BOOSTING REINFORCEMENT LEARNING WITH VERIFIABLE REWARDS VIA RANDOMLY SELECTED FEW-SHOT GUIDANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement Learning with Verifiable Rewards (RLVR) has achieved great success in developing Large Language Models (LLMs) with chain-of-thought rollouts. It is used for many tasks such as math and coding. Nevertheless, RLVR struggles with sample efficiency on difficult problems where correct rollouts are hard to generate. Prior works propose to address this issue via demonstration-guided RLVR, i.e., to conduct Supervised FineTuning (SFT) when RL fails. However, SFT often requires a lot of data, which can be expensive to acquire. In this paper, we propose FEST, a **FE**w-**S**ho**T** demonstration-guided RLVR algorithm. It attains compelling results with only 128 demonstrations randomly selected from an SFT dataset. We find that three components are vital for the success: supervised signal, on-policy signal, and decaying weights on the few-shot SFT dataset to prevent overfitting from multiple-epoch training. Across several benchmarks, FEST outperforms baselines in the few-shot setting and even matches the performance of baselines trained with orders of magnitude more SFT data.

1 INTRODUCTION

Two years after Reinforcement Learning from Human Feedback (RLHF) and GPT-3.5 (Ouyang et al., 2022) propelled Large Language Models (LLMs) to the forefront of AI research (Ke et al., 2025), a new RL paradigm has emerged. Following OpenAI o1 (Jaech et al., 2024) and DeepSeek-R1 (Guo et al., 2025a), Reinforcement Learning with Verifiable Rewards (RLVR) (Guo et al., 2025a) has become the second dominant RL paradigm in the community. Unlike RLHF, which assigns rewards based on subjective and often vague human preferences (Wang et al., 2024a), RLVR leverages objective, verifiable rewards—such as unit tests for coding (Jiang et al., 2025) or ground-truth comparisons for mathematics (Cobbe et al., 2021). Consequently, RLVR is exceptionally well-suited for reasoning-heavy tasks. Driven by this approach, state-of-the-art LLMs have attained gold-medal performance in international competitions (Huang & Yang, 2025) and are beginning to tackle open problems at the frontiers of human knowledge (OpenAI, 2026a;b).

Despite its impressive performance, RLVR is beset by a long-standing challenge in RL: low sample efficiency on complex tasks. While prior knowledge from pre-training and Supervised Fine-Tuning (SFT) can partially mitigate this—analogous to imitation learning (Attia & Dayan, 2018)—RL-trained LLMs still struggle to explore beyond the capabilities of their base models (Yue et al., 2025). For instance, in mathematical tasks with binary rewards, a batch that fails to yield a single correct answer results in an advantage of 0, providing no learning signal. To address this, state-of-the-art algorithms such as DAPO (Yu et al., 2025) and CISPO (Chen et al., 2025a) employ repeated sampling until a success is found, which can triple the computational overhead on average (Zhang et al., 2025b).

To address these challenges, recent research proposes a novel paradigm known as demonstration-guided RL, or unified post-training (Yan et al., 2025). In this framework, SFT is integrated with RL, particularly when RL sampling fails to produce positive rollouts (Lv et al., 2025; Ma et al., 2026). However, these approaches demand a large volume of SFT data, which can be prohibitively expensive (Au-Yeung, 2025). For instance, curating just 2,500 questions for "Humanity's Last Exam" (Phan et al., 2025) required *1,000 graduate-degree holders*, even when providing only brief rationales—a level of effort that scales poorly for training data involving long reasoning traces.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

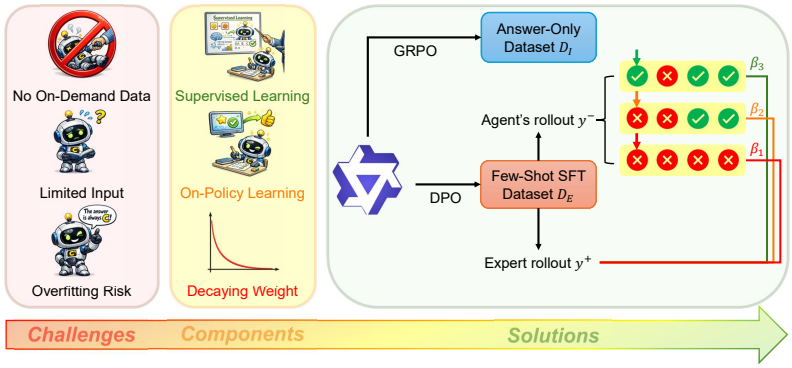


Figure 1: Overview of our work. We introduce a few-shot demonstration-guided RLVR paradigm to address three primary challenges: the lack of on-demand data, limited expert input, and high overfitting risk. To mitigate, FEST incorporates three vital components—supervised learning, on-policy learning, and decaying weight—to effectively leverage few-shot data. Specifically, FEST jointly optimizes the model using GRPO on the answer-only data (D_I) and a semi-online DPO loss on the few-shot SFT data (D_E). The latter utilizes adaptive β coefficients to weight different types of rollouts, bridging the gap between expert demonstrations (y^+) and agent-generated samples (y^-).

Table 1: Comparison of FEST with existing demonstration-guided RLVR baselines. SFT data requirements for HPT and SuperRL are estimated through replication (with SuperRL figures derived from training on OpenR1-220K); for ReLIFT, SASR, MIFO, and CHORD, we utilize the ratios reported in their respective papers. Notably, while other methods address competition-level problems, SASR focuses on simpler tasks. We omit DyME (Liu et al., 2026) due to its similarity to HPT and its focus on vision-based tasks rather than mathematical reasoning.

RLVR Method	SFT Data Used	SFT Data Prepared	Random SFT Data?
SuperRL (Liu et al., 2025b)	Approx. 50K	220K	×
LUFFY (Yan et al., 2025)	46K	46K	×
SRFT (Fu et al., 2026)	46K	46K	×
HPT (Lv et al., 2025)	10K	46K	×
ReLIFT (Ma et al., 2026)	8.6K	46K	×
MIFO (Yuan et al., 2025)	6.4K	46K	×
CHORD (Zhang et al., 2026b)	5K	5K	✓
SASR (Chen et al., 2025c)	2K*	4K*	×
FEST (Ours)	**128**	**128**	✓

While bootstrapping and distilling from existing models is an alternative (Comanici et al., 2025), such practices raise significant concerns regarding proprietary API costs (Irugalbandara et al., 2024) and potential model collapse (Shumailov et al., 2024). In contrast, answer-only RL data are more accessible; large-scale mathematical Q&A pairs can be mined from online forums, while raw community answers often necessitate extensive filtering or rewriting before serving as high-quality SFT demonstrations (Zhang, 2024; Mahdavi et al., 2025; Albalak et al., 2025).

In this paper, we propose FEST, a *FEw-ShoT demonstration-guided RLVR algorithm* designed to thrive with as few as 128 randomly selected examples from an SFT dataset. To extract substantial performance gains from such a limited, uncurated dataset, we identify three critical components: (i) a supervised learning signal to provide expert guidance; (ii) an on-policy signal to mitigate exposure bias (Bengio et al., 2015) and serve as adversarial training for enhanced robustness; and (iii) a decaying weight to prevent overfitting. We find that incorporating a semi-online Direct Preference Optimization (DPO) (Rafailov et al., 2023; Chen et al., 2024b; Tan et al., 2025) loss—where demonstrations serve as positive examples and agent rollouts as negative ones—satisfies all three requirements. Furthermore, while our primary RL framework, Group Relative Policy Optimization (GRPO) (Shao et al., 2024), operates on token-level log probabilities, standard DPO relies on sequence-level values. To resolve this mismatch, we introduce a variant of FEST that decomposes

the DPO loss, replacing the online component with a GRPO loss featuring negative advantages, as supported by prior work (Zhu et al., 2025b). Our framework is illustrated in Fig. 1.

Our contributions are summarized as follows: (i) We introduce a novel post-training paradigm, few-shot demonstration-guided RLVR, which significantly boosts RLVR performance with minimal SFT data (see Tab. 1 for a comparison with existing methods). (ii) We develop FEST by elucidating and integrating three vital components essential for this few-shot training regime. (iii) Theoretically, we extend the unified post-training framework of HPT (Lv et al., 2025) by incorporating DPO (Appendix B.3). (iv) We empirically validate our approach across multiple benchmarks, demonstrating that FEST consistently outperforms various strong baselines.

2 PRELIMINARIES

Reinforcement Learning with Verifiable Rewards (RLVR). RLVR is an emerging post-training framework that facilitates solving tasks with objective ground truths, such as mathematics and programming, by eliciting complex Chain-of-Thought (CoT) reasoning (Wei et al., 2022). For a given prompt x representing the input task[1], the model generates a response y_0 sampled from the policy $\pi(\cdot|x)$. Since y_0 is a token sequence generated autoregressively, the probability is defined as $\pi(y_0|x) = \prod_{i=1}^{m} \pi(y_{0,i}|x, y_{0,<i})$, where $y_0 = \{y_{0,1}, y_{0,2}, \ldots, y_{0,m}\}$ denotes a sequence of m tokens. Upon generation, a verifiable reward $r(x, y_0) \in \mathbb{R}$ quantifies the quality of the response, typically through deterministic rules such as unit tests, symbolic checkers, or exact string matching. The objective of RLVR is to optimize the policy π to maximize the expected reward $\max_\pi \mathbb{E}_{y\sim\pi(\cdot|x)}[r(x, y)]$.

Group Relative Policy Optimization (GRPO). GRPO is a well-established, critic-free RLVR method. For prompt x, GRPO first samples a group of n rollouts $\{y_1, y_2, \ldots, y_n\}$ with corresponding rewards $\{r_1, r_2, \ldots, r_n\}$, where each $y_i = \{y_{i,1}, y_{i,2}, \ldots, y_{i,m}\}$. The objective is to minimize

$$-\frac{1}{nM}\sum_{i=1}^{n}\sum_{j=1}^{|y_i|}\min\left[\frac{\pi_\theta(y_{i,j}|x, y_{i,<j})}{\pi_{\theta_{\text{old}}}(y_{i,j}|x, y_{i,<j})}A_i, \text{clip}\left(\frac{\pi_\theta(y_{i,j}|x, y_{i,<j})}{\pi_{\theta_{\text{old}}}(y_{i,j}|x, y_{i,<j})}, 1-\epsilon_1, 1+\epsilon_2\right)A_i\right], \quad (1)$$

where $\epsilon_2 > \epsilon_1 > 0$ following DAPO (Yu et al., 2025). $\pi_{\theta_{\text{old}}}$ represents the reference policy from the current sampling-training iteration, and M is the upper limit of y_i's length $|y_i|$.[2] The advantage A_i is calculated relative to the group mean as $A_i = r(x, y_i) - \frac{1}{n}\sum_{k=1}^{n} r(x, y_k)$. Following recent work (Lv et al., 2025; Ma et al., 2026), we omit the KL regularizer and the standard deviation in advantage for simplicity.

Direct Preference Optimization (DPO). DPO (Rafailov et al., 2023) is an offline RLHF framework that enables policy optimization directly from preference data without the need for an explicit reward model. Given an offline dataset D consisting of triples (x, y^+, y^-), where y^+ and y^- represent the preferred and non-preferred responses respectively, DPO minimizes the following objective:

$$-\mathbb{E}_{(x,y^+,y^-)\sim D}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y^+|x)}{\pi_{\text{ref}}(y^+|x)} - \beta\log\frac{\pi_\theta(y^-|x)}{\pi_{\text{ref}}(y^-|x)}\right)\right], \quad (2)$$

where π_{ref} denotes the reference policy prior to training, and $\beta > 0$ is a hyperparameter.

REINFORCE. REINFORCE (Williams, 1992) is a fundamental policy gradient algorithm that remains widely utilized in contemporary RLVR research (Ahmadian et al., 2024). In the RLVR setting, given a prompt x, a model response $y \sim \pi(\cdot|x)$, and a reward $r(x, y)$, REINFORCE aims to maximize the expected reward $\mathbb{E}_{x,y}[r(x, y)]$. The empirical loss gradient is defined as $-r(x, y) \cdot \nabla \log \pi(y|x)$.

3 METHODOLOGY

In this section, we first delineate the three unique challenges inherent in few-shot demonstration-guided RLVR and identify the vital components of a post-training objective designed to address them (Sec. 3.1). We then introduce our algorithm, FEST, in Sec. 3.2, followed by its variant, FEST-GRPO, to address its limitation on gradient mismatch in Sec. 3.3.

[1] Unless otherwise noted, we consider single-turn interactions with sequence-level rewards.

[2] Following HPT and "Dr. GRPO" (Liu et al., 2025c), we use the formulation without token mean.

3.1 THE THREE UNIQUE CHALLENGES AND THE THREE VITAL COMPONENTS

In this framework, we optimize an LLM using two distinct datasets: a few-shot SFT dataset D_E containing **E**xpert-curated reasoning traces, and a large-scale, answer-only (and thus **I**mperfect) RL dataset D_I. Our primary objective is to fully exploit the minimal reasoning traces in D_E to enhance model performance beyond what is achievable through standard RLVR on D_I alone.

This setting introduces three unique challenges: *(i) No on-demand data:* Due to limited expert access, demonstrations cannot be generated on-demand for arbitrary questions where the model fails. This removes the flexibility assumed in many prior works such as HPT (Lv et al., 2025), ReLIFT (Ma et al., 2026), LUFFY (Yan et al., 2025) and MIFO (Yuan et al., 2025). *(ii) Limited Semantic Coverage:* The limited volume of SFT data is insufficient to cover the broad reasoning paradigms required. Furthermore, unlike specialized few-shot works like LIMOv2 (Ye et al., 2025), we do not assume a carefully curated pipeline; D_E may simply be a random batch of samples. *(iii) Overfitting Risk:* Given the minuscule size of D_E, repeated training over multiple epochs risks severe overfitting, which can degrade the model's general reasoning capability.

