Extended Abstract

Motivation Reinforcement-learning-based post-training has become a central technique for improving the mathematical reasoning ability of large language models. Group Relative Policy Optimization (GRPO) is especially attractive because it avoids learning a separate value model and instead estimates advantages from relative rewards within a group of sampled responses. However, GRPO can be inefficient when rewards are sparse. For difficult math problems, all sampled responses in a group may be incorrect. In this all-failed case, all rewards are zero, so the group provides little or no useful relative advantage signal. This is problematic because all-failed groups often correspond to exactly the hard examples from which the model could learn the most.

Method This project proposes **Feedback-Augmented GRPO with Policy Shaping**, a method for reusing all-failed reasoning trajectories during GRPO post-training. The method keeps standard GRPO unchanged when a sampled group already contains reward variation and intervenes only when all sampled responses fail. For an all-failed group, the method selects a failed trajectory, obtains external natural-language feedback that identifies the main reasoning error, and asks the policy model to revise the failed trajectory. The revised trajectory is accepted only if it passes an automatic verifier. If accepted, it replaces the original failed trajectory and receives a positive reward weight, converting an all-zero group such as $(0, 0, 0, 0)$ into a contrastive group such as $(0, 0, \lambda, 0)$. To further strengthen learning from verified corrections, the method applies policy shaping to corrected trajectories. The shaped token-level ratio $\rho = p/(p+\gamma)$ gives a bounded and stable learning signal for verified-correct tokens, especially when these tokens are initially low-probability under the current policy.

Implementation I implement the training and evaluation pipeline for comparing feedback-based recovery against standard GRPO and self-feedback. The training set is `GSM8K`. The automatic verifier provides binary outcome rewards. I compare three settings: self-feedback, standard GRPO, and feedback-augmented GRPO. Self-feedback asks the model to critique and refine its own failed responses. Standard GRPO uses only outcome rewards. Feedback-augmented GRPO uses external feedback to guide failed-trajectory refinement and accepts only verifier-approved corrections.

Results I evaluate transfer validation accuracy on five math reasoning benchmarks not used for training: `math`, `amc`, `aime`, `olympiad_bench`, and `minerva`. Across checkpoints 400, 800, and 1200, feedback-augmented GRPO is consistently competitive and achieves the best or tied-best result for most dataset-step pairs. At step 1200, feedback-augmented GRPO achieves the best result on all five validation sets, obtaining 90.34% on `math`, 72.29% on `amc`, 36.67% on `aime`, 63.59% on `olympiad_bench`, and 48.90% on `minerva`. The average accuracy improves from 60.43% for standard GRPO to 62.36% for feedback-augmented GRPO with policy shaping.

Analysis Failure-recovery diagnostics show that 27.4% of sampled training groups are all-failed. Self-feedback produces verifier-approved refinements for 7.8% of attempted refinements and recovers 13.6% of all-failed questions. External feedback substantially improves recovery, producing valid refinements for 24.9% of attempts and recovering 38.2% of all-failed questions. An ablation further shows that feedback-based recovery without policy shaping reaches 61.38% average accuracy, while adding policy shaping increases average accuracy to 62.36%. Qualitative analysis shows that external critique can identify a key reasoning error and guide the model toward a verified correct refinement.

Conclusion Failure-only samples should not always be discarded. With external feedback, automatic verification, and policy shaping, all-failed GRPO groups can become useful training signal for mathematical reasoning. The method depends on the quality and cost of external feedback, and it is most useful when failed trajectories contain recoverable reasoning structure. Overall, the project shows that learning from failure is a practical way to improve GRPO-style post-training.

