Compatibility-Aware Dynamic Fine-Tuning for Large Language Models

Yucheng Zhou[1,*], **Junwei Sheng**[1,*], **Qianning Wang**[2], **Jianbing Shen**[1,✉]
[1] SKL-IOTSC, CIS, University of Macau, [2] Auckland University of Technology
yucheng.zhou@connect.um.edu.mo, jianbingshen@um.edu.mo

Abstract

Supervised Fine-Tuning (SFT) is the predominant paradigm for aligning large language models (LLMs), yet it suffers from optimization instability and limited generalization. Recent work attributes this issue to pathological gradient scaling and proposes Dynamic Fine-Tuning (DFT) to correct it at the token level. However, DFT assumes all demonstrations are equally suitable learning targets, an assumption violated by the strong heterogeneity of large-scale instruction data, where demonstration-policy mismatch induces high-variance updates at the sample level. We introduce **Compatibility-Aware Dynamic Fine-Tuning (CADFT)**, a principled extension of DFT that controls sample-level optimization variance. CADFT derives a dynamic, policy-dependent compatibility signal from model likelihoods to modulate supervised updates, suppressing high-variance gradients from incompatible demonstrations. We further propose a delayed, low-frequency compatibility-guided rewriting strategy to transform persistently incompatible demonstrations into learnable targets. We show that CADFT can be interpreted as a variance-controlled estimator that generalizes token-level stabilization in DFT to the sample level. Extensive experiments demonstrate improved stability, generalization, and cold-start reinforcement learning initialization, while remaining fully supervised and independent of explicit reward modeling.

1 Introduction

Supervised fine-tuning (SFT) is the dominant paradigm for aligning large language models (LLMs) with downstream tasks and instruction-following behaviors. By maximizing the likelihood of expert demonstrations under a teacher-forcing regime, SFT provides a simple, stable, and scalable training framework (Ouyang et al., 2022; Chung et al., 2024; Wei et al., 2022). Despite its empirical success, recent theoretical and empirical studies have revealed that standard SFT suffers from a fundamental optimization pathology. When viewed through the lens of policy optimization, the SFT gradient implicitly corresponds to a distorted objective in which low-probability tokens induce disproportionately large gradient updates (Wu et al., 2025; Chu et al., 2025). This inverse-probability amplification leads to high gradient variance, training instability, and poor generalization, particularly on reasoning-intensive tasks under distribution shift (Chu et al., 2025; Lin et al., 2017; Zheng et al., 2025).

Dynamic Fine-Tuning (DFT) was recently proposed to address this issue at the token level (Wu et al., 2025). DFT reformulates SFT into a probability-aware objective that corrects the pathological gradient scaling induced by rare tokens, yielding bounded and more stable updates. Crucially, DFT achieves this without introducing reinforcement learning components such as reward models, on-policy sampling, or policy optimization algorithms. However, DFT implicitly assumes that all demonstrations in the dataset are equally suitable learning targets. In practice, large-scale instruction datasets are highly heterogeneous. Some demonstrations are well-aligned with the model's current inductive biases and capability level, while others are excessively complex, poorly structured, or semantically mismatched. Even when token-level gradient instability is corrected, such *demonstration-policy mismatch* can induce high-variance updates at the sample level, leading to inefficient learning and unstable optimization (Zhou et al., 2023; Bengio et al., 2009).

[*] Equal Contribution.

[✉] Corresponding Author. This work was supported by the National Natural Science Foundation of China (No. 624B2002), the Science and Technology Development Fund of Macau SAR (FDCT) under grants 0134/2025/RIA2, and the Jiangyin Hi-tech Industrial Development Zone under the Taihu Innovation Scheme (EF2025-00003-SKL-IOTSC).

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 29997–30008

This observation suggests that stabilizing supervised fine-tuning requires controlling not only *how* token-level gradients are scaled, but also *which* demonstrations exert strong influence on parameter updates, and *to what extent*, under the current model state. In other words, effective fine-tuning demands a mechanism for regulating sample-level compatibility between demonstrations and the evolving policy.

In this work, we propose **Compatibility-Aware Dynamic Fine-Tuning (CADFT)**, a principled extension of DFT that incorporates a dynamic, policy-dependent compatibility signal into the supervised objective. CADFT treats compatibility as a relative measure of demonstration-policy alignment, computed from the model's own likelihoods and normalized adaptively during training. This signal is used to modulate the strength of sample-level updates, suppressing high-variance gradients induced by incompatible demonstrations while preserving informative supervision.

Importantly, CADFT does not discard low-compatibility demonstrations outright. To avoid permanently ignoring difficult but potentially valuable data, we further introduce a conservative, delayed rewriting mechanism that selectively reformulates persistently incompatible demonstrations into targets that lie within the model's current feasible region. Rewriting is activated only after a warm-up phase and at low frequency, preventing premature self-reinforcement and maintaining training stability.

CADFT preserves the supervised learning paradigm of DFT and introduces no reinforcement learning, reward modeling, or policy optimization machinery. From a theoretical perspective, CADFT can be understood as a variance-controlled extension of DFT that generalizes token-level stabilization to the sample level. Empirically, CADFT consistently improves optimization stability, generalization performance, and downstream reinforcement learning initialization across language, code, and multimodal reasoning tasks.

Our main contributions are as follows:

- We show that samples with low compatibility induce higher-variance gradient updates, and that mitigating such updates improves optimization stability and generalization.
- We propose **Compatibility-Aware Dynamic Fine-Tuning (CADFT)**, a simple and principled method that incorporates a dynamic, normalized compatibility signal to modulate sample-level update strength within a fully supervised framework.
- We introduce a delayed, low-frequency compatibility-guided rewriting strategy that transforms incompatible demonstrations into learnable targets.
- We provide a theoretical interpretation of CADFT as a variance-controlled estimator and empirically demonstrate its effectiveness across mathematical reasoning, code generation, multimodal reasoning, and cold-start reinforcement learning settings.