We identify three vital components to address these challenges: *supervised learning*, *on-policy learning*, and *adaptive weight scheduling*. We argue that these components must be carefully integrated when training on D_E. First, supervised learning is essential as it provides the only source of external knowledge beyond the binary reward signals in RLVR. Second, on-policy learning addresses the first and second challenges by allowing the model to evaluate its own rollouts against SFT traces. This expands the learning basis for the limited questions in D_E (Min et al., 2024) and mitigates exposure bias (Bengio et al., 2015). Finally, adaptive weight scheduling is crucial for tackling the third challenge. We employ a *decaying weight* strategy, ensuring the model prioritizes learning from D_E in early stages while refraining from overfitting as the RLVR signal on D_I becomes more dominant. Similar principles are observed in HPT (Lv et al., 2025), where the SFT data ratio is reduced to 2% toward the end of training (See Appendix D.4).

In conclusion, we require an algorithm for D_E that incorporates both supervised and on-policy loss terms, governed by a decaying weight. As we discuss below, semi-online DPO (Chen et al., 2024b; Tan et al., 2025) serves as an ideal framework to satisfy these requirements.

3.2 FEST: FEW-SHOT DEMONSTRATION-GUIDED RLVR

We define our training objective as follows. We optimize the parameters θ of the LLM policy π_θ using a semi-online DPO loss on the few-shot dataset D_E, where the SFT data serves as the preferred rollout y^+ and the RL-generated data acts as the non-preferred rollout y^-. As established, we utilize a GRPO loss for the answer-only dataset D_I and a semi-online DPO loss for the few-shot dataset D_E. Specifically, we use $L = c \cdot L_E + L_I$ where

$$
\begin{aligned}
L_E &= -\mathbb{E}_{(x,y^+)\sim D_E,\, y^-\sim\pi_{\theta_{\text{old}}}(\cdot|x)}\big[\log\sigma(\beta r^+ - \beta r^-)\big], \\
L_I &= -\frac{1}{nM}\mathbb{E}_{x\sim D_I,\, y\sim\pi_{\theta_{\text{old}}}(\cdot|x)}\sum_{i=1}^{n}\sum_{j=1}^{|y_i|}\min\big[\rho_{i,j}A_i,\ \text{clip}(\rho_{i,j},1-\epsilon_1,1+\epsilon_2)A_i\big].
\end{aligned}
\tag{3}
$$

In this objective, $\rho_{i,j} = \frac{\pi_\theta(y_{i,j}|x,y_{i,<j})}{\pi_{\theta_{\text{old}}}(y_{i,j}|x,y_{i,<j})}$, $r^+ = \log\frac{\pi_\theta(y^+|x)}{\pi_{\text{ref}}(y^+|x)}$, $r^- = \log\frac{\pi_\theta(y^-|x)}{\pi_{\text{ref}}(y^-|x)}$, $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function, and $c > 0$ is a constant coefficient. The detailed training pseudo-code is provided in Appendix C.1.

To justify the selection of the semi-online DPO loss for L_E, we examine its gradient:

$$
\nabla_\theta L_E = -\beta\,\mathbb{E}_{(x,y^+)\sim D_E,\, y^-\sim\pi_{\theta_{\text{old}}}}\Big[\sigma(\beta(r^- - r^+))\cdot\big(\nabla\log\pi_\theta(y^+|x) - \nabla\log\pi_\theta(y^-|x)\big)\Big]. \tag{4}
$$

The three terms in the gradient align precisely with our previously identified vital components: supervised learning, on-policy training, and decaying weights. Theoretically, as demonstrated in SPIN (Chen et al., 2024b), this paradigm is equivalent to an adversarial training process. In each iteration, a discriminator optimizes a loss inspired by Integral Probability Metrics (IPM) (Müller, 1997) to differentiate r^+ and r^-, while the LLM policy acts as the generator with a closed-form solution. Further theoretical details are provided in Appendix B.2.

However, this standard DPO paradigm applies uniform learning strength across all data in D_E, failing to account for varying task difficulty. For simpler questions, deviations from the SFT demonstration should be tolerated, whereas the model should prioritize learning from SFT traces for tasks it cannot solve independently. Thus, we apply an adaptive β strategy based on model solvability. Specifically, for a batch of n rollouts $\{y_1^-, \ldots, y_n^-\}$ with rewards $r \in \{0, 1\}$, we define β for the pair (x, y_i^-) as:

$$\beta(x, y_i^-) = \begin{cases} \beta_1, & \text{if } \forall j \in \{1, 2, \ldots, n\}, r(x, y_j^-) = 0 \\ \beta_2, & \text{if } r(x, y_i^-) = 0 \text{ and } \exists\, j \in \{1, 2, \ldots, n\}, r(x, y_j^-) = 1 \\ \beta_3, & \text{if } r(x, y_i^-) = 1 \end{cases} \tag{5}$$

where $\beta_1, \beta_2, \beta_3$ are constants. This allows us to control the learning strength of different sources of data in a more fine-grained manner, differentiating unsolvable questions (β_1), RLVR-solvable questions (β_2) and correct rollouts (β_3).

Remark 3.1. While DPO is often criticized for its inability to effectively flip preferences (Chen et al., 2024a) and the dominance of the rejected response in the loss term (Cho et al., 2025), these characteristics are not detrimental in our setting. Our objective on D_E is to "regularize" the model toward expert traces and catalyze RLVR performance—akin to an online version of TD3+BC (Fujimoto & Gu, 2021)—rather than enforcing strict preference or total rejection of non-preferred samples, particularly as agent rollouts are often correct.

Remark 3.2. The unique requirement of long-chain reasoning on few-shot data necessitates a distinct choice of β (0.001–0.1) compared to standard DPO practices (0.1–0.2) (Rafailov et al., 2023). This is because the extended sequence lengths and repeated training on sparse data lead to significantly larger log-ratio differences. See Appendix D.6 for a full hyperparameter analysis.

3.3 FEST-GRPO: Mitigation of Gradient Mismatch

While the paradigm described in Sec. 3.2 is effective for few-shot demonstration-guided RLVR, a critical challenge persists: DPO for L_E is a *sequence-level* objective, where the probability inside the log-sigmoid represents the joint probability of the entire response. In contrast, GRPO for L_I is a *token-level* algorithm that applies clipping independently to each token. This structural mismatch often results in significant differences in gradient magnitudes (see Appendix D.4 for an empirical validation), necessitating exhaustive tuning of the coefficient c to balance the gradients.

To mitigate this mismatch, we re-examine the decaying weight and on-policy components of the gradient in Eq. (4): $\mathbb{E}_{x\sim D_E, y^-\sim\pi_{\theta_{\text{old}}}}\left[\beta \cdot \sigma(\beta(r^- - r^+)) \cdot \nabla \log \pi_\theta(y^-|x)\right]$. By comparing this term to the REINFORCE gradient in Sec. 2, it becomes evident that this component is *functionally equivalent to REINFORCE with a negative reward* defined as $-\beta \cdot \sigma(\beta(r^- - r^+)) < 0$. Similarly, the supervised learning component can be interpreted as weighted SFT with a positive weight $\beta \cdot \sigma(\beta(r^- - r^+)) > 0$. This observation provides a novel perspective on the DPO loss L_E:

Semi-online DPO = *REINFORCE with negative reward* + *weighted SFT*.

Under this interpretation, the solution to the gradient mismatch follows intuitively: we **substitute REINFORCE with GRPO**. We denote this variant as **FEST-GRPO**. This method retains the L_I component in Eq. (3) while replacing the DPO-based L_E with a hybrid of weighted SFT and a GRPO loss applied to D_E; see Appendix C.1 for the detailed loss function.

Remark 3.3. While several recent works explore the decomposition of the DPO objective (Wu et al., 2025b; Du et al., 2026; Le Roux et al., 2025), our work establishes a formal equivalence between these specific algorithms, thereby extending the unified post-training framework proposed by HPT (Lv et al., 2025). See Appendix B for detailed comparisons.

Remark 3.4. The efficacy of RL with purely negative rewards is supported by prior theoretical work. Zhu et al. (2025b) demonstrated that negative RL redistributes probability mass toward other feasible solutions, thereby preventing overfitting and facilitating more robust exploration.

4 Experiments

In this section, we evaluate FEST across various settings and benchmarks to investigate the following research questions: (i) Can both the DPO and GRPO variants of FEST enhance RLVR performance

with a minimal number of demonstrations? (Sec. 4.1); (ii) How does FEST scale with the number of shots? (Sec. 4.2); (iii) To what extent is the performance sensitive to the choice of datasets? (Sec. 4.3); and (iv) What are the individual contributions of each component, and can FEST generalize to Out-Of-Distribution (OOD) test sets? (Sec. 4.4).

Training Recipe. We fine-tune the Qwen2.5-Math-1.5B (Yang et al., 2024) model for 600 steps on two NVIDIA GH200 (96GB) GPUs. Following prior work (Yan et al., 2025; Lv et al., 2025; Ma et al., 2026), we utilize the OpenR1-Math-46K-8192 dataset as our primary source. We randomly sample 128 problems with expert reasoning traces to form D_E, while the remaining data serve as the answer-only dataset D_I for reward verification. The number 128 is the batch size from prior works (Lv et al., 2025; Ma et al., 2026), which means an epoch on the dataset can be fitted into a single step. We generate $n = 8$ rollouts per prompt with a temperature of 1.0 and a maximum sequence length of 8192. We employ the AdamW optimizer (Loshchilov & Hutter, 2019) with a cosine learning rate decay from 1×10^{-5} to 5×10^{-6}. The training uses a global batch size of 128 questions from each of D_E and D_I, and a mini-batch size of 512 rollouts (all GPUs combined).

Baselines. We compare FEST against the following baselines: (i) *Vanilla RLVR* with pure GRPO, and *SFT-then-RL* where we first conduct SFT for the model with D_E, then conduct RL on D_I; (ii) *Multi-objective approaches*, such as SRFT (Fu et al., 2026), LUFFY (Yan et al., 2025), and CHORD-ϕ (Zhang et al., 2026b); and (iii) *RL-SFT switching strategies*, including MIFO (Yuan et al., 2025), HPT (Lv et al., 2025), and ReLIFT (Ma et al., 2026). We omit several related methods for specific reasons: SuperRL (Liu et al., 2025b) and DyME (Liu et al., 2026) are functionally identical to HPT in this context, while SASR (Chen et al., 2025c) focuses on simpler tasks and lacks an accessible codebase with readme files. Pure SFT and SPIN (Chen et al., 2024b) were excluded after failing to achieve competitive results (see Appendix D.8). Most baseline results were obtained with our own implementations, with two exceptions: SRFT and MIFO. SRFT is not open-source and does not work with the implementation in the HPT codebase, thus we test their official checkpoint; MIFO does not publish code or checkpoint, and we directly take the results from their paper. Importantly, we always report the better result between the available reimplementation and the released checkpoint trained on full 46K demonstrations, which favors the baselines. See Appendix B for an introduction to the baselines.

Evaluation and Metrics. Following the evaluation protocol in HPT, we assess performance on six prominent mathematical reasoning benchmarks: AIME25 (Li et al., 2024) (30 questions), AMC23 (Li et al., 2024) (40 questions), AIME24 (Li et al., 2024) (30 questions), MATH-500 (Hendrycks et al., 2021) (500 questions), OlympiadBench (He et al., 2024) (674 questions), and Minerva (Lewkowycz et al., 2022) (272 questions). To ensure statistical stability, particularly on smaller benchmarks, we report the mean and standard deviation across 8 rollouts (Avg@8). We also report Pass@8 (the percentage of questions with at least one correct response in 8 trials) to show the model's potential for further RL-driven improvement. Following ReLIFT, we report the result at 600 steps.

4.1 Main Results

Tab. 2 presents the primary results of this study, demonstrating that our proposed method outperforms all established baselines. We highlight three key observations: (i) **Instability in Naive SFT and RL Gradient Integration.** In Tab. 2, we evaluate pure RL, SFT-then-RL, ReLIFT, and HPT appended with a "-G" suffix, indicating that RL was also conducted on the "Gold" few-shot dataset D_E. Surprisingly, both HPT-G and ReLIFT-G perform significantly worse than their variants that utilize only SFT on D_E. An investigation of the training curves reveals that both models suffer from abrupt performance drops on the training set mid-process (see Appendix D.5). We hypothesize that this instability arises from rapid distribution shifts induced by SFT on a dataset already heavily overfitted by RL, reinforcing the necessity of our decaying weight strategy. (ii) **Efficacy of the Pure RL Baseline.** Contrary to findings in prior work (Lv et al., 2025; Ma et al., 2026; Yan et al., 2025), we observe that pure RL remains a formidable baseline when the learning rate is optimized (specifically, increased from 1e-6 to 5e-6[3]). Under this configuration, pure RL achieves performance parity with ReLIFT on the full dataset. (iii) **Consistency in Outperforming RL-G Variants.** FEST is the only method that consistently surpasses both pure RL and RL-G. While RL-G may achieve high nominal accuracy, it suffers from severe overfitting on the limited D_E dataset, resulting in a significantly lower

[3]We also evaluated pure RL using our specific learning rate schedule, which did not yield improvement.

Table 2: Average test set accuracy using a 128-shot configuration (higher is better). The suffix "-G" denotes RL training also performed on the few-shot **G**old dataset (D_E). SRFT and MIFO utilize the **full dataset**, while all other methods are evaluated under the same 128-shot constraint. Results demonstrate that FEST variants not only achieve superior performance, but also represent the only methodology to yield significant gains over the vanilla RL baseline in this sparse-data regime.