Learning from Failure: Natural Language Feedback for Reusing Failed GRPO Trajectories

Chenyue Li
Department of Electrical Engineering
Stanford University
chenyuel@stanford.edu

Abstract

Group Relative Policy Optimization (GRPO) improves mathematical reasoning by learning from relative rewards within sampled response groups. However, when all sampled responses are incorrect, the group contains no reward variation and provides little useful learning signal. This paper proposes Feedback-Augmented GRPO with Policy Shaping, which reuses such all-failed groups through external natural-language feedback, verifier-gated refinement, and shaped optimization. For an all-failed group, external feedback identifies the main reasoning error in a failed trajectory, the policy generates a revised solution, and only verifier-approved refinements are used to replace failed trajectories. Policy shaping further strengthens learning from verified corrections that are initially low-probability under the policy. Experiments on `GSM8K` training and five out-of-domain math reasoning benchmarks show that the proposed method outperforms standard GRPO and self-feedback at the final checkpoint, improving average accuracy from 60.42% to 62.36%. Recovery diagnostics further show that external feedback recovers substantially more all-failed questions than self-feedback. These results suggest that failed trajectories can provide useful training signal when combined with feedback, verification, and shaped optimization.

1 Introduction

Reinforcement learning has become an important approach for improving the reasoning ability of large language models Shao et al. (2024); Guo et al. (2025). In mathematical reasoning, outcome rewards are often easy to compute using automatic verifiers: a final answer is either correct or incorrect. This makes reinforcement learning attractive for post-training reasoning models.

Group Relative Policy Optimization (GRPO) is a widely used RL method for reasoning post-training. Instead of training a separate value model, GRPO samples multiple responses for the same problem and computes advantages from the relative rewards within the group. This works well when the group contains both correct and incorrect responses. In that case, correct responses receive positive relative advantage, and incorrect responses receive lower advantage.

However, difficult math problems often produce all-failed groups. In such a group, every sampled response is incorrect:

$$r_1 = r_2 = \cdots = r_G = 0.$$

This creates a key limitation. Since all rewards are identical, the group has little or no reward variation. Therefore, GRPO receives almost no useful relative advantage signal. These all-failed groups are not rare edge cases; they often correspond to the hardest examples, where additional learning signal would be most valuable.

This project studies whether failed trajectories can be reused instead of discarded. The core idea is to use external natural-language feedback to repair all-failed groups. If a failed trajectory can be revised

into a verifier-approved correct solution, then the original all-zero reward group can be converted into a contrastive group with a verified positive trajectory. I further introduce policy shaping to help the model internalize verified corrections that may be low-probability under the current policy.

The main contributions are:

- I identify all-failed GRPO groups as an important failure mode in sparse-reward mathematical reasoning, where standard group-relative optimization provides little useful advantage signal.
- I propose Feedback-Augmented GRPO, which uses external natural-language feedback and verifier-gated replacement to convert some all-failed groups into contrastive training examples.
- I introduce policy shaping for verifier-approved refined trajectories and show through quantitative, diagnostic, and qualitative analysis that the method improves over standard GRPO and self-feedback.

2 Related Work

Reinforcement learning for mathematical reasoning. Reinforcement-learning-based post-training has become an important approach for improving the reasoning ability of large language models. DeepSeekMath introduced GRPO as an efficient alternative to PPO-style training by replacing the learned value model with group-relative reward normalization Shao et al. (2024). DeepSeek-R1 further demonstrates that large-scale reinforcement learning can elicit reasoning behaviors such as reflection, verification, and long-form problem solving Guo et al. (2025). These methods show the effectiveness of RL for mathematical reasoning, but standard outcome-reward training still depends on reward variation within a sampled group. When all sampled responses are incorrect, the group provides little useful relative advantage signal. My work addresses this failure mode by reusing all-failed groups through feedback-guided recovery.

Learning from natural-language feedback. Natural-language feedback provides richer supervision than scalar rewards. RLHF shows that human preference feedback can substantially improve instruction-following behavior Ouyang et al. (2022), while Constitutional AI demonstrates that AI-generated critiques can reduce reliance on direct human labels Bai et al. (2022). However, these works mainly focus on general alignment, helpfulness, and harmlessness. In contrast, this project studies how natural-language feedback can be used during reasoning post-training to recover failed mathematical trajectories. The feedback is not used as a direct imitation target; instead, it diagnoses the error and conditions the policy to generate a revised response.