2 Related Work

2.1 SFT and RL in LLM Alignment

Supervised Fine-Tuning (SFT) aligns LLMs with downstream tasks (Zhou et al., 2025a, 2026a,b; Hu et al., 2025) by maximizing the likelihood of expert demonstrations (Wei et al., 2022; Zhou et al., 2023; Chung et al., 2024), effectively performing imitation learning or behavioral cloning (Mandlekar et al., 2021). However, SFT overfits training distributions and generalizes poorly to OOD inputs (Zhou et al., 2024), whereas RL optimizes task-level objectives via reward signals for improved generalization (Christiano et al., 2017; Ouyang et al., 2022; Bai et al., 2022), albeit with substantial computational overhead and instability (Schulman et al., 2017; Strubell et al., 2019). Empirical studies confirm that RL-based fine-tuning yields superior robustness on reasoning-intensive tasks, making the SFT–RL generalization gap a central alignment challenge (Chu et al., 2025; Swamy et al., 2025).

To bridge this gap, hybrid methods combine SFT and RL: RLHF refines SFT with a learned reward model (Ouyang et al., 2022), DPO directly optimizes from preference data without explicit rewards (Rafailov et al., 2023), group-relative variants reduce reliance on absolute rewards (Shao et al., 2024), Negative-aware Fine-Tuning uses incorrect generations as implicit negative feedback (Chen et al., 2025), and self-rewarding vision-language models optimize prompts via iterative self-feedback (Yang et al., 2025)—all extending beyond pure supervised learning (Ouyang et al., 2022; Rafailov et al., 2023). Theoretically, Du et al. (2025) reinterpret RLHF as reward-weighted SFT, Wang et al. (2025) analyze SFT as RL with an implicit reward, and Qin and Springenberg (2025)

model SFT as offline RL with importance weighting; however, these expectation-level analyses do not characterize the variance of the resulting gradient estimators, which is critical for stable optimization. From a broader perspective, structured constraints and feedback mechanisms further underscore the importance of principled objective design (Askell et al., 2021), as also evidenced by abnormal-aware feedback in medical VL models (Zhou et al., 2025b; Zheng et al., 2026), rubric-guided reinforcement learning for emotional support (Yuan et al., 2025), and reinforcing VL frameworks for sign language translation (Rao et al., 2025).

2.2 Stabilizing SFT and Gradient Reweighting

Several works stabilize SFT through loss reweighting or objective modification. MixCE combines forward and reverse cross-entropy to balance mode-covering and mode-seeking behaviors (Zhang et al., 2023), and related importance-weighting ideas appear in offline RL from demonstrations (Mandlekar et al., 2021; Qin and Springenberg, 2025). Entropy-guided optimization for autoregressive generation (Song et al., 2026) also demonstrates that controlling entropy during training yields more stable and coherent synthesis. Wu et al. (2025) show that the SFT gradient is equivalent to an offline policy gradient estimator with implicit rewards and inverse-probability importance weighting, and propose Dynamic Fine-Tuning (DFT) to rectify the resulting pathological scaling by rescaling token-level gradients with the model's own probabilities. Unlike heuristic methods such as Focal Loss (Lin et al., 2017), DFT focuses on variance correction rather than emphasizing difficult samples. Abdolmaleki et al. (2025) further show that improper feedback weighting under mixtures of positive and negative feedback leads to instability, reinforcing the need for principled gradient control.

2.3 Data Quality and Sample Compatibility

While DFT stabilizes token-level optimization, it assumes all demonstrations are equally suitable learning targets (Wu et al., 2025). In practice, heterogeneous datasets can induce demonstration-policy mismatch and high-variance updates at the sample level (Mandlekar et al., 2021; Liu and Zhang, 2025). Liu and Zhang (2025) show in knowledge distillation that selectively downweighting low-compatibility samples improves stability, and prior work on curriculum learning further confirms that supervision effectiveness depends on the alignment between sample difficulty and model capacity (Mandlekar et al., 2021; Liu and Zhang, 2025; Chu et al., 2025). Indiscriminately updating on low-compatibility demonstrations forces the model to memorize patterns it cannot reliably internalize (Liu and Zhang, 2025; Chu et al., 2025).

Inspired by these findings, our work introduces a compatibility-aware extension of DFT that addresses demonstration-policy mismatch at the sample level within a fully supervised framework. In summary, prior work has improved SFT either by combining it with RL or by stabilizing token-level gradients; our work is the first to unify token-level and sample-level variance control (Wu et al., 2025; Liu and Zhang, 2025).

3 Compatibility-Aware Dynamic Fine-Tuning

In this section, we present **Compatibility-Aware Dynamic Fine-Tuning (CADFT)**, a robust alignment framework designed to stabilize supervised fine-tuning under heterogeneous data distributions. We first revisit Dynamic Fine-Tuning (DFT), then introduce a dynamic, sample-level compatibility signal for reweighting updates, and finally describe an optional delayed rewriting mechanism. The overall procedure is summarized in Algorithm 1.

3.1 Preliminaries

Let $\mathcal{D} = \{(x, y)\}$ denote a dataset of instruction-response pairs, and $\pi_\theta(y|x)$ a language model parameterized by θ. Standard Supervised Fine-Tuning (SFT) minimizes the negative log-likelihood:

$$\mathcal{L}_{\text{SFT}}(x, y) = -\sum_{t=1}^{|y|} \log \pi_\theta(y_t \mid x, y_{<t}). \quad (1)$$

The gradient magnitude of $\mathcal{L}_{\text{SFT}}$ scales inversely with $\pi_\theta(y_t|\cdot)$, causing low-probability tokens to induce disproportionately large updates and high optimization variance.