Methods	AIME25	AMC23	AIME24	MATH-500	Olympiad	Minerva	Average
SRFT* (ICLR'26)	7.91±2.86	51.25±4.15	10.83±3.23	72.6±1.47	38.58±1.20	29.09±1.15	35.05±1.04
MIFO* (Preprint)	12.0	N/A	19.2	78.8	43.3	N/A	N/A
SFT-then-RL	12.5±2.76	55.94±4.83	12.08±3.71	76.9±1.11	43.73±0.68	31.85±1.31	38.83±1.52
SFT-then-RL-G	8.75±2.86	55.94±5.85	12.5±4.33	73.95±0.53	41.02±0.80	32.40±1.81	37.43±1.58
RL	11.67±4.41	54.37±5.41	17.91±3.31	77.78±1.24	43.25±1.27	33.78±1.49	39.79±1.04
RL-G	14.17±4.33	57.5±5.15	15.83±3.23	78±1.27	44.49±0.86	33.36±1.32	40.55±1.03
LUFFY (NeurIPS'25)	9.58±3.87	55±1.77	14.17±3.23	74.88±1.17	43.49±0.69	30.28±5.50	37.90±0.50
CHORD-ϕ (ICLR'26)	8.75±4.06	53.13±4.64	13.33±2.89	76.03±0.89	42.47±0.75	31.76±1.55	37.58±1.47
HPT (Preprint)	10.83±2.76	55±4.33	13.75±3.09	76.65±0.70	43.14±0.92	33.13±2.03	38.75±1.19
HPT-G	3.33±3.33	46.88±1.65	10.41±1.99	66.7±1.10	34.34±1.31	30.51±1.23	32.02±0.81
ReLIFT (ICLR'26)	11.25±3.30	57.81±2.32	15.83±2.20	79.03±0.87	44.35±0.79	34.79±1.27	40.51±1.03
ReLIFT-G	9.16±3.63	54.06±3.29	14.58±2.32	76.6±1.16	42.28±0.73	32.44±0.86	38.19±0.77
FEST-DPO	11.67±4.41	**59.38±3.25**	**20.41±3.51**	80.48±1.13	46.80±1.01	33.18±1.85	41.98±1.24
FEST-GRPO	**14.58±3.70**	57.81±5.36	18.33±1.67	**81.1±1.27**	**47.00±0.95**	**35.45±1.81**	**42.38±1.26**

Table 3: Pass@8 performance representing the models' upper-bound exploration potential. Notably, while RL-G achieves relatively high average accuracy, its significantly lower Pass@8 indicates limited reasoning diversity and a lower ceiling for further performance refinement. In contrast, FEST variants consistently maintain a superior Pass@8, demonstrating robust potential for subsequent optimization.

Methods	AIME25	AMC23	AIME24	MATH-500	Olympiad	Minerva	Average
SRFT*	23.33	72.5	30	91	63.54	48.90	54.87
SFT-then-RL	30	82.5	23.33	90.2	63.39	48.90	56.39
SFT-then-RL-G	23.33	82.5	30	87.8	59.67	47.43	55.12
RL	**36.67**	85	33.33	91.4	63.1	48.52	59.67
RL-G	33.33	72.5	23.33	90.6	63.69	45.59	54.84
LUFFY	30	82.5	26.67	89.6	61.31	45.22	55.88
HPT	26.67	**87.5**	26.67	91.2	64.73	51.84	58.10
HPT-G	16.67	70	16.67	80.6	53.57	45.22	47.12
ReLIFT	20	85	30	91.2	64.88	49.26	56.72
ReLIFT-G	26.67	77.5	30	89.8	61.76	47.43	55.52
CHORD-ϕ	20	80	26.67	91.6	62.5	50.37	55.19
FEST-DPO	26.67	82.5	**40**	**92.8**	65.17	**53.31**	60.08
FEST-GRPO	**36.67**	**87.5**	33.33	92	**65.77**	51.1	**61.06**

Pass@8 (see Tab. 3). This reduction highlights a diminished potential for further improvement—a pitfall FEST avoids by maintaining high exploration capability.

4.2 Scaling with Shot Counts

To evaluate the scalability of our approach across varying sizes of D_E, we further test our method with 64, 256, and 512 examples. The results, listed in Fig. 2, show that our method can work even with as few as 64 shots. Notably, while FEST-GRPO exhibits superior stability in extreme low-data regimes, FEST-DPO demonstrates more favorable scaling properties, eventually achieving performance comparable to HPT trained on the full 46K SFT dataset. See Appendix D.3 for further analysis of the scalability of FEST-GRPO.

4.3 Consistency of Performance Gain Across Different D_E

To evaluate the robustness of our algorithm across varying D_E, we test FEST on several alternative few-shot datasets (D_E): (i) two additional random 128-shot splits from OpenR1; and (ii) LIMOv2-8192, a subset of 257 examples from LIMOv2 (Ye et al., 2025) with CoT traces under 8,192 tokens.[4] Tab. 4 summarizes the results, demonstrating that FEST consistently achieves performance gains across different D_E. See further training details regarding LIMOv2 in Appendix D.7.

[4]Due to computational constraints, we exclude LIMOv2 examples with longer reasoning traces.

Table 4: Avg@8 performance of FEST across different D_E settings, with the results from Sec. 4.1 denoted as "OpenR1 (dataset 1)". The upper and middle blocks correspond to FEST-DPO and FEST-GRPO, respectively, while the lower block shows RL result for reference. The results indicate that our framework is consistently capable of extracting significant performance improvements from different few-shot demonstrations.

	AIME25	AMC23	AIME24	MATH-500	Olympiad	Minerva	Average
FEST-DPO							
LIMOv2-8192	14.16±3.23	54.69±2.63	17.5±5.71	80.23±1.21	46.28±0.71	34.37±1.34	41.21±1.25
OpenR1 (dataset 1)	11.67±4.41	59.38±3.25	20.41±3.51	80.48±1.13	46.80±1.01	33.18±1.85	41.98±1.24
OpenR1 (dataset 2)	14.58±6.44	53.75±4.67	17.08±3.09	79.7±0.76	46.56±0.57	36.02±1.09	41.28±1.01
OpenR1 (dataset 3)	12.91±2.60	56.88±2.07	18.33±2.89	80.6±0.75	46.24±1.20	35.47±1.25	41.74±0.81
FEST-GRPO							
LIMOv2-8192	14.17±3.23	55±6.25	20.83±2.76	79.55±0.79	46.11±0.90	34.15±1.99	41.63±1.16
OpenR1 (dataset 1)	14.58±3.70	57.81±5.36	18.33±1.67	81.1±1.27	47.00±0.95	35.45±1.81	42.38±1.26
OpenR1 (dataset 2)	12.08±3.70	58.75±4.51	15.83±2.20	80.58±1.10	47.63±0.73	33.08±1.50	41.33±1.46
OpenR1 (dataset 3)	12.08±5.76	58.13±4.28	16.67±3.33	80±1.24	46.88±1.19	33.36±1.18	41.18±1.68
RL (no D_E)	11.67±4.41	54.37±5.41	17.91±3.31	77.78±1.24	43.25±1.27	33.78±1.49	39.79±1.04

4.4 Ablations

Due to the page limit, the details of ablation for hyperparameter β is deferred to Appendix D.6. Generally, we find FEST is reasonably robust to the choice of β, and works best with $\beta \in [0.001, 0.1]$.

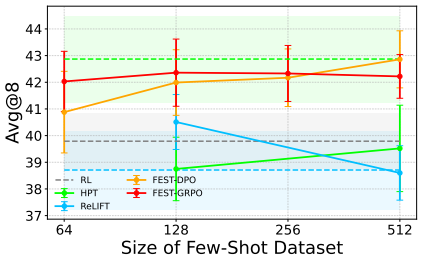


Figure 2: Performance scalability across varying shot counts. Dashed lines represent baseline results with the full 46K SFT dataset. While FEST-GRPO provides higher robustness in the extreme few-shot case (64 shots), FEST-DPO scales better with more data.

4.4.1 Components

In Sec. 3.1, we identified three vital components of few-shot demonstration-guided RLVR. To quantify the contribution of each component to the final performance, we conduct an ablation study under the experimental configuration described in Sec. 4.1. The results, summarized in Tab. 5, demonstrate that optimal performance is achieved only through the synergy of all three components. Specifically, the poor performance of the "RL-G with our weight" variant stems from model divergence: when the model is repeatedly exposed only to negative signals on D_E during late-stage training, it lacks sufficient constructive guidance to keep a coherent policy, eventually leading to collapse.

Table 5: Ablation study of the vital components identified in Sec. 3.1. The results indicate that peak performance relies on the effective collaboration of all three components.

Supervised	On-Policy	Decaying Weight	Equivalent to	Avg@8
×	×	×/✓	RL	39.79±1.04
×	✓	×	RL-G	40.55±1.03
×	✓	✓	RL-G with our weight	28.83±0.92
✓	×/✓	×	Fixed weight SFT+RL on **full dataset**	37.26±1.12
✓	×	✓	RL + few-shot SFT with our weight	39.91±0.67
✓	✓	✓	FEST-GRPO	**42.38±1.26**

4.4.2 Out-of-Distribution Dataset

To further evaluate the cross-domain generalization of FEST, we follow ReLIFT and assess the model trained in Sec. 4.1 on the MMLU-Pro benchmark (Wang et al., 2024b) with over 12,000 problems. The Pass@1 results are listed in Tab. 6. Our results indicate that FEST generalizes robustly to OOD tasks, achieving superior performance compared to all evaluated baselines under this zero-shot setting.

Table 6: Pass@1 performance on the OOD MMLU-Pro benchmark. FEST variants consistently achieve the highest results, highlighting strong generalization capabilities beyond training data.

FEST-DPO (ours)	FEST-GRPO (ours)	RL	RL-G	LUFFY	HPT	ReLIFT	SRFT	CHORD-ϕ	Random
38.68	36.32	34.81	33.18	31.72	32.83	33.99	33.54	35.82	10

5 Related Work

For a more detailed discussion on related work, also see Appendix B, which also discusses the differences of FEST and SPIN. Generally, while the design of FEST resembles SPIN to some degree, FEST differs in several critical ways including sampling paradigm, adaptive use of β, our GRPO-based objective, and especially the novel few-shot demonstration-guided RLVR paradigm.

Demonstration-Guided RLVR. A fundamental limitation of RLVR is its difficulty in surpassing the inherent capabilities of the base model (Zhao et al., 2025a; Cheng et al., 2026; Yue et al., 2025). While RL effectively sharpens the output distribution toward correct answers (Wu et al., 2025a), it often fails to explore solutions beyond the model's initial reasoning capacity. To address this, various hybrid post-training methods integrating SFT with RL have been proposed (Jiang et al., 2026), leveraging expert demonstrations to provide guidance beyond the base model's limits. The most prevalent approaches are reward-based, conducting SFT on problems where the model fails to receive a positive reward (Lv et al., 2025; Ma et al., 2026; Liu et al., 2025b; 2026; Yuan et al., 2025). Other integration strategies include using fixed data ratios (Yan et al., 2025), gradient-norm balancing (Chen et al., 2025c), token probability weighting (Zhang et al., 2026b; Yuan et al., 2025), entropy-based regularization (Fu et al., 2026), and trajectory blending (Liu et al., 2025a; Huang et al., 2025). However, as shown in Tab. 1, these methods typically require substantial SFT data, which is often prohibitively expensive to acquire (Au-Yeung, 2025). In contrast, FEST is specifically designed to enhance RLVR performance using only a few-shot SFT dataset.

Few-Shot LLM Post-Training. In response to the tension between the increasing demand for high-quality reasoning data and its high cost, the community has actively explored post-training LLMs with minimal data via SFT (Muennighoff et al., 2025), DPO (Singh et al., 2026), or RL (Zhu et al., 2025a). Existing few-shot SFT and DPO research generally falls into two categories: (i) *careful curation* via heuristic metrics—such as CoT length (Wei et al., 2022), diversity, and difficulty—exemplified by s1K (Muennighoff et al., 2025) and LIMO (Ye et al., 2025); and (ii) *automated data selection* based on distribution similarity to the base model (Zhang et al., 2025a), gradient-based performance estimation (Wang et al., 2025a), embedding-based diversity (Bukharin et al., 2024; Yang et al., 2025b; Cao et al., 2024) or regression of multiple factors (Cao et al., 2024). On the RL side, researchers have investigated few-shot RLVR (Li et al., 2025b; Fang et al., 2025; Li et al., 2025a; Köksal & Alatan, 2025) using question-only data (Yang et al., 2026) or extremely limited question sets (Wang et al., 2025c). However, none of these works focus on combining RLVR with few-shot SFT data as we do. Furthermore, unlike many prior works, FEST does not require labor-intensive dataset selection or curation to be effective.

6 Conclusion

In this work, we introduced a novel post-training paradigm, few-shot demonstration-guided RLVR, alongside a dedicated training framework, **FEST**. By elucidating the core challenges of training with sparse expert data, we designed a framework based on three vital components: (i) supervised learning from SFT data to provide expert guidance; (ii) on-policy learning with negative advantages to expand the exploration base and ensure robustness; and (iii) a decaying weight strategy to prevent overfitting. Furthermore, by decomposing the DPO objective, we proposed **FEST-GRPO** to bridge the gradient magnitude gap between token-level GRPO and sequence-level DPO. Extensive experiments across multiple benchmarks demonstrate that FEST effectively boosts RLVR performance using as few as 128 randomly sampled demonstrations. We believe this paradigm and its associated algorithms offer a scalable solution to the growing scarcity of high-quality SFT data in the LLM community.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods, supporting data analysis, interpreting results by running AI-generated scripts, editing the paper to improve readability, verifying our mathematical claims and proofs, and identifying/summarizing related literature. We have not used generative AI tools for brainstorming, translation assistance or developing theoretical models / conceptual frameworks. We have reviewed all AI-assisted work. For experiment results, we check via detailed visualization; for the correctness of the code, we check by carefully reading the code; for literature, we check via searching each paper online manually. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This paper fully conforms to ICLR 2027 Code of Ethics statement, and does not raise question regarding it. As for the general broader impact of this work, it introduces a resource-efficient paradigm for post-training LLM agents using minimal expert data. Beyond its technical contributions, our research carries several broader societal implications:

Ethical Data Sourcing and Labor Practices. A significant portion of current LLM development relies on large-scale crowdsourcing for data annotation, which often involves "ghost work" that can lead to the economic and psychological exploitation of workers in the Global South. By demonstrating that performance can be significantly boosted with as few as 128 random samples, FEST reduces the systemic dependence on massive human-labeled datasets. This shift helps mitigate the ethical risks associated with labor exploitation and improves the sustainability of high-performance AI development.