Self-correction and feedback-guided refinement. Several prior methods use self-generated feedback or reflection to improve model outputs. Self-Refine uses iterative self-feedback to revise model responses Madaan et al. (2023), and Reflexion uses verbal feedback to improve agent behavior over multiple trials Shinn et al. (2023). Recent work also studies training models to self-correct through reinforcement learning Kumar et al. (2025). At the same time, prior analysis suggests that language models often cannot reliably self-correct reasoning errors without external feedback Huang et al. (2024). This motivates the comparison in my experiments between self-feedback and GPT-5-based external feedback.

Training infrastructure and base models. My implementation follows the GRPO-style online RL setting and uses Qwen3-series thinking models as the policy backbone Yang et al. (2025). The training pipeline is implemented in a verl/HybridFlow-style RLHF framework Sheng et al. (2024). Unlike prior work that mainly focuses on scaling RL post-training, this project focuses on a targeted intervention: converting all-failed groups into useful contrastive training examples through external critique, verification, and policy shaping.

3 Method

This section presents **Feedback-Augmented GRPO with Policy Shaping**. The method targets a specific failure mode of GRPO: when all sampled responses for a problem are incorrect, the group contains little reward variation and therefore provides almost no useful advantage signal. The

proposed method keeps standard GRPO unchanged for groups that already contain both correct and incorrect responses, and only modifies all-failed groups through feedback-guided recovery and shaped optimization.

3.1 From Standard GRPO to All-Failed Groups

For each problem x, the old policy samples a group of G reasoning trajectories:

$$y_i = (y_{i,1}, \ldots, y_{i,T_i}) \sim \pi_{\theta_{\text{old}}}(\cdot \mid x), \qquad i = 1, \ldots, G.$$

Each trajectory is evaluated by an automatic verifier:

$$r_i = R(x, y_i) \in \{0, 1\},$$

where $r_i = 1$ means that the final answer is correct and $r_i = 0$ means that it is incorrect. Standard GRPO computes group-relative advantages:

$$A_i = \frac{r_i - \mu_G}{\sigma_G + \epsilon}, \qquad \mu_G = \frac{1}{G}\sum_{j=1}^{G} r_j, \qquad \sigma_G = \text{std}(\{r_j\}_{j=1}^{G}).$$

When the reward vector contains both successes and failures, such as $(0, 1, 0, 1)$, these normalized advantages provide a useful contrastive signal. However, for difficult problems, the model may sample an all-failed group:

$$\sum_{i=1}^{G} r_i = 0, \qquad (r_1, \ldots, r_G) = (0, \ldots, 0).$$

In this case, scalar rewards only indicate that all responses are wrong. They do not indicate which trajectory is closest to being correct, where the reasoning fails, or how the model should revise its solution. This makes all-failed groups underused by standard GRPO, even though they often correspond to hard and informative training examples.

3.2 Feedback-Guided Recovery

For groups with at least one correct response, I directly apply the standard GRPO update. For an all-failed group, I select one failed trajectory y_k and use an external critic to generate natural-language feedback:

$$f_k = C(x, y_k, a^\star),$$

where $a^\star$ is the reference answer or reference solution. The feedback f_k diagnoses the main reasoning error and provides a correction direction. It is used only as conditioning information; the policy is not optimized to reproduce the critique text.

Conditioned on the original problem, the failed trajectory, and the feedback, the old policy generates a refined response:

$$\tilde{y}_k \sim \pi_{\theta_{\text{old}}}(\cdot \mid x, y_k, f_k).$$

The refined response is then evaluated by the same verifier:

$$\tilde{r}_k = R(x, \tilde{y}_k).$$

Only verifier-approved refinements are used as positive training trajectories. The replacement rule is:

$$(y_k, r_k) \leftarrow \begin{cases} (\tilde{y}_k, \lambda), & \tilde{r}_k = 1, \\ (y_k, 0), & \tilde{r}_k = 0, \end{cases} \qquad \lambda > 0.$$

In the binary-reward setting, λ can be set to 1. This step converts an uninformative all-zero group into a contrastive group:

$$(0, 0, 0, 0) \quad \longrightarrow \quad (0, 0, \lambda, 0).$$

After this replacement, the modified group contains reward variation, so GRPO can compute meaningful relative advantages and learn from a hard example that would otherwise provide little training signal.