Dynamic Fine-Tuning (DFT) (Wu et al., 2025) addresses this issue by rectifying token-level gradient scaling. From a gradient perspective, DFT induces updates proportional to $(1+\log p_t)$, which remain bounded as $p_t \to 0$. This effectively neutralizes the inverse-probability amplification present in SFT and stabilizes token-level optimization. However, DFT operates purely at the token level and

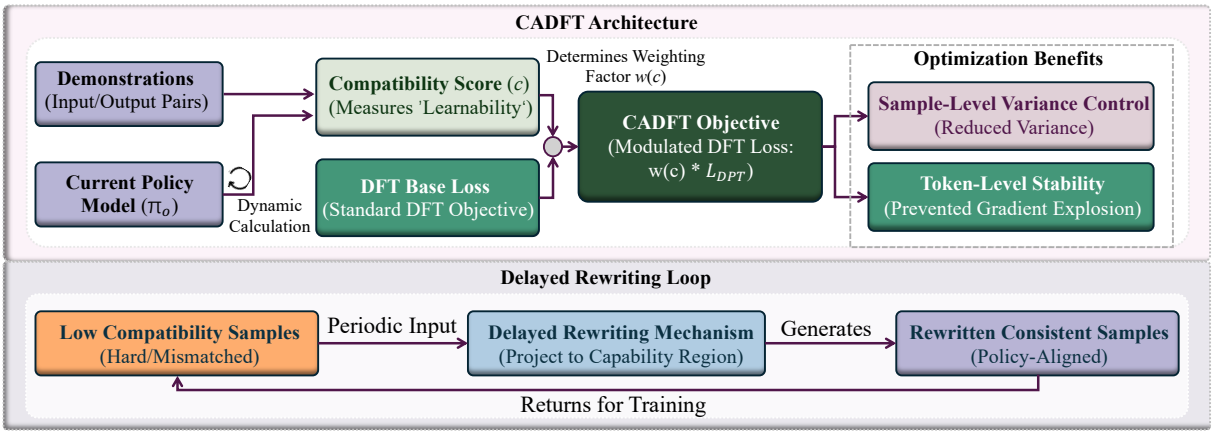

Figure 1: Overall framework of Compatibility-Aware Dynamic Fine-Tuning (CADFT). CADFT extends DFT by incorporating a sample-level compatibility signal that modulates update strength and optionally guides delayed demonstration rewriting.

implicitly assumes all demonstrations are equally suitable learning targets.

3.2 Dynamic Compatibility Assessment

We posit that demonstrations vary in their suitability for learning at different stages of training. We therefore introduce a *dynamic compatibility* signal that measures how well a demonstration aligns with the model's current inductive bias.

Raw Compatibility Score. For a sample (x, y), we define the raw compatibility score as the length-normalized negative log-likelihood:

$$c_{\text{raw}}(x, y; \theta) = \frac{1}{|y|} \sum_{t=1}^{|y|} -\log \pi_\theta(y_t \mid x, y_{<t}). \quad (2)$$

Lower values indicate higher compatibility. As training progresses, however, the absolute scale of c_{raw} shifts, rendering static thresholds ineffective.

Adaptive Normalization. To obtain a scale-invariant signal, we normalize raw compatibility scores within each effective global mini-batch $\mathcal{B}$, where $\mathcal{B}$ aggregates all micro-batches across data-parallel workers. Let $\mu_{\mathcal{B}}$ and $\sigma_{\mathcal{B}}$ denote the mean and standard deviation of c_{raw} in the batch, computed via distributed all-reduce synchronization to ensure consistency across devices. The normalized score is:

$$\hat{c}_i = \frac{c_{\text{raw}}(x_i, y_i) - \mu_{\mathcal{B}}}{\sigma_{\mathcal{B}} + \epsilon}, \quad (3)$$

where ϵ ensures numerical stability. Importantly, $\hat{c}$ represents a *relative and model-dependent* notion of compatibility, reflecting alignment with the current model state rather than an absolute measure of difficulty.

3.3 Compatibility-Aware Objective

CADFT integrates the compatibility signal into Dynamic Fine-Tuning (DFT) through a sample-level weighting function $w(\hat{c})$ that modulates the strength of supervised updates. We employ a soft exponential decay:

$$w(\hat{c}_i) = \exp(-\beta \cdot \max(0, \hat{c}_i)), \quad (4)$$

where $\beta \geq 0$ controls the sensitivity of the weighting mechanism.

This design preserves the full contribution of samples whose normalized compatibility $\hat{c}_i$ is no worse than the batch average ($\hat{c}_i \leq 0$), while progressively down-weighting less compatible samples ($\hat{c}_i > 0$). As a result, demonstrations that are misaligned with the model's current inductive bias exert reduced influence on optimization, mitigating high-variance updates without discarding potentially useful data.

The resulting compatibility-aware objective is defined as:

$$\mathcal{L}_{\text{CADFT}}(\mathcal{B}) = \frac{1}{|\mathcal{B}|} \sum_{(x,y)\in\mathcal{B}} w(\hat{c}(x, y)) \cdot \mathcal{L}_{\text{DFT}}(x, y) \quad (5)$$

From an optimization perspective, this formulation is conceptually related to self-paced or curriculum learning: samples exhibiting higher compatibility exert stronger influence on parameter updates, while harder or misaligned demonstrations are incorporated more conservatively as training progresses.

3.4 Delayed Compatibility-Guided Rewriting

While compatibility-based reweighting mitigates the influence of incompatible samples, it may un-

Algorithm 1 Compatibility-Aware Dynamic Fine-Tuning

Require: Dataset $\mathcal{D}$, model π_θ, batch size B, warm-up steps T_{warm}, rewrite interval K, compatibility sensitivity β

for training step $t = 1, \ldots, T_{\max}$ **do**
  Sample mini-batch $\mathcal{B} = \{(x_i, y_i)\}_{i=1}^{B}$ from $\mathcal{D}$
  // Compute compatibility (no gradient)
  for each $(x_i, y_i) \in \mathcal{B}$ **do**
    $c_i \leftarrow \frac{1}{|y_i|} \sum_t -\log \pi_\theta(y_{i,t} \mid x_i, y_{i,<t})$
  end for
  Compute batch mean $\mu_{\mathcal{B}}$ and std $\sigma_{\mathcal{B}}$
  $\hat{c}_i \leftarrow \text{stop\_grad}\left(\frac{c_i - \mu_{\mathcal{B}}}{\sigma_{\mathcal{B}} + \epsilon}\right)$
  $w_i \leftarrow \exp(-\beta \cdot \max(0, \hat{c}_i))$
  // Compatibility-aware DFT update
  $\mathcal{L} \leftarrow \frac{1}{B} \sum_i w_i \cdot \mathcal{L}_{\text{DFT}}(x_i, y_i)$
  Update parameters $\theta \leftarrow \text{Optimizer}(\theta, \nabla\mathcal{L})$
  // Optional delayed rewriting
  if $t > T_{\text{warm}}$ **and** $t \bmod K = 0$ **then**
    Identify small subset $\mathcal{S} \subset \mathcal{D}$ with highest moving-average compatibility scores
    for each $(x, y) \in \mathcal{S}$ **do**
      Generate $\hat{y} \sim \pi_\theta(\cdot|x)$ via nucleus sampling
      Optionally replace y with $\hat{y}$
    end for
  end if
end for

derutilize demonstrations that are correct but substantially misaligned with the model's current capability. To address this, we further explore a conservative delayed rewriting mechanism that optionally reformulates persistently incompatible samples into more learnable targets.