AI Democratization. The high cost of acquiring extensive SFT data creates a barrier to entry for smaller research entities. Our few-shot approach provides a cost-effective alternative, democratizing access to state-of-the-art RLVR techniques for academic and independent researchers who lack industrial-scale resources.

Potential Risks and Mitigations. Despite these benefits, we acknowledge that enhancing the reasoning capabilities of LLMs carries inherent risks. Improved mathematical and logical proficiency could be dual-purposed for malicious activities or accelerate technological displacement in certain labor markets. Furthermore, while RLVR targets "verifiable" rewards, there remains a risk of models developing "reward-hacking" behaviors or unintended biases if the verifiers are not perfectly specified. We believe that addressing these challenges requires a multi-stakeholder approach involving proactive safety alignment, robust policy frameworks, and transparent reporting from the research community.

REPRODUCIBILITY STATEMENT

Training and evaluation details are provided in Sec. 4; implementation details and pseudo-code are provided in Appendix C, extended ablations are provided in Appendix D, and the computational resource usage is reported in Appendix E.

REFERENCES

Maksim Afanasyev and Illarion Iov. Slime: Stabilized likelihood implicit margin enforcement for preference optimization. *arXiv preprint arXiv:2602.02383*, 2026.

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In *ACL*, 2024.

Alon Albalak, Duy Phung, Nathan Lile, Rafael Rafailov, Kanishk Gandhi, Louis Castricato, Anikait Singh, Chase Blagden, Violet Xiang, Dakota Mahan, et al. Big-math: A large-scale, high-quality math dataset for reinforcement learning in language models. *arXiv preprint arXiv:2502.17387*, 2025.

Alexandre Attia and Sharone Dayan. Global overview of imitation learning. *arXiv preprint arXiv:1801.06503*, 2018.

Angel Au-Yeung. The ai startup fueling chatgpt's expertise is now valued at $10 billion. *The Wall Street Journal*, 2025. URL https://www.wsj.com/tech/ai/the-ai-startup-fueling-chatgpts-expertise-is-now-valued-at-10-billion-f1281e56.

Chenjia Bai, Yang Zhang, Shuang Qiu, Qiaosheng Zhang, Kang Xu, and Xuelong Li. Online preference alignment for language models via count-based exploration. In *ICLR*, 2025.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS*, 2015.

Alexander Bukharin, Shiyang Li, Zhengyang Wang, Jingfeng Yang, Bing Yin, Xian Li, Chao Zhang, Tuo Zhao, and Haoming Jiang. Data diversity matters for robust instruction tuning. In *Findings of EMNLP*, 2024.

Yihan Cao, Yanbin Kang, Chi Wang, and Lichao Sun. Instruction mining: Instruction data selection for tuning large language models. In *COLM*, 2024.

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, et al. Minimax-m1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025a.

Angelica Chen, Sadhika Malladi, Lily H Zhang, Xinyi Chen, Qiuyi Zhang, Rajesh Ranganath, and Kyunghyun Cho. Preference learning algorithms do not learn preference rankings. In *NeurIPS*, 2024a.

Hardy Chen, Haoqin Tu, Fali Wang, Hui Liu, Xianfeng Tang, Xinya Du, Yuyin Zhou, and Cihang Xie. Sft or rl? an early investigation into training r1-like reasoning large vision-language models. *TMLR*, 2025b.

Jack Chen, Fazhong Liu, Naruto Liu, Yuhan Luo, Erqu Qin, Harry Zheng, Tian Dong, Haojin Zhu, Yan Meng, and Xiao Wang. Step-wise adaptive integration of supervised fine-tuning and reinforcement learning for task-specific llms. *arXiv preprint arXiv:2505.13026*, 2025c.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. In *ICML*, 2024b.

Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Wayne Xin Zhao, Zhenliang Zhang, and Furu Wei. Reasoning with exploration: An entropy perspective. In *AAAI*, 2026.

Jay Hyeon Cho, JunHyeok Oh, Myunsoo Kim, and Byung-Jun Lee. Rethinking dpo: The role of rejected responses in preference misalignment. In *EMNLP (Findings)*, 2025.

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

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

Karel D'Oosterlinck, Winnie Xu, Chris Develder, Thomas Demeester, Amanpreet Singh, Christopher Potts, Douwe Kiela, and Shikib Mehri. Anchored preference optimization and contrastive revisions: Addressing underspecification in alignment. *TACL*, 2025.

Yuhao Du, Zhuo Li, Pengyu Cheng, Zhihong Chen, Yuejiao Xie, Xiang Wan, and Anningzhe Gao. Rlhf in an sft way: From optimal solution to reward-weighted alignment. *TMLR*, 2026.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *ICML*, 2024.

Wenkai Fang, Shunyu Liu, Yang Zhou, Kongcheng Zhang, Tongya Zheng, Kaixuan Chen, Mingli Song, and Dacheng Tao. Serl: Self-play reinforcement learning for large language models with limited data. In *NeurIPS*, 2025.

Duanyu Feng, Bowen Qin, Chen Huang, Zheng Zhang, and Wenqiang Lei. Towards analyzing and understanding the limitations of dpo: A theoretical perspective. *arXiv preprint arXiv:2404.04626*, 2024.

Yuqian Fu, Tinghong Chen, Jiajun Chai, Xihuai Wang, Songjun Tu, Guojun Yin, Wei Lin, Qichao Zhang, Yuanheng Zhu, and Dongbin Zhao. Srft: A single-stage method with supervised and reinforcement fine-tuning for reasoning. In *ICLR*, 2026.

Scott Fujimoto and Shixiang Shane Gu. A minimalist approach to offline reinforcement learning. In *NeurIPS*, 2021.

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

Kaiyang Guo, Yinchuan Li, and Zhitang Chen. Proximalized preference optimization for diverse feedback types: A decomposed perspective on dpo. In *NeurIPS*, 2025b.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, et al. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*, 2024.

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

Li He, Qiang Qu, He Zhao, Stephen Wan, Dadong Wang, Lina Yao, and Tongliang Liu. Unifying stable optimization and reference regularization in rlhf. In *ICLR*, 2026.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In *NeurIPS*, 2021.

Yichen Huang and Lin F Yang. Winning gold at imo 2025 with a model-agnostic verification-and-refinement pipeline. In *MATH-AI Workshop at NeurIPS*, 2025.

Zeyu Huang, Tianhao Cheng, Zihan Qiu, Zili Wang, Yinghui Xu, Edoardo M Ponti, and Ivan Titov. Blending supervised and reinforcement fine-tuning with prefix sampling. *arXiv preprint arXiv:2507.01679*, 2025.

Chandra Irugalbandara, Ashish Mahendra, Roland Daynauth, Tharuka Kasthuri Arachchige, Jayanaka Dantanarayana, Krisztian Flautner, Lingjia Tang, Yiping Kang, and Jason Mars. Scaling down to scale up: A cost-benefit analysis of replacing openai's llm with open source slms in production. In *ISPASS*, 2024.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Johan Ludwig William Valdemar Jensen. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica*, 1906.

Haitao Jiang, Wenbo Zhang, Jiarui Yao, Hengrui Cai, Sheng Wang, and Rui Song. Supervised fine-tuning versus reinforcement learning: A study of post-training methods for large language models. *arXiv preprint arXiv:2603.13985*, 2026.

Xue Jiang, Yihong Dong, Mengyang Liu, Hongyi Deng, Tian Wang, Yongding Tao, Rongyu Cao, Binhua Li, Zhi Jin, Wenpin Jiao, et al. Coderl+: Improving code generation via reinforcement with execution semantics alignment. *arXiv preprint arXiv:2510.18471*, 2025.

Zixuan Ke, Fangkai Jiao, Yifei Ming, Xuan-Phi Nguyen, Austin Xu, Do Xuan Long, Minzhi Li, Chengwei Qin, Peifeng Wang, Silvio Savarese, et al. A survey of frontiers in llm reasoning: Inference scaling, learning to reason, and agentic systems. *TMLR*, 2025.

Aybora Köksal and A Aydın Alatan. Few-shot vision-language reasoning for satellite imagery via verifiable rewards. In *ICCV 2025 Workshop on Curated Data for Efficient Learning (CDEL)*, 2025.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Hynek Kydlíček and Hugging Face Team. Math-verify: A library for verifying mathematical answers, 2025. URL `https://github.com/huggingface/Math-Verify`. GitHub repository.

Nicolas Le Roux, Marc G Bellemare, Jonathan Lebensold, Arnaud Bergeron, Joshua Greaves, Alexandre Fréchette, Carolyne Pelletier, Eric Thibodeau-Laufer, Sándor Toth, and Sam Work. Tapered off-policy reinforce: Stable and efficient reinforcement learning for large language models. In *NeurIPS*, 2025.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. In *NeurIPS*, 2022.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, et al. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face Repository*, 2024.

Pengyi Li, Matvey Skripkin, Alexander Zubrey, Andrey Kuznetsov, and Ivan Oseledets. Confidence is all you need: Few-shot rl fine-tuning of language models. *arXiv preprint arXiv:2506.06395*, 2025a.

Xuefeng Li, Haoyang Zou, and Pengfei Liu. Limr: Less is more for rl scaling. *arXiv preprint arXiv:2502.11886*, 2025b.

Jiazhen Liu, Yuchuan Deng, and Long Chen. Empowering small vlms to think with dynamic memorization and exploration. In *ICLR*, 2026.

Mingyang Liu, Gabriele Farina, and Asuman Ozdaglar. Uft: Unifying supervised and reinforcement fine-tuning. In *NeurIPS*, 2025a.

Yihao Liu, Shuocheng Li, Lang Cao, Yuhang Xie, Mengyu Zhou, Haoyu Dong, Xiaojun Ma, Shi Han, and Dongmei Zhang. Superrl: Reinforcement learning with supervision to boost language model reasoning. *arXiv preprint arXiv:2506.01096*, 2025b.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *COLM*, 2025c.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Xingtai Lv, Yuxin Zuo, Youbang Sun, Hongyi Liu, Yuntian Wei, Zhekai Chen, Xuekai Zhu, Kaiyan Zhang, Bingning Wang, Ning Ding, et al. Towards a unified view of large language model post-training. *arXiv preprint arXiv:2509.04419*, 2025.

Lu Ma, Hao Liang, Meiyi Qiang, Lexiang Tang, Xiaochen Ma, Zhen Hao Wong, Junbo Niu, Chengyu Shen, Runming He, Yanhao Li, et al. Learning what reinforcement learning can't: Interleaved online fine-tuning for hardest questions. In *ICLR*, 2026.

Qinwei Ma, Jingzhe Shi, Can Jin, Jenq-Neng Hwang, Serge Belongie, and Lei Li. Gradient imbalance in direct preference optimization. *arXiv preprint arXiv:2502.20847*, 2025.

Sadegh Mahdavi, Muchen Li, Kaiwen Liu, Christos Thrampoulidis, Leonid Sigal, and Renjie Liao. Leveraging online olympiad-level math problems for LLMs training and contamination-resistant evaluation. In *ICML*, 2025.

Yingqian Min, Zhipeng Chen, Jinhao Jiang, Jie Chen, Jia Deng, Yiwen Hu, Yiru Tang, Jiapeng Wang, Xiaoxue Cheng, Huatong Song, et al. Imitate, explore, and self-improve: A reproduction report on slow-thinking reasoning systems. *arXiv preprint arXiv:2412.09413*, 2024.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. s1: Simple test-time scaling. In *EMNLP*, 2025.

Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 1997.

OpenAI. On the Navier–Stokes millennium prize problem. https://openai.com/index/navier-stokes-solution/, September 2026a. Published September 8, 2026.

OpenAI. Ten advances in mathematics and theoretical computer science. https://openai.com/index/ten-advances-in-mathematics/, August 2026b. Published August 1, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White. Smaug: Fixing failure modes of preference optimisation with dpo-positive. *arXiv preprint arXiv:2402.13228*, 2024.

Xuchen Pan, Yanxi Chen, Yushuo Chen, Yuchang Sun, Daoyuan Chen, Wenhao Zhang, Yuexiang Xie, Yilun Huang, Yilei Zhang, Dawei Gao, Yaliang Li, Bolin Ding, and Jingren Zhou. Trinity-rft: A general-purpose and unified framework for reinforcement fine-tuning of large language models, 2025a. URL https://arxiv.org/abs/2505.17826.

Yu Pan, Zhongze Cai, Huaiyang Zhong, Guanting Chen, and Chonghuan Wang. What matters in data for dpo? In *NeurIPS*, 2025b.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *ICLR*, 2016.

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

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

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *ICLR*, 2017.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *EuroSys*, 2025.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, et al. Ai models collapse when trained on recursively generated data. *Nature*, 2024.

Anikait Singh, Sheryl Hsu, Kyle Hsu, Eric Mitchell, Stefano Ermon, Tatsunori Hashimoto, Archit Sharma, and Chelsea Finn. Fspo: Few-shot optimization of synthetic preferences personalizes to real users. In *ICLR*, 2026.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2024.

Xiaofeng Tan, Hongsong Wang, Xin Geng, and Pan Zhou. Sopo: Text-to-motion generation using semi-online preference optimization. In *NeurIPS*, 2025.