It is important to distinguish the three text objects involved in this process. The original failed response is y_k, the feedback is f_k, and the refined response is $\tilde{y}_k$. The feedback f_k is not treated as a target response. If the refinement fails verification, the original failed response remains in the group with zero reward. If the refinement passes verification, the original response is replaced by the refined response, and optimization is applied to the verified refined trajectory $\tilde{y}_k$.

3.3 Policy Shaping Objective

Feedback-guided replacement gives an all-failed group a verified positive trajectory. However, the verified refined response $\tilde{y}_k$ may still be unlikely under the current policy. This means that it may contain correct reasoning tokens that the model would not naturally generate during ordinary sampling. To better learn from these rare but verified-correct patterns, I apply policy shaping only to verifier-approved refined responses.

For ordinary sampled trajectories, I use the standard GRPO token-level probability ratio:

$$r_{i,t}(\theta) = \frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})}.$$

For a verified refined trajectory $\tilde{y}_k$, I define the current token probability:

$$p_{k,t}(\theta) = \pi_\theta(\tilde{y}_{k,t} \mid x, \tilde{y}_{k,<t}),$$

and use the shaped token weight:

$$\rho_{k,t}(\theta) = \frac{p_{k,t}(\theta)}{p_{k,t}(\theta) + \gamma}.$$

In all experiments, I set $\gamma = 0.1$. The parameter γ controls the saturation behavior of the shaping function. Since $0 < \rho_{k,t}(\theta) < 1$, the shaped weight is bounded and stable. When $p_{k,t} = \gamma = 0.1$, the shaped weight is 0.5, so γ can be interpreted as the probability scale at which a corrected token receives half of the maximum shaped weight. The derivative

$$\frac{\partial \rho_{k,t}}{\partial p_{k,t}} = \frac{\gamma}{(p_{k,t} + \gamma)^2}$$

is larger when $p_{k,t}$ is small and smaller when $p_{k,t}$ is large. Thus, the shaping function is more sensitive to low-probability corrected tokens and gradually saturates for tokens that the model already assigns high probability to. Intuitively, this encourages the model to internalize verified-correct reasoning patterns that are initially unfamiliar, while preventing already-likely tokens from dominating the update.

The final objective combines the ordinary GRPO loss and the shaped correction loss. For ordinary trajectories, the GRPO loss is:

$$\mathcal{L}_{\text{GRPO}} = -\sum_{i \in \mathcal{I}_{\text{std}}} \sum_{t=1}^{T_i} \min(r_{i,t}(\theta)A_i,\ \text{clip}(r_{i,t}(\theta), 1-\eta, 1+\eta)A_i)\,,$$

where η is the clipping range and $\mathcal{I}_{\text{std}}$ denotes ordinary sampled trajectories. For verifier-approved refined trajectories, the shaped loss is:

$$\mathcal{L}_{\text{shape}} = -\sum_{k \in \mathcal{I}_{\text{corr}}} \sum_{t=1}^{\tilde{T}_k} \min(\rho_{k,t}(\theta)A_k,\ \text{clip}(\rho_{k,t}(\theta), 1-\eta, 1+\eta)A_k)\,,$$

where $\mathcal{I}_{\text{corr}}$ denotes verified refined trajectories. The total loss is:

$$\mathcal{L} = \mathcal{L}_{\text{GRPO}} + \alpha \mathcal{L}_{\text{shape}},$$

where α controls the relative weight of the shaped correction loss.

Overall, the method first uses feedback and verification to convert some all-failed groups into contrastive training examples, and then uses policy shaping to strengthen learning from the verified refined responses. Feedback explains the error, verification filters noisy revisions, and the shaped objective helps the model learn corrected reasoning patterns that are correct but initially low-probability under the policy.