Two-Stage Training. We divide training into two stages to avoid premature self-reinforcement:

1. **Warm-up Stage ($t < T_{\text{warm}}$):** Training proceeds solely with the compatibility-aware objective, allowing the model to acquire stable instruction-following behavior.
2. **Rewriting Stage ($t \geq T_{\text{warm}}$):** At periodic intervals, a small subset of samples with persistently high moving-average compatibility scores may be selected for optional rewriting. Their original targets can be replaced with model-generated alternatives that lie within the model's current feasible region.

Specifically, rewritten targets are sampled as:

$$\hat{y} \sim \text{NucleusSampling}(\pi_\theta(\cdot|x); p = 0.9, T = 0.7). \tag{6}$$

This process can be viewed as projecting overly hard demonstrations onto the model's current hypothesis class, converting high-variance supervision into stable but simplified learning signals.

3.5 Theoretical Perspective: Variance Reduction

Let $g_i = \nabla\mathcal{L}_{\text{DFT}}(x_i, y_i)$ denote the stochastic gradient of sample i. Under the commonly observed assumption that incompatible or low-probability demonstrations induce disproportionately large gradient norms in SFT-style training, such samples tend to dominate the second moment of the gradient estimator without contributing proportionally to the mean update direction.

By applying the compatibility weight, CADFT uses $\tilde{g}_i = w(\hat{c}_i)\, g_i$. As $w(\hat{c}_i)$ decays for increasingly incompatible samples, the weighted second moment $\mathbb{E}[\|\tilde{g}\|^2]$ is reduced relative to standard DFT. Consequently, CADFT acts as a variance-controlled estimator that stabilizes optimization by modulating gradients based on semantic compatibility rather than arbitrary norm clipping.

4 Experiments

We conduct comprehensive experiments to evaluate the effectiveness of **Compatibility-Aware Dynamic Fine-Tuning (CADFT)**. Our experimental study is designed to answer the following questions: (i) whether CADFT consistently improves over SFT and DFT across tasks and models scales; (ii) how compatibility-aware reweighting affects optimization stability; (iii) whether CADFT provides a stronger initialization for downstream reinforcement learning; and (iv) which design choices are critical to its effectiveness?

We evaluate CADFT on mathematical reasoning, code generation, and multimodal reasoning tasks, under both supervised fine-tuning and reinforcement learning settings.

4.1 Experimental Setup

Models. We evaluate CADFT on a diverse set of open-source language and vision-language models, including LLaMA-3 series (Team, 2024), DeepSeekMath (Shao et al., 2024), Qwen2.5-Math (Yang et al., 2024), Qwen2.5-Coder (Hui et al., 2024), and Qwen2.5-VL (Bai et al., 2025), covering multiple parameter scales. For fair comparison, all methods share identical model architectures and initial checkpoints.

Datasets, Benchmarks, and Evaluation. We evaluate CADFT on benchmarks spanning mathematical reasoning, code generation, and multimodal reasoning. Mathematical reasoning is evaluated on Math500 (Hendrycks et al., 2021), Min-

erva Math (Lewkowycz et al., 2022), OlympiadBench (He et al., 2024), AIME 2024 (Hendrycks et al., 2021), and AMC 2023 (Hendrycks et al., 2021). Code generation is evaluated on HumanEval (Chen et al., 2021), HumanEval+ (Liu et al., 2023), and MultiPL-E (Cassano et al., 2023) across nine programming languages. Multimodal reasoning is evaluated on MathVerse (Zhang et al., 2024), MathVision (Wang et al., 2024), and We-Math (Qiao et al., 2025).

For all benchmarks, we strictly follow the official evaluation protocols and metrics provided by each dataset. Mathematical reasoning performance is reported using Average@16 accuracy, code generation using pass@1 accuracy, and multimodal benchmarks using accuracy-based metrics.

Training Details. We follow the training protocol of DFT (Wu et al., 2025). All models are trained using identical batch sizes, optimizers, learning rates, and training steps across SFT, DFT, and CADFT. The effective global batch size is 256, achieved via data-parallel synchronization and gradient accumulation. Compatibility scores are computed per mini-batch and normalized dynamically using $\mu_{\mathcal{B}}$ and $\sigma_{\mathcal{B}}$ synchronized across all data-parallel workers via all-reduce, ensuring that normalization is performed on the effective global batch (rather than per-device micro-batches) and is invariant to sharding strategy. The compatibility statistics are detached from gradient computation. For CADFT-specific hyperparameters, we set the compatibility normalization to per-mini-batch z-score with $\epsilon = 10^{-6}$ and the weighting sensitivity to $\beta = 1.0$. When delayed rewriting is enabled, we use a warm-up of $T_{\text{warm}} = 3000$ steps, a rewriting interval of $K = 1000$ steps, and rewrite a fraction of 0.5% of the dataset per interval with replacement probability 0.5. Rewritten targets are generated via nucleus sampling with $p = 0.9$ and temperature $T = 0.7$.

4.2 Main Results

Mathematical Reasoning Performance Table 1 presents the main results on mathematical reasoning benchmarks. Across all evaluated model families and scales, CADFT consistently outperforms both SFT and DFT, with particularly large gains on the most challenging benchmarks. Compared to vanilla SFT, CADFT avoids the severe performance degradation observed on OlympiadBench, AIME 2024, and AMC 2023, where demonstration-policy mismatch is pronounced. While DFT mitigates token-level instability, it still treats all demonstrations as equally informative, allowing incompatible samples to induce noisy updates. By explicitly down-weighting such samples, CADFT further reduces optimization variance and enables more stable learning from heterogeneous supervision. Notably, the relative improvement of CADFT over DFT grows with model scale. This suggests that as model capacity increases, sample-level mismatch becomes a dominant source of optimization noise, making compatibility-aware control increasingly important.