Binghai Wang, Rui Zheng, Lu Chen, Yan Liu, Shihan Dou, Caishuang Huang, Wei Shen, Senjie Jin, Enyu Zhou, Chenyu Shi, et al. Secrets of rlhf in large language models part ii: Reward modeling. *arXiv preprint arXiv:2401.06080*, 2024a.

Jingtan Wang, Xiaoqiang Lin, Rui Qiao, Pang Wei Koh, Chuan-Sheng Foo, and Bryan Kian Hsiang Low. NICE data selection for instruction tuning in LLMs with non-differentiable evaluation metric. In *ICML*, 2025a.

Yibo Wang, Hai-Long Sun, Qing-Guo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, and Lijun Zhang. Triplets better than pairs: Towards stable and effective self-play fine-tuning for llms. In *NeurIPS*, 2025b.

Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Lucas Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, Weizhu Chen, Shuohang Wang, Simon Shaolei Du, and Yelong Shen. Reinforcement learning for reasoning in large language models with one training example. In *NeurIPS*, 2025c.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *NeurIPS*, 2024b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 1992.

Fang Wu, Weihao Xuan, Ximing Lu, Mingjie Liu, Yi Dong, Zaid Harchaoui, and Yejin Choi. The invisible leash: Why rlvr may or may not escape its origin. *arXiv preprint arXiv:2507.14843*, 2025a.

Yihong Wu, Liheng Ma, Lei Ding, Muzhi Li, Xinyu Wang, Kejia Chen, Zhan Su, Zhanguang Zhang, Chenyang Huang, Yingxue Zhang, et al. It takes two: Your grpo is secretly dpo. In *Efficient Reasoning Workshop at NeurIPS*, 2025b.

Wei Xiong, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. Iterative preference learning from human feedback: Bridging theory and practice for rlhf under kl-constraint. In *ICML*, 2024.

Jing Xu, Andrew Lee, Sainbayar Sukhbaatar, and Jason Weston. Some things are more cringe than others: Iterative preference optimization with the pairwise cringe loss. *arXiv preprint arXiv:2312.16682*, 2023.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. In *NeurIPS*, 2025.

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

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

Shenzhi Yang, Guangcheng Zhu, Xing Zheng, Yingfan Ma, Zhongqi Chen, Bowen Song, Weiqiang Wang, Junbo Zhao, Gang Chen, and Haobo Wang. Trapo: A semi-supervised reinforcement learning framework for boosting llm reasoning. In *ICLR*, 2026.

Yuming Yang, Yang Nan, Junjie Ye, Shihan Dou, Xiao Wang, Shuo Li, Huijie Lv, Tao Gui, Qi Zhang, and Xuan-Jing Huang. Measuring data diversity for instruction tuning: A systematic analysis and a reliable metric. In *ACL*, 2025b.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. Limo: Less is more for reasoning. In *COLM*, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Xiangchi Yuan, Xiang Chen, Tong Yu, Dachuan Shi, Can Jin, Wenke Lee, and Saayan Mitra. Mitigating forgetting between supervised and reinforcement learning yields stronger reasoners. *arXiv preprint arXiv:2510.04454*, 2025.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? In *NeurIPS*, 2025.

Dylan Zhang, Qirun Dai, and Hao Peng. The best instruction-tuning data are those that fit. In *NeurIPS*, 2025a.

Ming Zhang, Yawei Wang, Xiaoteng Ma, Li Xia, Jun Yang, Zhiheng Li, and Xiu Li. Wasserstein distance guided adversarial imitation learning with reward shape exploration. In *DDCLS*, 2020.

Shaokun Zhang, Yi Dong, Jieyu Zhang, Jan Kautz, Bryan Catanzaro, Andrew Tao, Qingyun Wu, Zhiding Yu, and Guilin Liu. Nemotron-research-tool-n1: Exploring tool-using language models with reinforced reasoning. In *ICLR*, 2026a.

Wenhao Zhang, Yuexiang Xie, Yuchang Sun, Yanxi Chen, Guoyin Wang, Yaliang Li, Bolin Ding, and Jingren Zhou. On-policy rl meets off-policy experts: Harmonizing supervised fine-tuning and reinforcement learning via dynamic weighting. In *ICLR*, 2026b.

Yifan Zhang. StackMathQA: A curated collection of 2 million mathematical questions and answers sourced from stack exchange. Technical report, ASI Research, 2024. Released January 11, 2024.

Yuheng Zhang, Wenlin Yao, Changlong Yu, Yao Liu, Qingyu Yin, Bing Yin, Hyokun Yun, and Lihong Li. Improving sampling efficiency in rlvr through adaptive rollout and response reuse. *arXiv preprint arXiv:2509.25808*, 2025b.

Rosie Zhao, Alexandru Meterez, Sham Kakade, Cengiz Pehlevan, Samy Jelassi, and Eran Malach. Echo chamber: Rl post-training amplifies behaviors learned in pretraining. In *COLM*, 2025a.

Xuandong Zhao, Will Cai, Tianneng Shi, David Huang, Licong Lin, Song Mei, and Dawn Song. Improving llm safety alignment with dual-objective optimization. In *ICML*, 2025b.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

Speed Zhu, Jianwei Cai, Guang Chen, Lulu Wu, Saiyong Yang, and Wiggin Zhou. Drive: Data curation best practices for reinforcement learning with verifiable reward in competitive code generation. *arXiv preprint arXiv:2511.06307*, 2025a.

Xinyu Zhu, Mengzhou Xia, Zhepei Wei, Wei-Lin Chen, Danqi Chen, and Yu Meng. The surprising effectiveness of negative reinforcement in llm reasoning. In *NeurIPS*, 2025b.

APPENDIX FOR BOOSTING REINFORCEMENT LEARNING WITH VERIFIABLE REWARDS VIA RANDOMLY SELECTED FEW-SHOT GUIDANCE

This appendix is organized as follows. First, in Sec. A, we discuss the limitation and future works. Next, we discuss more related works in Sec. B, provide implementation details of our method and baselines in Sec. C, and show more experimental results in Sec. D. Finally, we report our computational resource in Sec. E and licenses in Sec. F.

A LIMITATIONS AND FUTURE WORKS

Due to computational resource constraints in academia, our evaluation was mostly conducted with a 1.5B parameter model, and we focused on math reasoning following most prior works in our line (SRFT, LUFFY, ReLIFT, HPT and MIFO). Investigating how FEST scales to larger model architectures and generalizes to broader task domains, such as code generation and tool-use agents, remains a significant and promising direction for future research.

B EXTENDED RELATED WORKS

B.1 MORE RELATED WORKS

(Semi-)Online DPO and Self-Play Preference Learning. To enable models to learn from self-generated data and mitigate exposure bias (Bengio et al., 2015), several studies have explored iterative (Xu et al., 2023; Min et al., 2024; Wang et al., 2025b) or online DPO (Guo et al., 2024; Xiong et al., 2024; Bai et al., 2025), where the preference dataset is updated during training using the agent's own rollouts. FEST adopts a similar principle but operates as a *semi-online* DPO algorithm, where only the non-preferred responses are generated on-the-fly (Pan et al., 2025b). Under this definition, the closest works to ours are SoPo (Tan et al., 2025) and SPIN (Chen et al., 2024b), both of which utilize static SFT data as preferred examples and agent rollouts as non-preferred ones. However, SoPo is specifically tailored for human motion generation with diffusion models. While the design of FEST resembles SPIN to some degree, FEST differs in several critical ways: (i) while SPIN targets RLHF with large SFT datasets, FEST is optimized for few-shot demonstration-guided RLVR; (ii) unlike the iterative nature of SPIN or STILL-2 (Min et al., 2024), FEST utilizes a truly online sampling paradigm[5]; (iii) inspired by RLVR research (Lv et al., 2025), we introduce adaptive β values to handle varying task difficulties and long-horizon reasoning—factors not considered in the SPIN framework; and (iv) we propose FEST-GRPO, which leverages a token-level clipped gradient that is fundamentally different from the sequence-level SPIN objective.

Unifying RL and SFT in LLM Post-Training. The conventional paradigm for modern LLM post-training typically follows a decoupled, two-stage process: an initial "cold start" via SFT followed by RLVR to enhance reasoning capabilities and mitigate exposure bias (Guo et al., 2025a). While several state-of-the-art models employ multiple alternating stages of SFT and RL to handle diverse tasks—such as distinguishing between improving core reasoning ability and integration of Mixture-of-Experts (MoE) (Shazeer et al., 2017) routing (Yang et al., 2025a; Guo et al., 2025a)—the objectives of these stages remain largely distinct. However, recent findings have challenged this sequential approach, suggesting it may not consistently outperform pure RL due to its tendency to disrupt established behavioral patterns or induce overfitting (Zhang et al., 2026a; Chen et al., 2025b; Zhang et al., 2026b). Consequently, a burgeoning line of research has shifted toward a unified framework where SFT and RL are conducted jointly (Yan et al., 2025; Lv et al., 2025; Ma et al., 2026; Zhang et al., 2026b; Fu et al., 2026). Our work aligns with this movement but distinguishes itself by significantly reducing the dependency on extensive SFT datasets.

DPO Gradient Decomposition. Since the introduction of DPO, numerous studies have identified the underlying dual-nature of its objective: a combination of a weighted SFT loss and a negative reinforcement learning signal (Wu et al., 2025b; Le Roux et al., 2025; Du et al., 2026). This gradient decomposition has become a cornerstone for analyzing and refining DPO, leading to innovations

[5]The verl implementation of SPIN (`https://verl.readthedocs.io/en/latest/algo/spin.html`) is online but deviates significantly from the original framework by omitting SFT data.

such as the elimination of explicit preference pairs (Ethayarajh et al., 2024), the introduction of "positive anchoring" to ensure the absolute log-probability of preferred responses increases (Pal et al., 2024; D'Oosterlinck et al., 2025; Feng et al., 2024; Guo et al., 2025b; Ma et al., 2025; Afanasyev & Iov, 2026), more targeted safety alignment (Zhao et al., 2025b), and the development of broader unified RLHF frameworks (Du et al., 2026; He et al., 2026). Despite these advancements, most existing works treat this structural decomposition as a conceptual analogy to general RL rather than establishing a formal mathematical equivalence between online DPO and specific algorithms like REINFORCE (Williams, 1992). The closest efforts in this direction include TOPR (Le Roux et al., 2025), which discusses off-policy REINFORCE, and 2-GRPO (Wu et al., 2025b), which establishes equivalence between GRPO with two rollouts and online DPO. However, these studies focus on algorithmic variants that remain distinct from the semi-online DPO formulation proposed in this work.

B.2 Semi-Online DPO and Adversarial Training

Preliminary: Integral Probability Metrics (IPM). Integral Probability Metrics (IPMs) (Müller, 1997) provide a general framework for measuring the statistical distance between two probability distributions. Formally, given two distributions p and q, the IPM is defined as:

$$\sup_{f \in \mathcal{F}} \left| \mathbb{E}_{x \sim p}[f(x)] - \mathbb{E}_{x \sim q}[f(x)] \right|, \tag{6}$$

where $\mathcal{F}$ denotes a class of real-valued measurable functions integrable under p and q. The choice of $\mathcal{F}$ determines the discrepancy; for instance, if $\mathcal{F}$ is the set of 1-Lipschitz functions, the IPM recovers the 1-Wasserstein distance.

Following the principles of Wasserstein inverse RL (Zhang et al., 2020), we consider an adversarial training process with alternating updates of a generator (the policy π) and a discriminator (the reward function f). At each step, the discriminator seeks to maximize the discrepancy between the fixed sampling policy π_{old} and the expert behavioral policy π^E represented by D_E. This yields the following discriminator objective:

$$\max_f \mathbb{E}_{x \sim D_E} \left[\mathbb{E}_{y \sim \pi^E(\cdot|x)}[f(x,y)] - \mathbb{E}_{y' \sim \pi_{\text{old}}(\cdot|x)}[f(x,y')] \right], \tag{7}$$

where the absolute value is omitted assuming a symmetric critic class (i.e., $f \in \mathcal{F} \implies -f \in \mathcal{F}$). However, this linear objective can be unbounded above for unconstrained f, potentially leading to instability during training. Therefore, we introduce a non-linear surrogate using $g(\cdot)$. Specifically, we let $g(z) = -\log(1 + e^{-z})$ be the log-sigmoid function, whose negative is the logistic loss used in SPIN (Chen et al., 2024b), leading to an objective bounded above by zero:

$$\max_f \mathbb{E}_{x \sim D_E} g\left(\mathbb{E}_{y \sim \pi^E(\cdot|x)}[f(x,y)] - \mathbb{E}_{y' \sim \pi_{\text{old}}(\cdot|x)}[f(x,y')] \right). \tag{8}$$

For the generator update, we hold f fixed and maximize its expectation under the new policy π, with a KL regularization term anchored at the reference policy π_{ref}:

$$\max_\pi \mathbb{E}_{x \sim D_E} \left[\mathbb{E}_{y' \sim \pi(\cdot|x)}[f(x,y')] - \beta \mathrm{KL}\left(\pi(\cdot|x) \| \pi_{\text{ref}}(\cdot|x)\right) \right]. \tag{9}$$

For $\beta > 0$, assuming π_{ref} has full support and the normalizing constant is finite, the optimal policy over the unrestricted policy space is given by the Gibbs distribution: $\pi^*(y'|x) \propto \pi_{\text{ref}}(y'|x) \exp(f(x,y')/\beta)$. Motivated by this reward–policy relationship, we parameterize f using the trainable policy π as $f(x,y') = \beta \log \frac{\pi(y'|x)}{\pi_{\text{ref}}(y'|x)} + c(x)$. Substituting this parameterization into Eq. (8), the terms $c(x)$ cancel, yielding:

$$\min_\pi \; -\mathbb{E}_{x \sim D_E} \log \sigma \left(\beta \mathbb{E}_{y \sim \pi^E(\cdot|x)} \left[\log \frac{\pi(y|x)}{\pi_{\text{ref}}(y|x)} \right] - \beta \mathbb{E}_{y' \sim \pi_{\text{old}}(\cdot|x)} \left[\log \frac{\pi(y'|x)}{\pi_{\text{ref}}(y'|x)} \right] \right). \tag{10}$$

Finally, by invoking Jensen's inequality (Jensen, 1906) and the concavity of the log-sigmoid function, we obtain a tractable upper bound on the loss in Eq. (10) for each π. Minimizing this upper bound gives:

$$\min_\pi \; -\mathbb{E}_{(x,y) \sim D_E,\, y' \sim \pi_{\text{old}}(\cdot|x)} \log \sigma \left(\beta \log \frac{\pi(y|x)}{\pi_{\text{ref}}(y|x)} - \beta \log \frac{\pi(y'|x)}{\pi_{\text{ref}}(y'|x)} \right), \tag{11}$$

which corresponds precisely to our semi-online DPO loss. Here, π_{old} and π_{ref} are held fixed during each policy update, while π is optimized through the log-probability ratios.

