima and better generalization. *Preprint*, arXiv:1803.05407.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

Peng Lai, Jianjie Zheng, Sijie Cheng, Yun Chen, Peng Li, Yang Liu, and Guanhua Chen. Beyond the surface: Enhancing llm-as-a-judge alignment with human via internal representations. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

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

Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. 2024. Numinamath. [https://huggingface.co/AI-MO/NuminaMath-CoT](https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf).

Pin-Jie Lin, Rishab Balasubramanian, Fengyuan Liu, Nikhil Kandpal, and Tu Vu. 2025. Efficient model development through fine-tuning transfer. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2617–2636, Suzhou, China. Association for Computational Linguistics.

Mathematical Association of America. 2023. Amc 2023 competition problems.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR.

Zhiwen Ruan, Yun Chen, Yutao Hou, Peng Li, Yang Liu, and Guanhua Chen. 2025a. Unveiling overmemorization in finetuning llms for reasoning tasks. *Preprint*, arXiv:2508.04117.

Zhiwen Ruan, Yixia Li, Yefeng Liu, Yun Chen, Weihua Luo, Peng Li, Yang Liu, and Guanhua Chen. 2025b. G2: Guided generation for enhanced output diversity in LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14116–14134, Suzhou, China. Association for Computational Linguistics.

Zhiwen Ruan, Yixia Li, He Zhu, Yun Chen, Peng Li, Yang Liu, and Guanhua Chen. 2025c. Enhancing large language model reasoning via selective critical token fine-tuning. *Preprint*, arXiv:2510.10974.

Zhiwen Ruan, Yixia Li, He Zhu, Longyue Wang, Weihua Luo, Kaifu Zhang, Yun Chen, and Guanhua Chen. 2025d. LayAlign: Enhancing multilingual reasoning in large language models via layer-wise adaptive fusion and alignment strategy. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1481–1495, Albuquerque, New Mexico. Association for Computational Linguistics.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2025. Scaling llm test-time compute optimally can be more effective than scaling model parameters. International Conference on Learning Representations (ICLR 2025).

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Hanqing Wang, Yixia Li, Shuo Wang, Guanhua Chen, and Yun Chen. 2025. MiLoRA: Harnessing minor singular components for parameter-efficient LLM finetuning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4823–4836, Albuquerque, New Mexico. Association for Computational Linguistics.

Tianyi Wang, Yixia Li, Long Li, Yibiao Chen, Shaohan Huang, Yun Chen, Peng Li, Yang Liu, and Guanhua Chen. 2026. Sppo: Sequence-level ppo for long-horizon reasoning tasks. *Preprint*, arXiv:2604.08865.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Super-naturalinstructions:generalization via declarative instructions on 1600+ tasks. In *EMNLP*.

Sean Welleck, Amanda Bertsch, Matthew Finlayson, Hailey Schoelkopf, Alex Xie, Graham Neubig, Ilia Kulikov, and Zaid Harchaoui. 2024. From decoding to meta-generation: Inference-time algorithms for large language models. *Transactions on Machine Learning Research*. Survey Certification.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. *Preprint*, arXiv:2203.05482.

Taiqiang Wu, Runming Yang, Jiayi Li, Pengfei Hu, Yik-Chung Wu, Ngai Wong, and Yujiu Yang. 2025a. Shadow-ft: Tuning instruct model via training on paired base model. *Preprint*, arXiv:2505.12716.

Yongliang Wu, Yizhou Zhou, Zhou Ziheng, Yingzhe Peng, Xinyu Ye, Xinting Hu, Wenbo Zhu, Lu Qi, Ming-Hsuan Yang, and Xu Yang. 2025b. On the generalization of sft: A reinforcement learning perspective with reward rectification. *Preprint*, arXiv:2508.05629.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models. *Preprint*, arXiv:2306.01708.

Jian Yang, Wei Zhang, Jiajun Wu, Junhang Cheng, Tuney Zheng, Fanglin Xu, Weicheng Gu, Lin Jing, Yaxin Du, Joseph Li, et al. 2026. Incoder-32b-thinking: Industrial code world model for thinking. *arXiv preprint arXiv:2604.03144*.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. *Preprint*, arXiv:2311.03099.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

He Zhu, Junyou Su, Peng Lai, Ren Ma, Wenjia Zhang, Linyi Yang, and Guanhua Chen. 2026. Anchored supervised fine-tuning. In *The Fourteenth International Conference on Learning Representations*.

A Supplementary Experimental Results

Table 8 reports the full per-benchmark results corresponding to the model size scaling study in Section 5.4. We include detailed accuracies on each mathematical benchmark to complement the averaged results in Table 6 and facilitate more fine-grained comparison across model scales and methods.