Code Generation Performance Table 2 summarizes code generation results on HumanEval and MultiPL-E. CADFT consistently improves performance over both SFT and DFT across all evaluated models. Beyond aggregate gains, CADFT exhibits particularly strong improvements on lower-resource and syntactically diverse languages such as Bash and PHP within MultiPL-E. This indicates that compatibility-aware reweighting discourages overfitting to high-frequency patterns in dominant languages (e.g., Python), thereby improving cross-language generalization. These results support the hypothesis that sample-level heterogeneity is a major source of optimization variance in multilingual code generation.

Multimodal Mathematical Reasoning We further evaluate CADFT on multimodal mathematical reasoning benchmarks. As shown in Table 3, CADFT consistently improves performance across vision-only, vision-intensive, and vision-dominant regimes on MathVerse, MathVision, and WeMath. Multimodal settings introduce additional sources of demonstration-policy mismatch due to imperfect visual grounding and varying degrees of visual dependency. While DFT stabilizes token-level updates, it cannot distinguish between well-grounded and poorly grounded demonstrations. CADFT alleviates this issue by suppressing high-variance updates from incompatible multimodal samples, leading to more robust vision-language alignment.

4.3 Cold-Start Reinforcement Learning Initialization

We investigate whether CADFT provides a stronger initialization for downstream reinforcement learning. Following prior work (Wu et al., 2025), models are further optimized using GRPO (DeepSeek-AI, 2025) after SFT, DFT, or CADFT initialization.

Model	Math500	Minerva Math	Olympiad Bench	AIME24	AMC23	Avg.
LLaMA-3.2-3B	1.63	1.36	1.01	0.41	1.56	1.19
w/ SFT	8.65	2.38	2.06	0.00	3.13	3.24
w/ DFT	12.79	2.84	2.90	0.83	3.91	4.65
w/ **CADFT**	**14.80**	**3.20**	**3.35**	**1.05**	**4.60**	**5.40**
LLaMA-3.1-8B	1.86	0.98	0.94	0.21	1.01	1.00
w/ SFT	16.85	5.78	3.88	0.00	5.16	6.33
w/ DFT	27.44	8.26	6.94	0.41	12.03	11.02
w/ **CADFT**	**31.20**	**9.40**	**8.20**	**0.65**	**14.20**	**12.73**
DeepSeekMath-7B	6.15	2.15	1.74	0.21	2.97	2.64
w/ SFT	26.83	7.26	6.33	0.41	8.28	9.82
w/ DFT	41.46	16.79	15.00	1.24	16.25	18.15
w/ **CADFT**	**47.80**	**18.10**	**17.80**	**1.65**	**19.40**	**20.15**
Qwen2.5-Math-1.5B	31.66	8.51	15.88	4.16	19.38	15.92
w/ SFT	43.76	13.04	12.63	1.87	18.75	18.01
w/ DFT	64.89	20.94	27.08	6.87	38.13	31.58
w/ **CADFT**	**72.30**	**22.80**	**33.69**	**8.76**	**45.87**	**33.42**
Qwen2.5-Math-7B	40.12	14.39	17.12	6.68	27.96	21.25
w/ SFT	53.96	16.66	18.93	2.48	26.09	23.62
w/ DFT	68.20	30.16	33.83	8.56	45.00	37.15
w/ **CADFT**	**75.50**	**32.63**	**39.50**	**10.41**	**52.20**	**41.44**

Table 1: **Mathematical reasoning performance (Average@16).** Accuracy (%) of five representative large language models on diverse mathematical reasoning benchmarks. For each backbone model, we report results under vanilla fine-tuning (SFT), Dynamic Fine-Tuning (DFT), and the proposed Compatibility-Aware DFT (CADFT).

Model	HE	HE+	Python	C++	Java	PHP	TS	C#	Bash	JS	Avg.
Qwen2.5-3B	43.3	36.0	43.29	40.99	37.34	37.89	47.17	43.04	24.68	45.96	40.05
w/ SFT	41.5	34.8	42.07	42.24	37.97	37.27	43.40	41.77	20.25	47.83	39.10
w/ DFT	45.7	39.0	45.73	44.72	41.77	45.34	42.14	43.04	27.85	44.10	41.84
w/ **CADFT**	**47.8**	**41.0**	**48.20**	**46.50**	**43.60**	**47.80**	**44.30**	**45.10**	**30.20**	**46.80**	**44.14**
Qwen2.5-Coder-3B	52.4	42.7	51.83	53.42	46.20	47.20	54.09	55.06	25.32	54.04	48.39
w/ SFT	51.8	43.9	51.22	51.55	48.10	54.66	59.12	51.27	34.18	54.04	50.52
w/ DFT	56.7	50.0	57.32	54.66	51.27	58.39	58.49	60.76	31.01	53.42	53.16
w/ **CADFT**	**59.5**	**53.0**	**60.20**	**57.00**	**53.80**	**61.20**	**61.00**	**63.50**	**34.50**	**56.00**	**56.67**
Qwen2.5-Coder-7B	62.2	53.0	63.41	63.98	53.16	59.01	62.89	59.49	39.24	60.87	57.76
w/ SFT	54.9	48.8	54.88	64.60	51.27	62.11	68.55	60.76	33.54	65.22	57.62
w/ DFT	67.7	59.8	67.68	67.70	54.43	60.87	70.44	65.19	48.73	63.35	62.30
w/ **CADFT**	**71.3**	**61.5**	**70.50**	**70.00**	**56.80**	**63.00**	**72.80**	**67.80**	**52.00**	**66.00**	**65.24**

Table 2: **Code generation performance on HumanEval and MultiPL-E.** We report pass@1 accuracy (%) on HumanEval (HE, HE+) and MultiPL-E across nine programming languages. HE and HE+ are subsets of the HumanEval benchmark, while all language-specific scores belong to MultiPL-E.