Remark B.1. Our derivation complements SPIN's original framework, where the pairwise objective $\mathbb{E}_{(x,y)\sim D_E,\, y'\sim\pi_{\text{old}}(\cdot|x)}[g(f(x,y)-f(x,y'))]$ is introduced as a non-linear generalization of the IPM-inspired objective. Our derivation explicitly establishes, through Jensen's inequality, that the loss in Eq. (11) upper-bounds the log-sigmoid-transformed discrepancy loss in Eq. (10), rather than the original IPM itself. The Jensen bound does not require a particular function class, provided the relevant expectations are finite; the reward parameterization above nevertheless induces a function class determined by the policy model.

B.3 Unifying Post-Training Framework

HPT (Lv et al., 2025) establishes a unified post-training framework that synthesizes the gradients of various algorithms into a generalized policy gradient estimator for the policy π_θ:

$$\nabla\pi_\theta \cdot \mathbb{1}_{\text{stable}} \frac{1}{\pi_0} A, \tag{12}$$

where $\mathbb{1}_{\text{stable}}$ denotes a stability mask (e.g., clipping in GRPO) and π_0 represents the weighting distribution used for importance sampling (e.g., π_θ for SFT and $\pi_{\theta_{\text{old}}}$ for GRPO). Despite its breadth, the original HPT framework omits Direct Preference Optimization (DPO), a pivotal tool in the post-training pipeline. Based on the connection between DPO and REINFORCE established in Sec. 3.3, we extend this framework to incorporate DPO, thereby completing the taxonomy. Specifically, Tab. 7 presents the extended framework for optimizing over a trajectory τ (a prompt-response pair), with our proposed additions highlighted in red.

Table 7: The extended unified post-training framework. Here, $\pi_{\theta_{\text{old}}}$ refers to the policy prior to the current update; $\mathbb{1}_{\text{clip}}$ denotes token-level clipping in GRPO/PPO, while $\mathbb{1}_{\text{CIS-mask}}$ and $\mathbb{1}_{\text{Seq-Clip}}$ refer to the specialized clipping strategies proposed in their respective papers. For FEST-GRPO, the gradient is defined as $\nabla\pi_\theta(\tau)\cdot\frac{A_1\mathbb{1}_{\text{clip}}}{\pi_{\theta_{\text{old}}}(\tau)} + c\left(\frac{A_2\nabla\pi_\theta(\tau)}{\pi_\theta(\tau)} + \frac{A_2\nabla\pi_\theta(\tau)\mathbb{1}_{\text{clip}}}{\pi_{\theta_{\text{old}}}}\right)$.

Algorithm	π_0	Advantage A	Token Gradient
SFT	π_θ	1	$\nabla\pi_\theta(\tau)\cdot\frac{1}{\pi_0(\tau)}$
PPO (Schulman et al., 2017)	$\pi_{\theta_{\text{old}}}$	GAE (Schulman et al., 2016)	$\nabla\pi_\theta(\tau)\cdot\frac{A\mathbb{1}_{\text{clip}}}{\pi_0(\tau)}$
GRPO (Shao et al., 2024)	$\pi_{\theta_{\text{old}}}$	$\frac{R(\tau_i)-\text{mean}(R(\tau_i))}{\text{std}(R(\tau_i))}$	$\nabla\pi_\theta(\tau)\cdot\frac{A\mathbb{1}_{\text{clip}}}{\pi_0(\tau)}$
REINFORCE (Ahmadian et al., 2024)	π_θ	±1	$\nabla\pi_\theta(\tau)\cdot\frac{A}{\pi_0(\tau)}$
CISPO (Chen et al., 2025a)	$\pi_{\theta_{\text{old}}}$	$\frac{R(\tau_i)-\text{mean}(R(\tau_i))}{\text{std}(R(\tau_i))}$	$\nabla\pi_\theta(\tau)\cdot\frac{A\mathbb{1}_{\text{CIS-mask}}}{\pi_0(\tau)}$
GSPO (Zheng et al., 2025)	$\pi_\theta\cdot(\frac{\pi_{\theta_{\text{old}}}}{\pi_\theta})^{1/\lvert\tau\rvert}$	$\frac{R(\tau_i)-\text{mean}(R(\tau_i))}{\text{std}(R(\tau_i))}$	$\nabla\pi_\theta(\tau)\frac{A\mathbb{1}_{\text{Seq-Clip}}}{\pi_0(\tau)}$
SRFT (Fu et al., 2026)	1	$\frac{R(\tau_i)-\text{mean}(R(\tau_i))}{\text{std}(R(\tau_i))}$ (incl. SFT trace)	$\nabla\pi_\theta(\tau)\cdot A$
LUFFY (Yan et al., 2025)	1	$\frac{R(\tau_i)-\text{mean}(R(\tau_i))}{\text{std}(R(\tau_i))}$ (incl. SFT trace)	$\nabla\pi_\theta(\tau)\cdot A$
DPO (Rafailov et al., 2023)	π_θ	$\pm\beta\cdot\sigma(\beta(r^- - r^+))$	$\nabla\pi_\theta(\tau)\cdot\frac{A}{\pi_0(\tau)}$
FEST (ours)	π_θ, $\pi_{\theta_{\text{old}}}$	$A_1 = R(\tau_i) - \text{mean}(R(\tau_i))$, $A_2 = \pm\beta\cdot\sigma(\beta(r^- - r^+))$	$\nabla\pi_\theta(\tau)\cdot\frac{A_1\mathbb{1}_{\text{clip}}}{\pi_{\theta_{\text{old}}}(\tau)} +$ $c\cdot\frac{A_2\nabla\pi_\theta(\tau)}{\pi_\theta(\tau)}$

C Implementation Details

C.1 Pseudo-Code

Alg. 1 outlines the implementation of our proposed FEST algorithm. For simplicity, we omit the inner loop for multiple epochs per step, as all experiments in this study are conducted using a single epoch. We define the indicator function $\mathbb{I}[\text{condition}]$ as 1 if the condition holds and 0 otherwise. Specifically,

the loss function for FEST-GRPO is

$$L = c \cdot L_E + L_I, \text{ where}$$

$$\begin{aligned} L_I = \mathbb{E}_{x\sim D_I,\, y_i\sim\pi_{\theta_{\text{old}}}(\cdot|x)} \frac{-1}{nM} \sum_{i=1}^{n}\sum_{j=1}^{|y_i|} \min \Bigg[& \frac{\pi_\theta(y_{i,j} \mid x, y_{i,<j})}{\pi_{\theta_{\text{old}}}(y_{i,j} \mid x, y_{i,<j})} A_i, \\ & \text{clip}\left(\frac{\pi_\theta(y_{i,j} \mid x, y_{i,<j})}{\pi_{\theta_{\text{old}}}(y_{i,j} \mid x, y_{i,<j})}, 1-\epsilon_1, 1+\epsilon_2\right) A_i \Bigg], \\ L_E = \mathbb{E}_{(x,y^+)\sim D_E,\, y_i\sim\pi_{\theta_{\text{old}}}(\cdot|x)} \frac{-1}{nM} \sum_{i=1}^{n} \Bigg[& \sum_{j=1}^{|y_i|} \min\left(\frac{\pi_\theta(y_{i,j} \mid x, y_{i,<j})}{\pi_{\theta_{\text{old}}}(y_{i,j} \mid x, y_{i,<j})} A'_i, \right. \\ & \left. \text{clip}\left(\frac{\pi_\theta(y_{i,j} \mid x, y_{i,<j})}{\pi_{\theta_{\text{old}}}(y_{i,j} \mid x, y_{i,<j})}, 1-\epsilon_1, 1+\epsilon_2\right) A'_i\right) \\ & - A'_i \log \pi_\theta(y^+ \mid x) \Bigg]. \end{aligned} \tag{13}$$

In this equation, $A'_i = -sg\left[\beta_i \sigma(\beta_i \log \frac{\pi(y_i|x)}{\pi_{\text{ref}}(y_i|x)} - \beta_i \log \frac{\pi(y^+|x)}{\pi_{\text{ref}}(y^+|x)})\right]$, with sg being the stop-gradient operator. L_I is the same as FEST-DPO in Eq. (3), where A_i is the RL advantage.

Algorithm 1 FEST

Input: LLM with policy π, number of training steps T, batch size B, number of rollouts per question N, minibatch size B_{mini}, few-shot SFT dataset D_E, answer-only RL dataset D_I, hyperparameter $\beta_1, \beta_2, \beta_3$, loss coefficient $c > 0$

Set $\pi_{\text{ref}} = \pi$

for $t = 1, 2, \ldots, T$ **do**

  Sample questions $x_1^E, \ldots, x_B^E$ from D_E and $x_1^I, \ldots, x_B^I$ from D_I

  Set $\pi_{\theta_{\text{old}}} = \pi$

  for $i = 1, 2, \ldots, B$ **do**

    for $j = 1, 2, \ldots, N$ **do**

      Sample $y_{i,j}^E \sim \pi_{\theta_{\text{old}}}(\cdot|x_i^E)$, $y_{i,j}^I \sim \pi_{\theta_{\text{old}}}(\cdot|x_i^I)$

    end for

    Use verifier to get reward $r_{i,j}^E(x_i^E, y_{i,j}^E)$ and $r_{i,j}^I(x_i^I, y_{i,j}^I)$

    Calculate advantage on I: $A_{i,j}^I = r_{i,j}^I(x_i^I, y_{i,j}^I) - \frac{1}{N}\sum_{k=1}^{N} r^I(x_i^I, y_{i,k}^I)$

    Calculate mask on E for determining β: $M_{i,j}^{\text{correct}} = \mathbb{I}\left[r_{i,j}^E(x_i^E, y_{i,j}^E) = 1\right]$, $M_i^{\text{solvable}} = \mathbb{I}\left[\sum_{k=1}^{N} r_{i,k}^E(x_i^E, y_{i,k}^E) > 0\right]$

  end for

  for $t_{\text{mini}} = 1, 2, \ldots, 2 \times B \times N / B_{\text{mini}}$ **do**

    Sample $B_{\text{mini}}/2$ rollout from D_E and D_I respectively

    Calculate L_I, the GRPO loss on I using Eq. (1)

    Determine β: $\beta_{i,j}^E = \left(\beta_1 \cdot \left(1 - M_i^{\text{solvable}}\right) + \beta_2 \cdot M_i^{\text{solvable}}\right) \cdot \left[1 - M_{i,j}^{\text{correct}}\right] + \beta_3 \cdot \left[M_{i,j}^{\text{correct}}\right]$

    Calculate L_E using Eq. (3) for FEST-DPO, or according to Eq. (13) for FEST-GRPO

    Update π with loss $L = c \cdot L_E + L_I$

  end for

end for

C.2 Hyperparameters

Tab. 8 lists the hyperparameters we used. Generally, hyperparameters are designed to match those of ReLIFT (Ma et al., 2026). The few-shot dataset is uniformly randomly sampled from the OpenR1-Math-46K-8192 dataset.

The reported hyperparameter does not require extensive tuning efforts. Thanks to our quantitative analysis on the ratio indicator z (Appendix D.6), calibrating the hyperparameters typically requires

no more than 1-2 additional partial runs. For c in FEST-DPO, one can run approximately 200 gradient steps, compare the DPO and GRPO gradient norms as in Figure 4, and set c to the ratio between the GRPO and DPO gradient norms, rounded to the nearest order of magnitude. For β, our FEST-DPO ablation shows that values between 0.001 and 0.1 generally work well. In practice, one can run FEST once or twice using our default values, inspect z as in Figure 6, and adjust β so that z generally lies between 1 and 10.

Table 8: Experimental Hyperparameters. Due to occasional deadlock issues encountered with the standard `math-verify` library during distributed training, we utilized a modified reward verifier implementation from `https://github.com/RLHFlow/Reinforce-Ada`. To validate such verifier, we applied it to the ground-truth answers in the OpenR1-Math-46K-8192 D_I training set, and obtained an average reward of 0.996. We also manually inspected model outputs and found no evidence of reward hacking.