B Additional Experiments

B.1 Results on Qwen3-8B

We additionally evaluate GIFT on Qwen3-8B under the same NuminaMath-CoT training setup used in the main mathematical experiments. At evaluation time, rather than relying on training-free techniques (Lai et al.; Huang et al.,

Model	Math500	Minerva	Olympiad	AIME24	AMC23	Avg
Qwen2.5-0.5B-Instruct	22.9	3.1	4.2	0.6	8.6	7.9
w. Shadow-FT	9.5	1.2	1.5	0.0	1.7	2.8
w. GIFT	25.2	3.7	5.4	0.4	6.9	8.3
Qwen2.5-1.5B-Instruct	44.4	11.0	13.6	0.6	21.7	18.3
w. Shadow-FT	26.6	5.8	6.8	0.8	8.8	9.8
w. GIFT	51.1	12.8	16.5	2.3	27.5	22.0
Qwen2.5-3B-Instruct	62.9	23.3	24.3	5.2	36.1	30.4
w. Shadow-FT	54.7	18.6	18.6	3.1	27.3	24.5
w. GIFT	65.8	27.5	26.2	4.0	40.6	32.8
Qwen2.5-7B-Instruct	73.2	38.2	35.4	11.2	48.6	41.3
w. Shadow-FT	70.6	36.6	31.3	7.9	43.4	38.0
w. GIFT	75.0	38.4	36.1	12.7	52.5	42.9
Qwen2.5-14B-Instruct	77.5	44.5	40.0	14.2	58.1	46.9
w. Shadow-FT	76.9	43.8	38.1	10.8	54.8	44.9
w. GIFT	78.9	47.4	41.1	14.0	59.1	48.1
Qwen2.5-32B-Instruct	81.3	47.0	44.3	16.7	63.6	50.6
w. Shadow-FT	79.8	40.1	43.0	14.2	64.1	48.2
w. GIFT	81.5	48.6	44.8	16.0	65.2	51.2

Table 8: Full results on five mathematical reasoning benchmarks for Qwen2.5 models at different scales.

2026), we use the no-thinking mode by setting `enable_thinking=False` in the chat template, since the training responses are standard chain-of-thought solutions rather than explicit thinking-mode traces. Table 9 shows that GIFT remains effective on this newer model family, improving the Qwen3-8B baseline by 1.2 average points and outperforming Shadow-FT by 3.2 points.

B.2 Confidence Token Analysis

To better understand why instruction-model confidence can serve as a useful weighting signal, we compare token-type shares in the full NuminaMath-CoT training responses and in the high-confidence subset with $q_t \geq 0.9999$. We use digits as a simple proxy for math-critical content. Digits are consistently enriched among high-confidence tokens for both teachers: for Llama3.1-8B-Instruct, digits account for 11.35% of all response tokens but 20.69% of the high-confidence subset (1.82×); for Qwen2.5-7B-Instruct, the corresponding values are 14.49% and 23.27% (1.61×). This pattern suggests that instruction-derived confidence tends to upweight more structured, numerically grounded portions of mathematical solutions.

B.3 Confidence Across Difficulty Levels

To connect Figure 3 with benchmark difficulty, we additionally compute token-level confidence statistics on reference solutions from an easier set (GSM8K) and a harder set (AIME24). Across both teacher models, AIME24 consistently contains a larger share of low-confidence regions. With Llama3.1-8B-Instruct as teacher, GSM8K has $q_{\text{mean}} = 0.748$, $\text{frac}(q_t \geq 0.9) = 60.58\%$, and $\text{frac}(q_t < 0.5) = 23.86\%$, whereas AIME24 has $q_{\text{mean}} = 0.652$, $\text{frac}(q_t \geq 0.9) = 45.19\%$, and $\text{frac}(q_t < 0.5) = 34.04\%$. With Qwen2.5-7B-Instruct as teacher, GSM8K has $q_{\text{mean}} = 0.816$, $\text{frac}(q_t \geq 0.9) = 74.08\%$, and $\text{frac}(q_t < 0.5) = 17.80\%$, while AIME24 has $q_{\text{mean}} = 0.690$, $\text{frac}(q_t \geq 0.9) = 52.04\%$, and $\text{frac}(q_t < 0.5) = 30.07\%$. These statistics support the intuition that harder reasoning benchmarks expose more instruction-uncertain regions, where GIFT suppresses unstable learning signals.

B.4 Prompt Formatting Details

For completeness, we summarize the prompt formatting used in our experiments. For NuminaMath-CoT, we apply the chat template with `add_generation_prompt=True` and append the instruction line "Please reason step by step, and put your final answer within `\boxed{}`." to the user message. The training target is the provided solution response, which includes both the reasoning process and the boxed final answer.

For MedMCQA, we format each example as a

Method	Math-500	Minerva Math	Olympiad Bench	AIME24	AMC23	Average
Qwen3-8B	82.5	**37.6**	47.1	24.8	67.2	51.8
w. Shadow-FT	81.2	32.2	45.1	24.4	65.9	49.8
w. GIFT	**83.2**	36.9	**48.2**	**26.7**	**70.2**	**53.0**

Table 9: Additional mathematical reasoning results on Qwen3-8B.

multiple-choice prompt with options in the standard "A. / B. / C. / D." format, then append the instruction "Please reason step by step. At the end of your response, you MUST conclude with the exact phrase: 'So the answer to this question is [Option]'." This formatting makes answer extraction unambiguous during evaluation.

For the additional summarization experiment in Super-NaturalInstructions, each instance is serialized as `Task definition: <Definition>\n\n Input: <input>\n\n Output:`, and the first reference output is used as the response target.

C LLM Usage

In the preparation of this paper, we only used large language models (LLMs) as an assistive tool for grammar correction and text polishing.