Model	Vision Only	Vision Intensive	Vision Dominant	Overall	MathVision	WeMath
Qwen2.5-VL-3B	28.81	30.96	31.60	33.83	21.25	4.10
w/ SFT	30.96	33.63	32.74	35.66	21.02	23.33
w/ DFT	32.49	35.91	33.50	37.54	22.30	23.71
w/ **CADFT**	**34.20**	**38.20**	**35.60**	**39.90**	**23.60**	**25.10**

Table 3: **Multi-modal mathematical reasoning performance.** Comparison on MathVerse, MathVision, and WeMath benchmarks under different visual reasoning regimes. Scores reflect overall accuracy (%). The proposed CADFT consistently improves performance across vision-only and vision-intensive settings.

Model	Math500	Minerva Math	Olympiad Bench	AIME24	AMC23	Avg.
Qwen2.5-Math-1.5B w/ SFT+GRPO	62.54	23.10	26.92	5.00	40.15	31.54
Qwen2.5-Math-1.5B w/ DFT+GRPO	65.96	23.51	28.37	8.63	41.40	33.57
Qwen2.5-Math-1.5B w/ **CADFT+GRPO**	**73.40**	**25.10**	**35.80**	**9.82**	**48.60**	**35.62**

Table 4: **Cold-start mathematical reasoning with GRPO.** All models are first initialized via supervised fine-tuning (SFT), DFT, or CADFT, and subsequently optimized using GRPO. Results demonstrate that CADFT provides a stronger initialization for downstream reinforcement learning.

Model	HE	HE+	Python	C++	Java	PHP	TS	C#	Bash	JS	Avg.
Qwen2.5-Coder-3B w/ SFT+GRPO	57.3	50.6	57.32	63.35	51.27	63.98	68.55	60.76	33.54	66.46	58.15
Qwen2.5-Coder-3B w/ DFT+GRPO	68.9	61.0	68.90	67.08	55.06	62.73	70.44	65.19	49.37	62.11	62.61
Qwen2.5-Coder-3B w/ **CADFT+GRPO**	**70.2**	**62.5**	**70.10**	**68.30**	**56.40**	**63.80**	**71.60**	**66.40**	**51.00**	**63.50**	**63.88**

Table 5: **Code generation with GRPO fine-tuning.** All models are further optimized with GRPO after SFT or DFT initialization. Results are reported on HumanEval (HE, HE+) and MultiPL-E benchmarks. CADFT yields consistently stronger GRPO-aligned representations.

Model	Vision Only	Vision Intensive	Vision Dominant	Overall	MathVision	WeMath
Qwen2.5-VL-3B w/ SFT+GRPO	32.48	33.50	43.78	35.93	21.44	21.43
Qwen2.5-VL-3B w/ DFT+GRPO	34.64	37.31	37.06	39.06	23.35	26.19
Qwen2.5-VL-3B w/ **CADFT+GRPO**	**36.14**	**39.30**	**38.10**	**42.10**	**24.90**	**29.30**

Table 6: **Multi-modal reasoning with GRPO optimization.** Comparison of SFT-, DFT-, and CADFT-initialized models after GRPO fine-tuning. CADFT consistently delivers stronger alignment between visual perception and reasoning under reinforcement learning.

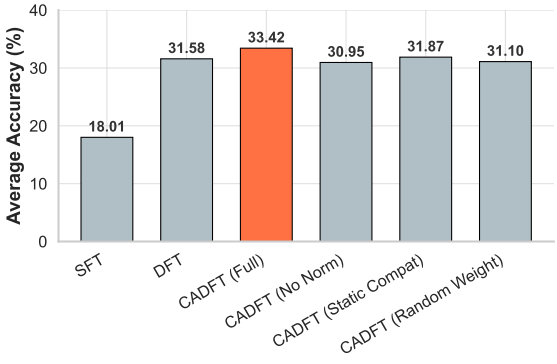

Figure 2: **Ablation of compatibility definition and dynamicity.** Dynamic, normalized compatibility yields consistent gains over DFT, while static, unnormalized, or random reweighting degrades performance.

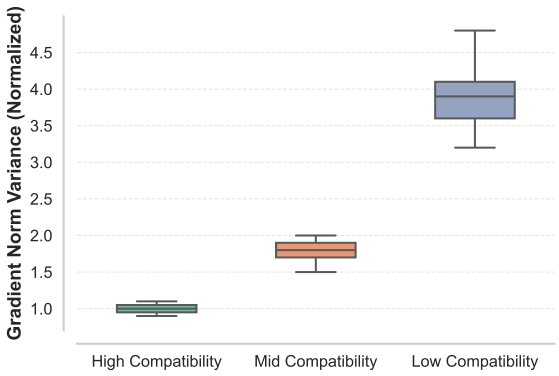

Figure 3: **Gradient norm variance across compatibility groups.** Low-compatibility samples induce significantly higher gradient variance, motivating compatibility-aware variance control.

Method	$w(c)$	Avg Acc.
DFT	-	31.58
CADFT (Exp)	$\exp(-\alpha c)$	**33.42**
CADFT (Linear Clip)	$\max(0, 1 - c/\tau)$	32.10
CADFT (Binary)	$\mathbb{I}[c < \tau]$	30.88
CADFT (Inverse)	$1/c$	28.94

Table 7: Ablation of weighting functions $w(c)$ on mathematical reasoning. We compare a soft monotonic exponential decay, a linearly clipped weighting, a binary filter, and an inverse weighting scheme.

Sample Group	Compat. Level	Grad Var.
Top 30%	Low	1.00
Middle 40%	Medium	1.78
Bottom 30%	High	3.92

Table 8: Gradient norm variance across sample groups with different compatibility levels. Lower compatibility samples induce substantially higher gradient variance.

Specifically, we adopt the same GRPO protocol and implementation choices as Wu et al. (2025). Correctness is determined by `math_verify` as the verifier-based reward signal. GRPO is trained in the `verl` framework with learning rate 1e-6, global batch size 256, warmup ratio 0.1, and number of sampled responses per prompt $n = 4$. All other GRPO hyperparameters follow the official DFT implementation scripts. As shown in Tables 4-6, CADFT-initialized models consistently outperform SFT+GRPO and DFT+GRPO across mathematical reasoning, code generation, and multimodal reasoning tasks. These results indicate that CADFT produces representations with lower gradient noise and better-aligned supervision, which facilitates subsequent policy optimization. Importantly, CADFT does not optimize for any reinforcement learning objective during pretraining. The observed gains suggest that reducing supervised optimization variance is complementary to, rather than competing with, reinforcement learning.