Hyperparameter	Value	Note
Learning rate	1e-5 decaying to 5e-6	
Learning rate scheduler	Cosine	
Max question length	1024	in tokens
Max response length	8192	in tokens
Batch size	128	questions sampled from each dataset
Minibatch size	512	rollouts per gradient step
Rollouts	8	samples per question
KL regularizer	N/A	
Entropy coefficient	0.0001	
Grad clip	1.0	
Max grad	80	step beyond this norm will be discarded
GRPO clip ratio	(-0.2,0.3)	
Training temperature	1	
Eval temperature	0.6	
$(\beta_1, \beta_2, \beta_3)$	$(0.1, 0.01, 0.01)$ (FEST-DPO); $(0.005, 0.01, 0.05)$ (FEST-GRPO)	
c	0.01; 1 (FEST-GRPO)	loss coefficient for D_E
Verifier	math-verify 0.8	
Max pos embedding	16384	
RoPE (Su et al., 2024) θ	40000	following ReLIFT (Ma et al., 2026)

C.3 Baselines

We evaluate FEST against several state-of-the-art baselines. All implemented methods, including our own, are integrated into a unified codebase derived from ReLIFT[6]. We utilize `verl` 0.4.0 (Sheng et al., 2025) as our primary reinforcement learning backbone and `vllm` 0.8.4 (Kwon et al., 2023) for rollout generation.[7]

- **HPT (Lv et al., 2025):** HPT employs a selective training policy that applies SFT to samples where the average rollout reward falls at or below a specific threshold (set to 0 for Qwen models) and RL otherwise. Our results are based on an implementation with a learning rate of 5×10^{-6} (following their work) and generation configurations aligned with the original study. Notably, in our few-shot setting, we found that restricting HPT to SFT-only on the 128-shot gold dataset D_E yielded better performance than combining it with RL (see Tab. 2).
- **ReLIFT (Ma et al., 2026):** Similar to HPT, ReLIFT identifies samples with 0 reward but stores them in a buffer, executing a standalone SFT step once the buffer reaches a threshold size. We implement ReLIFT with a learning rate of 5×10^{-6}, an adjustment from the 1×10^{-6} used in the original paper and the Unify-Post-Training repository, as we observed that this higher rate significantly enhanced performance in our experiments.
- **LUFFY (Yan et al., 2025):** LUFFY interleaves SFT and RL gradients during every update step by treating SFT demonstrations as rollouts within a regularized importance sampling

[6] `https://github.com/TheRoadQaQ/ReLIFT`

[7] Due to the ARM-based architecture of our hardware, `vllm` was compiled from source.

framework. Consistent with our other evaluations, we found that increasing the learning rate to 5×10^{-6} improved results, though further increases to 1×10^{-5} induced training instability. We maintain consistency in online sample counts by using 8 online rollouts and 1 offline SFT sample per update.

- **CHORD (Zhang et al., 2026b):** CHORD conducts SFT on the SFT dataset and RL on the answer-only dataset simultaneously with decaying weight. There are two variants: CHORD-μ naively applies a decaying weight on SFT, while CHORD-ϕ applies a weight of $\phi(x) = p_x(1 - p_x)$ on each token, where p_x is the probability of the sampled token x. Note, while CHORD provides code base, it is part of a highly integrated framework called Trinity-RFT (Pan et al., 2025a), and limits training data to problems with integer answer as it lacks a comprehensive verifier like `math-verify` (Kydlíček & Team, 2025). Thus, the code base is not directly usable and we implement it in our code base. As CHORD-ϕ outperforms CHORD-μ in their paper, we implement and test CHORD-ϕ with a learning rate of 5e-6.
- **SRFT (Fu et al., 2026):** Similar to CHORD, SRFT mixes RL and SFT loss from both datasets for every gradient step with entropy-based weights. As SRFT does not release code (the link to their repository `https://anonymous.4open.science/w/SRFT2025/` has expired) but has available checkpoints based on Qwen2.5-Math-1.5B, we directly download their checkpoint (`https://huggingface.co/Yuqian-Fu/SRFT-Qwen2.5-Math-1.5B`) and test the results, which is far worse than our method. The code base of HPT also implemented SRFT; however, SRFT training constantly collapse with their implementation. Based on results from ReLIFT and LUFFY, we hypothesize that the performance of the official checkpoint could be limited due to conservative learning rate.
- **MIFO (Yuan et al., 2025):** MIFO utilizes a buffer-based SFT approach similar to ReLIFT but freezes model parameters that exhibit high sensitivity during RL. In the absence of an open-source codebase or public checkpoints, we cite the performance metrics directly from the original paper for the Qwen2.5-Math-1.5B base model.
- **SFT-then-RL:** Though all methods above prove to outperform SFT-then-RL, for the completeness of evaluation, we also test this baseline in Tab. 2. As our expert dataset D_E is so small that it can fit in a single batch, we train SFT on D_E for 40 steps, which translates to 40 epochs.

D More Experimental Results

D.1 Main Result with Multiple Seeds

We verify the performance of our method across multiple seeds in Tab. 9 by running 3 seeds of the same setting for FEST-DPO and FEST-GRPO. The results show that the performance of our method is consistent across multiple seeds.

Table 9: Avg@8 of our method under the same setting of Tab. 2 across multiple seeds, where the standard deviation is the result across multiple seeds.

	AIME25	AMC23	AIME24	MATH-500	Olympiad	Minerva	Average
FEST-DPO							
Seed 1	11.67	59.38	20.41	80.48	46.80	33.18	41.98
Seed 2	15.41	62.81	15.42	81.08	45.52	32.35	42.09
Seed 3	12.92	57.81	18.33	79.4	45.96	34.6	41.50
Avg. of 3 runs	13.33±1.90	60±2.55	18.05±2.51	80.32±0.85	46.09±0.65	33.37±1.14	41.85±0.31
FEST-GRPO							
Seed 1	14.58	57.81	18.33	81.1	47.00	35.45	42.38
Seed 2	17.08	55.94	14.17	80.25	46.34	35.98	41.62
Seed 3	17.92	59.06	15.83	79.8	47.56	33.13	42.21
Avg. of 3 runs	16.52±1.73	57.60±1.57	16.11±2.09	80.38±0.66	46.97±0.61	34.85±1.52	42.07±0.40
RL (no D_E)	11.67	54.37	17.91	77.78	43.25	33.78	39.79

D.2 Scaling Our Algorithm to Larger Models

In order to test how FEST works with larger models, we further train FEST-DPO and FEST-GRPO on Qwen2.5-Math-7B with no hyperparameter retuning and the same 128 demonstrations as those used in Tab. 2. We align our evaluation protocol with HPT, and the result is illustrated in Tab. 10, which shows that our method outperforms RL and RL based on SFT over the full 46K dataset, and the margin is concentrated where measurement is most reliable (i.e. the larger datasets, MATH-500 and OlympiadBench).

Table 10: FEST-DPO and FEST-GRPO on Qwen2.5-Math-7B models evaluated under the same protocol of HPT, which shows consistent performance increase over RL and SFT-then-RL.

	AIME25	AMC	AIME24	MATH-500	Olympiad	Minerva	Average
FEST-DPO	15.4	68.4	20.8	85.8	54.6	39.9	47.5
FEST-GRPO	21.3	64.4	22.5	86.4	54.0	38.8	47.9
RL	13.8	59.1	19.4	81.8	46.2	38.2	43.1
SFT-then-RL	21.6	62.2	25.7	84.6	46.8	38.2	46.5

D.3 Further Analysis on the Scalability of FEST-GRPO

In Fig. 2, we see the result of FEST-GRPO is stronger in the extreme low-shot regime, while FEST-DPO scales better with more demonstrations. While FEST-GRPO's performance curve seems flat when scaling above 256 shots, such phenomenon is driven by small high-variance benchmarks such as AIME24 with only 30 questions. Weighting by question count, for FEST-GRPO we find that the performance is indeed improving with more data as shown in Tab. 11:

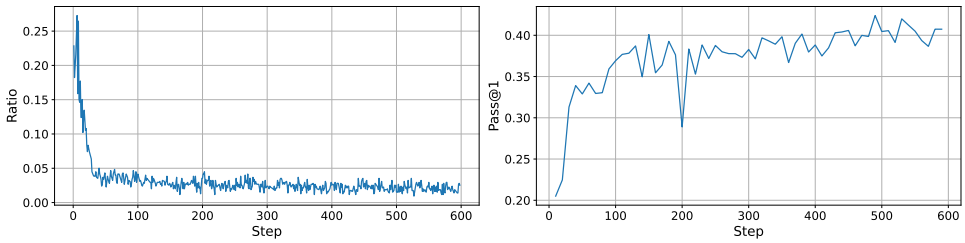
Table 11: FEST-GRPO's avg@8 accuracy, weighted by question count. With more shots, the performance trend is positive.

Shots	Avg@8
64	54.30
128	55.07
256	54.93
512	55.17

D.4 Supplementary Results for Methodology

HPT Data Ratio Dynamics. We replicated the HPT framework using the official implementation (Lv et al., 2025) and observed that the proportion of SFT trajectories decreases significantly as training progresses, reaching as low as 2% in the terminal stages (see Fig. 3). Notably, test set performance continues to improve even as the SFT signal diminishes. This empirical finding reinforces our hypothesis that the SFT component in demonstration-guided RLVR—regardless of whether the setting is few-shot or large-scale—should follow a decaying schedule. This principle is further corroborated by the design of CHORD (Zhang et al., 2026b), which incorporates similar decaying mechanisms in both its variants.

Gradient Mismatch Analysis. As discussed in Sec. 3.3, a fundamental structural discrepancy exists between the sequence-level objective of DPO and the token-level objective of GRPO. In DPO, the log-probabilities of the entire sequence are aggregated within the log-sigmoid function, whereas GRPO computes the average gradient across individual tokens. This difference, exacerbated by the extended reasoning traces characteristic of RLVR—which are significantly longer than typical RLHF sequences—results in a severe magnitude mismatch between the respective gradients. Specifically, as illustrated in Fig. 4, the norm of the DPO gradient typically ranges between 10^1 and 10^2, while the GRPO gradient norm consistently remains below 0.1. Our proposed FEST-GRPO variant naturally resolves this imbalance by unifying both objectives within a consistent token-level framework, thereby eliminating the need for exhaustive hyperparameter search for the coefficient c.



(a) SFT trajectory ratio (b) Test set performance

Figure 3: Evolution of SFT trajectory ratios and corresponding Pass@1 performance during HPT replication on the full dataset. The ratio of expert trajectories declines to approximately 2% of total rollouts, yet performance metrics consistently trend upward. This observation provides empirical justification for the use of a decaying weight strategy in the SFT objective during later training phases.

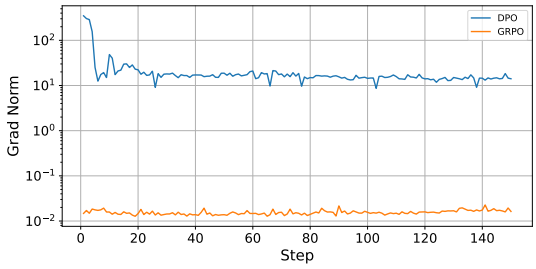
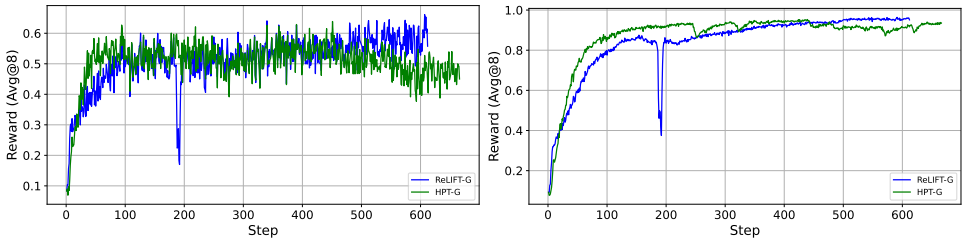


Figure 4: Gradient norm comparison between DPO and GRPO objectives when applied independently. The sequence-based DPO objective yields gradients multiple orders of magnitude larger than the token-based GRPO. FEST-GRPO harmonizes these scales to ensure stable joint optimization.

D.5 HPT-G AND RELIFT-G

Fig. 5 characterizes the training dynamics of HPT-G and ReLIFT-G, specifically focusing on the impact of applying reinforcement learning to the few-shot gold dataset D_E. As evidenced by the plots, both methods experience acute performance degradation in the middle of the training process across both D_E and the RL dataset D_I. We hypothesize that this instability is driven by abrupt distribution shifts: the SFT gradients introduce localized updates that conflict with a policy already heavily optimized (and potentially overfitted) by RL on the limited samples of D_E. This behavioral collapse underscores the difficulty of naively mixing objectives on sparse data and highlights the importance of the decaying weight strategy employed by FEST to maintain optimization stability.



(a) Reward (Avg@8) performance on D_I (b) Reward (Avg@8) performance on D_E

Figure 5: Training set accuracy profiles for ReLIFT-G and HPT-G. Both variants exhibit significant training instability, characterized by sudden and simultaneous performance drops.

D.6 ANALYSIS ON β

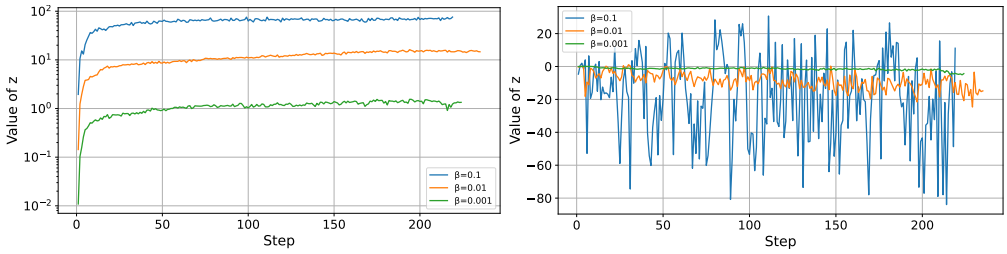
To investigate how the choice of β influences performance, we conduct a sensitivity analysis on the hyperparameter β within the FEST-DPO framework. Our setting diverges from standard DPO

applications in two fundamental ways: **extended Chain-of-Thought (CoT) reasoning** and the **few-shot regime**. The former involves reasoning traces of up to 8,192 tokens—over a magnitude longer (Bai et al., 2022) than typical RLHF sequences—leading to significantly larger cumulative log-probabilities. The latter necessitates repeated training epochs over a minuscule dataset, which further drives the discrepancy between $\log \pi_\theta(y^+|x)$ and $\log \pi_\theta(y^-|x)$. Consequently, the log-ratio difference reaches magnitudes far exceeding those encountered in traditional alignment tasks.