4.4 Ablation Studies and Analysis

We conduct a series of ablation studies to isolate the contribution of each design component in CADFT, including compatibility definition, weighting function shape, gradient variance control, and delayed rewriting. All ablations are evaluated under the

Method	SFT	DFT	CADFT
Grad Norm Variance	4.85	2.61	**1.72**

Table 9: Overall gradient norm variance under different fine-tuning methods. CADFT achieves the lowest variance, indicating the most stable optimization.

Method	Rewrite Start	Interval	Avg Acc.
DFT	-	-	31.58
CADFT (No Rewrite)	-	-	32.25
CADFT (Early)	0	Epoch	30.72
CADFT (Delayed)	Warm-up	Epoch	**33.42**
CADFT (Aggressive)	Warm-up	100 steps	31.40

Table 10: Ablation of compatibility-guided rewriting strategies. Rewriting too early or too frequently leads to premature self-reinforcement, while delayed rewriting after a warm-up phase achieves the best performance.

same training and evaluation settings for fair comparison.

Effect of Compatibility Definition and Dynamicity. We first study how the definition and dynamicity of compatibility affect performance. As shown in Figure 2, dynamically normalized compatibility consistently outperforms all static or unnormalized variants. Static compatibility definitions fail to account for the evolving model policy, causing samples to be permanently over- or underweighted as training progresses. Unnormalized compatibility is sensitive to scale drift in likelihood values, leading to unstable weighting behavior. In contrast, dynamic, batch-normalized compatibility provides a relative and policy-dependent signal, allowing CADFT to adaptively suppress incompatible samples throughout training. The random reweighting baseline further confirms that the observed gains do not arise from implicit regularization or noise injection, but from structured, compatibility-aware modulation.

Impact of Weighting Function Shape. Table 7 examines the effect of different weighting function shapes $w(c)$ while keeping all other components fixed. The exponential decay function achieves the best overall performance, as it provides smooth and monotonic suppression of incompatible samples without introducing discontinuities. Hard filtering or binary weighting removes gradient contributions abruptly, leading to optimization instability and reduced sample efficiency. Inverse weighting overly amplifies low-compatibility samples, resulting in high-variance updates. These results indicate that soft, monotonic weighting is critical for balancing stability and information retention in compatibility-aware optimization.

Gradient Variance Across Compatibility Levels. To directly validate our theoretical motivation, we analyze gradient norm variance across samples grouped by compatibility. Table 8 and Figure 3 show that low-compatibility samples induce substantially higher gradient variance than high-compatibility ones. This observation provides empirical evidence that demonstration-policy mismatch is a major source of optimization noise in supervised fine-tuning. By down-weighting such samples, CADFT effectively suppresses high-variance updates at the sample level. Consistently, Table 9 shows that CADFT achieves the lowest overall gradient variance among SFT, DFT, and CADFT, confirming its role as a variance-controlled estimator.

Effect of Delayed Compatibility-Guided Rewriting. Finally, we study the impact of delayed demonstration rewriting. As shown in Table 10, early or aggressive rewriting significantly degrades performance, indicating premature self-reinforcement when the model is not yet stable. In contrast, delayed rewriting after a warm-up phase consistently improves performance. This suggests that rewriting is beneficial only after the model has acquired a stable inductive bias, at which point projecting incompatible demonstrations into the model's feasible region reduces variance without reinforcing spurious solutions. These results show that delayed, conservative rewriting complements compatibility-aware reweighting, while aggressive rewriting undermines training stability.

5 Conclusion

We presented **Compatibility-Aware Dynamic Fine-Tuning (CADFT)**, a principled extension of Dynamic Fine-Tuning that explicitly controls sample-level optimization variance in supervised fine-tuning. By introducing a dynamic, policy-dependent compatibility signal, CADFT suppresses high-variance updates from mismatched demonstrations while preserving informative supervision. A delayed and low-frequency rewriting strategy further enables conservative utilization of persistently incompatible data. Both theoretically and empirically, CADFT generalizes token-level stabilization in DFT to the sample level, yielding improved stability, generalization, and stronger initialization for downstream reinforcement learning, without introducing reward models or on-policy optimization.

Limitations

CADFT builds on signals derived from the model's own likelihood estimates and therefore inherits the inductive biases and representational capacity of the underlying backbone. As a result, the effectiveness of compatibility estimation is naturally bounded by the model's current expressive power and pretraining quality.

References

Abbas Abdolmaleki, Bilal Piot, Bobak Shahriari, Jost Tobias Springenberg, Tim Hertweck, Michael Bloesch, Rishabh Joshi, Thomas Lampe, Junhyuk Oh, Nicolas Heess, Jonas Buchli, and Martin A. Riedmiller. 2025. Learning from negative feedback, or positive feedback or both. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Benjamin Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, and 3 others. 2021. A general language assistant as a laboratory for alignment. *CoRR*, abs/2112.00861.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Ming-Hsuan Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *CoRR*, abs/2502.13923.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, and 12 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *CoRR*, abs/2204.05862.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, volume 382 of *ACM International Conference Proceeding Series*, pages 41–48. ACM.

Federico Cassano, John Gouwar, Daniel Nguyen, Sydney Nguyen, Luna Phipps-Costin, Donald Pinckney, Ming-Ho Yee, Yangtian Zi, Carolyn Jane Anderson, Molly Q. Feldman, Arjun Guha, Michael Greenberg, and Abhinav Jangda. 2023. Multipl-e: A scalable and polyglot approach to benchmarking neural code generation. *IEEE Trans. Software Eng.*, 49(7):3675–3691.

Huayu Chen, Kaiwen Zheng, Qinsheng Zhang, Ganqu Cui, Yin Cui, Haotian Ye, Tsung-Yi Lin, Ming-Yu Liu, Jun Zhu, and Haoxiang Wang. 2025. Bridging supervised learning and reinforcement learning in math reasoning. *CoRR*, abs/2505.18116.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. Evaluating large language models trained on code. *CoRR*, abs/2107.03374.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4299–4307.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V. Le, Sergey Levine, and Yi Ma. 2025. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*. OpenReview.net.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, and 16 others. 2024. Scaling instruction-finetuned language models. *J. Mach. Learn. Res.*, 25:70:1–70:53.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948.

Yuhao Du, Zhuo Li, Pengyu Cheng, Zhihong Chen, Yuejiao Xie, Xiang Wan, and Anningzhe Gao. 2025. Simplify RLHF as reward-weighted SFT: A variational method. *CoRR*, abs/2502.11026.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. 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), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 3828–3850. Association for Computational Linguistics.

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 *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

He Hu, Yucheng Zhou, Qianning Wang, Yingjian Zou, Chiyuan Ma, Juzheng Si, Jianzhuang Liu, Zitong Yu, Laizhong Cui, and Fei Ma. 2025. From pattern recognizers to personalized companions: A survey of large language models in mental health.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Kai Dang, An Yang, Rui Men, Fei Huang, Xingzhang Ren, Xuancheng Ren, Jingren Zhou, and Junyang Lin. 2024. Qwen2.5-coder technical report. *CoRR*, abs/2409.12186.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. 2022. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2999–3007. IEEE Computer Society.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Lingyuan Liu and Mengxiang Zhang. 2025. Less is more: Selective reflection for compatible and efficient knowledge distillation in large language models. *CoRR*, abs/2508.06135.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. 2021. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning, 8-11 November 2021, London, UK*, volume 164 of *Proceedings of Machine Learning Research*, pages 1678–1690. PMLR.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Runqi Qiao, Qiuna Tan, Guanting Dong, Minhui Wu, Chong Sun, Xiaoshuai Song, Jiapeng Wang, Zhuoma Gongque, Shanglin Lei, Yifan Zhang, Zhe Wei, Miaoxuan Zhang, Runfeng Qiao, Xiao Zong, Yida Xu, Peiqing Yang, Zhimin Bao, Muxi Diao, Chen Li, and Honggang Zhang. 2025. We-math: Does your large multimodal model achieve human-like mathematical reasoning? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 20023–20070. Association for Computational Linguistics.

Chongli Qin and Jost Tobias Springenberg. 2025. Supervised fine tuning on curated data is reinforcement learning (and can be improved). *CoRR*, abs/2507.12856.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Zhi Rao, Yucheng Zhou, Benjia Zhou, Yiqing Huang, Sergio Escalera, and Jun Wan. 2025. Rvlf: A reinforcing vision-language framework for gloss-free sign language translation. *arXiv preprint arXiv:2512.07273*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347.

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

Han Song, Yucheng Zhou, Jianbing Shen, and Yu Cheng. 2026. From broad exploration to stable synthesis: Entropy-guided optimization for autoregressive image generation. In *The Fourteenth International Conference on Learning Representations*.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3645–3650. Association for Computational Linguistics.

Gokul Swamy, Sanjiban Choudhury, Wen Sun, Zhiwei Steven Wu, and J. Andrew Bagnell. 2025. All roads lead to likelihood: The value of reinforcement learning in fine-tuning. *CoRR*, abs/2503.01067.

Llama Team. 2024. Cathe llama 3 herd of models. *CoRR*, abs/2407.21783.

Bo Wang, Qinyuan Cheng, Runyu Peng, Rong Bao, Peiji Li, Qipeng Guo, Linyang Li, Zhiyuan Zeng, Yunhua Zhou, and Xipeng Qiu. 2025. Implicit reward as the bridge: A unified view of SFT and DPO connections. *CoRR*, abs/2507.00018.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024. Measuring multimodal mathematical reasoning with math-vision dataset. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

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

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. 2024. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *CoRR*, abs/2409.12122.

Hongji Yang, Yucheng Zhou, Wencheng Han, and Jianbing Shen. 2025. Self-rewarding large vision-language models for optimizing prompts in text-to-image generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 7332–7349.

Jiahao Yuan, Zhiqing Cui, Hanqing Wang, Yuansheng Gao, Yucheng Zhou, and Usman Naseem. 2025. Kardia-r1: Unleashing llms to reason toward understanding and empathy for emotional support via rubric-as-judge reinforcement learning. *arXiv preprint arXiv:2512.01282*.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, Peng Gao, and Hongsheng Li. 2024. MATHVERSE: does your multi-modal LLM truly see the diagrams in visual math problems? In *ECCV 2024*, volume 15066, pages 169–186. Springer.

Shiyue Zhang, Shijie Wu, Ozan Irsoy, Steven Lu, Mohit Bansal, Mark Dredze, and David S. Rosenberg. 2023. Mixce: Training autoregressive language models by mixing forward and reverse cross-entropies. In *ACL 2023*, pages 9027–9050. Association for Computational Linguistics.

Huan Zheng, Yucheng Zhou, Tianyi Yan, Dubing Chen, Hongbo Lu, Wenlong Liao, Tao He, Pai Peng, and Jianbing Shen. 2026. Clinical cognition alignment for gastrointestinal diagnosis with multimodal llms. *arXiv preprint arXiv:2603.20698*.

Huan Zheng, Yucheng Zhou, Tianyi Yan, Jiayi Su, Hongjun Chen, Dubing Chen, Xingtai Gui, Wencheng Han, Runzhou Tao, Zhongying Qiu, and 1 others. 2025. From human intention to action prediction: Intention-driven end-to-end autonomous driving. *arXiv preprint arXiv:2512.12302*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: less is more for alignment. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Yucheng Zhou, Xiang Li, Qianning Wang, and Jianbing Shen. 2024. Visual in-context learning for large vision-language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15890–15902.

Yucheng Zhou, Jianbing Shen, and Yu Cheng. 2025a. Weak to strong generalization for large language models with multi-capabilities. In *The Thirteenth International Conference on Learning Representations*.

Yucheng Zhou, Lingran Song, and Jianbing Shen. 2025b. Improving medical large vision-language models with abnormal-aware feedback. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12994–13011.

Yucheng Zhou, Jihai Zhang, Guanjie Chen, Jianbing Shen, and Yu Cheng. 2026a. Less is more: Vision representation compression for efficient video generation with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 13826–13834.

Yucheng Zhou, Huan Zheng, Dubing Chen, Hongji Yang, Wencheng Han, and Jianbing Shen. 2026b. From medical llms to versatile medical agents: A comprehensive survey. *Authorea Preprints*.