To understand how these factors dictate the optimal β, we re-examine the DPO gradient in Eq. (4), where the weight applied to the score gradient is $\beta \cdot \sigma(\beta(r^- - r^+))$. Adopting DPO's terminology for "implicit reward," we define the *implicit advantage* z for a given pair as:

$$z = \beta \cdot \left(\log \frac{\pi_\theta(y^+|x)}{\pi_{\text{ref}}(y^+|x)} - \log \frac{\pi_\theta(y^-|x)}{\pi_{\text{ref}}(y^-|x)} \right) = \beta \cdot \Delta, \tag{14}$$

and the resulting gradient coefficient is $w = \beta \cdot \sigma(-z) = \frac{\beta}{1+e^z}$. Because z is typically very large in our reasoning tasks (see Fig. 6 (a)), this coefficient is dominated by the exponential term rather than the linear factor β. Paradoxically, decreasing β can lead to a much stronger learning signal. Assuming a fixed log-ratio difference Δ, reducing β to 0.1β causes the coefficient to scale by a factor of approximately $0.1\frac{1+e^z}{1+e^{0.1z}}$. The balance threshold for this scaling is $z \approx 3.12$, which, according to Fig. 6 (a), corresponds to a β value slightly above 0.001. Thus, counterintuitively, **when $\beta > 0.001$, smaller β values provide stronger and more persistent learning signals**, necessitating the use of values significantly lower than the standard 0.1–0.2 range (Rafailov et al., 2023) (e.g., 0.001–0.1).



(a) Average z (b) Minimum z

Figure 6: Implicit advantage z as a function of β. Panel (a) shows that average z scales approximately linearly with β, i.e., the log-ratio difference is nearly consistent and our assumption holds for $\beta \in [0.001, 0.1]$. Panel (b) reveals that higher β values result in a wider distribution of z, where a few examples with very low z receive intense "switch-like" signals. Conversely, lower β values yield a more concentrated z distribution, resulting in a more constant and stable learning signal.

However, according to Fig. 6 (b), on the other hand, the absolute value of the minimum value of z also greatly increases. Such phenomenon indicates that the learning signal strength is not monotonic with respect to β, but instead forms a "switch-constant" dichotomy:

- With larger β, the learning is closer to a "switch", where the signal strength will be exceptionally strong at the beginning of training or in some corner cases where the implicit reward of y^+ is lower than that of y^-, but quickly decrease to almost 0;
- With smaller β, the learning signal is strong (instead of weak in usual belief (Rafailov et al., 2023)) and constant.

Based on these findings, we conducted an ablation study on β for FEST-DPO, with results summarized in Tab. 12. As we need to align our training set and test set with the baselines, and none of the baselines that uses the same dataset reports a separate validation set (ReLIFT, HPT, LUFFY, MIFO, SRFT) in their main result, we curate a held-out validation set of 2000 questions from OpenR1-Math-220K for validation with most sources being `olympiad` and `cn_contest` questions. The results show that while our method is robust across a range of hyperparameters, values that are too far from the balance point (either too large or too small) can degrade performance. We selected the optimal configuration for FEST-DPO. For FEST-GRPO, we heuristically chose a set of hyperparameters

such that z predominantly falls between 1 and 10, maintaining the signal near the balance point of $z \approx 3.12$.

Table 12: Test set performance (Avg@8) with FEST-DPO using different β. Generally, we find that our algorithm works well with β between 0.001 and 0.1, but suffers from performance loss if all hyperparameters are too large (close to 0.1) or too small (close to 0.001).

$(\beta_1, \beta_2, \beta_3)$	Val Avg@8	Test Avg@8
(0.1, 0.1, 0.1)	35.56	40.22
(0.1, 0.1, 0.05)	36.90	41.51
(0.1, 0.1, 0)	38.53	41.76
(0.1, 0.01, 0.01)	**39.47**	**41.98**
(0.01, 0.01, 0.001)	37.45	41.33
(0.001, 0.001, 0.001)	36.66	40.24

D.7 Training Curves

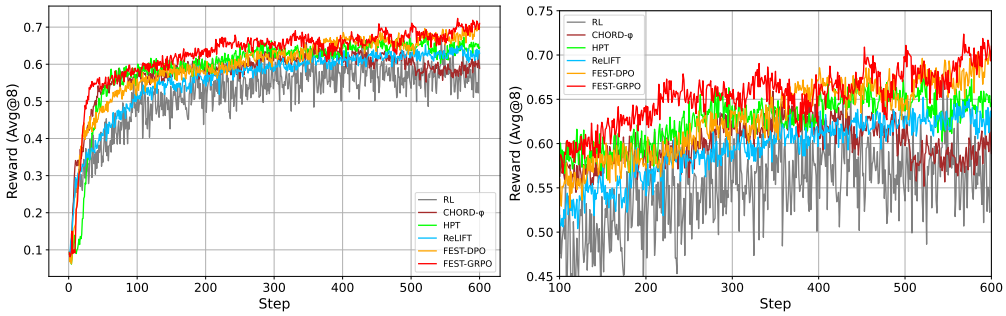
Primary Training Results (Sec. 4.1). Figures 7, 8, and 9 illustrate the accuracy profiles on the few-shot SFT dataset D_E, the RL dataset D_I, and the test set, respectively. For enhanced readability, we provide zoomed-in views focusing on the 100–600 step interval. The test set performance is smoothed due to its high variance nature (to prevent interrupting the training process, we use Pass@1 for evaluation during training). We identify three critical trends in these dynamics:

- **Stratification of Reward Profiles on D_E:** The accuracy curves on D_E bifurcate into two distinct groups. The higher-performing group (ReLIFT-G, HPT-G, RL-G, and LUFFY) consists of methods that directly apply RL to the gold dataset. Within this group, ReLIFT-G and HPT-G exhibit significant instability compared to RL-G and LUFFY, likely due to the high SFT weighting. Conversely, LUFFY maintains better stability through regularized importance sampling; however, this stability is sensitive to the learning rate, as we observed instability when LUFFY was scaled to our specific configuration.
- **Convergence vs. Overfitting:** While LUFFY, CHORD-ϕ, and HPT exhibit rapid initial convergence comparable to FEST, they eventually suffer from performance stagnation on the test set, indicating a susceptibility to overfitting. In contrast, under few-shot constraints, ReLIFT closely mirrors vanilla RL dynamics because its SFT triggering mechanism—averaging only once every 20 RL steps—is too infrequent to provide meaningful expert guidance.
- **FEST Superiority:** Across the D_E (lower group), D_I, and test set metrics, both FEST-DPO and FEST-GRPO consistently maintain a substantial performance margin over all baseline methods throughout the training process.

Dynamics on LIMOv2-8192. Fig. 10 presents the training dynamics for the LIMOv2-8192 experiment (Sec. 4.3). While performance on D_I and the test set shows continuous growth, accuracy on D_E remains relatively stationary. We attribute this to the substantially higher reasoning difficulty of the LIMOv2 dataset compared to standard splits. Nevertheless, the framework successfully extracts performance gains from this specialized few-shot dataset, demonstrating robust cross-dataset applicability.

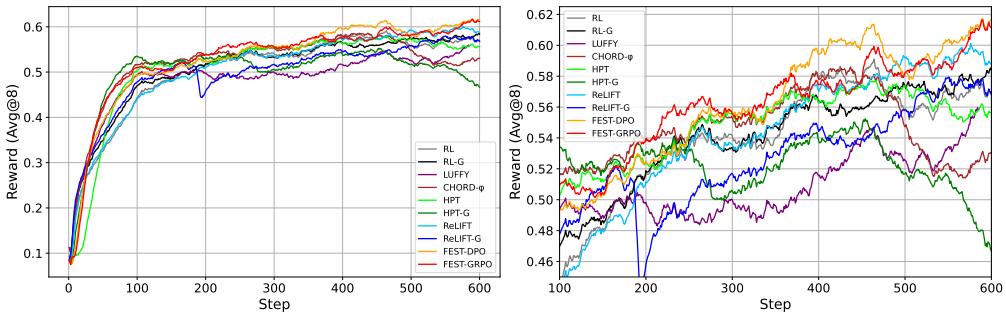
D.8 SFT and SPIN

In Sec. 4.1, we omit the formal performance report for standalone SFT and SPIN (Chen et al., 2024b) on the 128-shot dataset. This exclusion is due to the inherent inability of these methods to generalize within extreme few-shot regimes without the benefit of an auxiliary RL objective on D_I. Specifically, both methods exhibit rapid and severe overfitting from the earliest stages of optimization. Consequently, these training runs were terminated prior to the 600-step mark to prioritize computational resource allocation. Figure 11 illustrates these dynamics.



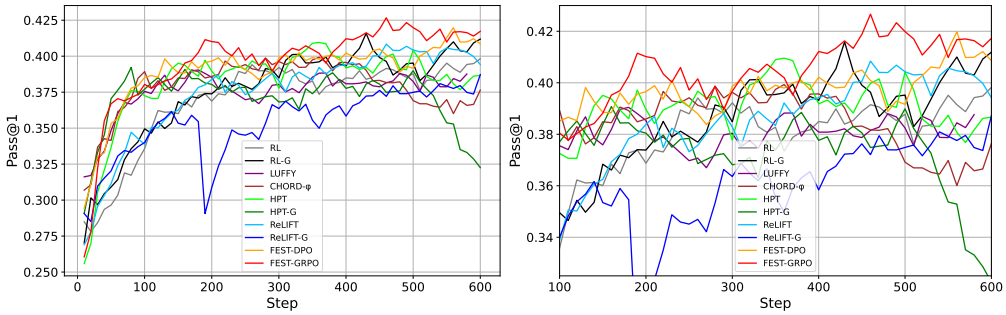
(a) D_E (few-shot SFT training set) accuracy (b) zoomed-in curve (100–600 step)

Figure 7: Reward curves on D_E during training. Results for the "higher group" (LUFFY, ReLIFT-G, HPT-G, and RL-G) are excluded from these plots as their direct RL training on D_E leads to near 100% training accuracy, masking meaningful comparative dynamics. The performance implications of this overfitting are discussed in Appendix D.5 and Tab. 3.



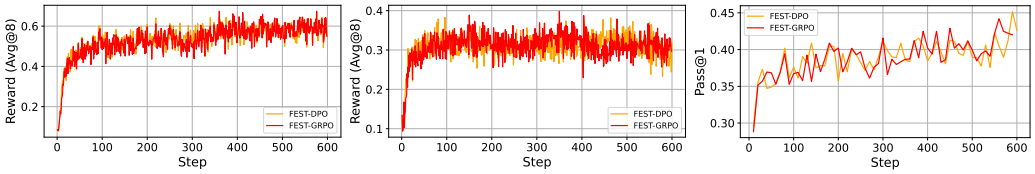
(a) D_I (RL training set) accuracy (b) zoomed-in curve (100–600 step)

Figure 8: Smoothed reward curves on D_I utilizing a time-weighted exponential moving average.



(a) test set accuracy (b) zoomed-in curve (100–600 step)

Figure 9: Smoothed Pass@1 performance on the test set through the training process.



(a) D_I (RL training set) accuracy (b) D_E (gold SFT set) accuracy (c) Pass@1 on test set

Figure 10: Training and test set performance curves for the LIMOv2-8192 experiment described in Sec. 4.3.

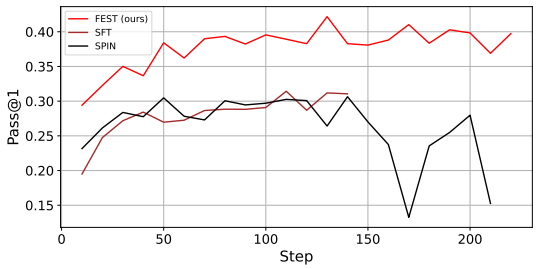


Figure 11: Test set performance comparison between FEST, standalone SFT, and SPIN. Our method significantly outperforms both baselines. SPIN, in particular, demonstrates a characteristic overfitting trajectory on the few-shot dataset, with performance peaking early and then rapidly declining. Due to this divergence and limited computational resources, these baselines were not extended to the full training duration or reported in the main paper.

E Computational Resources

All experiments were conducted on a single compute node equipped with two NVIDIA GH200 (96GB) Grace Hopper Superchips, 64 ARM-based CPU cores, and 256GB of system memory. A standard training run of 600 steps requires approximately 4–5 days of wall-clock time to complete, while an evaluation across our benchmark suite (utilizing 8 rollouts per question) takes approximately 30 minutes.

F Licenses

The licensing information for the datasets, benchmarks, and baseline models utilized in this study is summarized in Tab. 13. While several assets lack explicit licensing documentation, they are recognized as standard resources within the research community and have been extensively utilized in prior work (Lv et al., 2025; Ma et al., 2026; Yan et al., 2025).

Table 13: Licenses of the assets utilized in this work. Entries marked as "N/A" indicate that no explicit license was provided by the original authors or repositories.

Asset Name	Category	License
Qwen2.5-Math-1.5B	base model	Apache-2.0
OpenR1-Math-46K-8192	training set	MIT
LIMOv2	training set	Apache-2.0
AMC23	benchmark	N/A
AIME24	benchmark	Apache-2.0
AIME25	benchmark	Apache-2.0
OlympiadBench	benchmark	Apache-2.0
MATH-500	benchmark	N/A
Minerva	benchmark	N/A
MMLU-Pro	benchmark	MIT
ReLIFT	code base	N/A
HPT	code base	MIT
CHORD	code base	Apache-2.0
LUFFY	code base	N/A
SRFT	checkpoint	MIT