4 Experimental Setup

Training data. I use `GSM8K` as the RL training set Cobbe et al. (2021). Each training example consists of a grade-school math word problem and a reference answer. The reference answer is used only for automatic verification and for generating external feedback in the feedback-augmented setting; it is not directly provided to the policy during ordinary GRPO sampling. Training on `GSM8K` allows the model to learn from relatively short mathematical reasoning problems while testing whether the learned recovery behavior transfers to harder benchmarks.

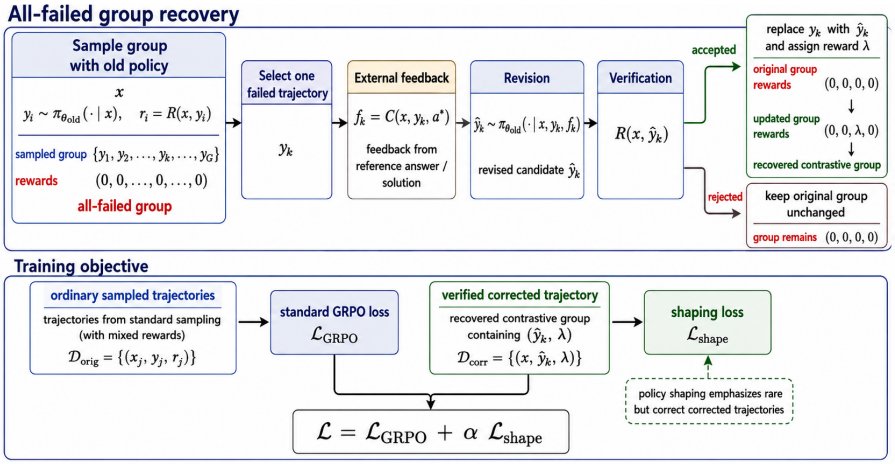

Figure 1: Method overview. Standard GRPO is used for groups with reward variation. For all-failed groups, external feedback guides revision of a failed trajectory, the verifier accepts only correct refinements, and policy shaping strengthens learning from verified corrected trajectories.

Policy model and training framework. I implement the training pipeline in a GRPO-style online RL setting using a Qwen3-8b-instruct thinking model as the backbone Yang et al. (2025). The pipeline follows the standard sample-score-update loop used in GRPO-style reasoning post-training Shao et al. (2024). For each problem, the old policy samples a group of reasoning trajectories, an automatic verifier assigns binary outcome rewards, and the policy is updated using group-relative advantages. The feedback-augmented version adds all-failed group detection, GPT-5 feedback generation, feedback-conditioned refinement, verifier-gated trajectory replacement, and policy shaping for verified refined trajectories. The implementation follows a verl/HybridFlow-style training framework for scalable RLHF and RL post-training Sheng et al. (2024).

Automatic verifier. All methods use the same automatic outcome verifier. Given a problem x and a generated solution y, the verifier extracts the final answer and checks whether it matches the reference answer. The reward is binary:

$$R(x, y) \in \{0, 1\}.$$

A reward of 1 indicates a verified correct answer, while a reward of 0 indicates an incorrect answer. Using the same verifier across all methods ensures that performance differences come from the training strategy rather than from different evaluation rules.

External feedback source. For feedback-augmented GRPO, I use GPT-5 as the external critic OpenAI (2025). GPT-5 receives the original problem, the failed model response, and the reference answer or solution. It then produces natural-language feedback that identifies the main reasoning error and gives a concise correction direction. The feedback is not optimized as a target output. Instead, it is used as conditioning information for the policy model to generate a refined response. The refined response is accepted only if it passes the same automatic verifier.

Compared methods. I compare three training methods.

- **Standard GRPO:** the model samples a group of responses and is trained only with binary outcome rewards Shao et al. (2024). All-failed groups remain unchanged, so they provide little useful group-relative signal.
- **Self-feedback:** when a sampled response fails, the policy model critiques or refines its own response. This baseline is motivated by self-refinement and reflection-style methods Madaan et al. (2023); Shinn et al. (2023), but tests whether the policy model's own feedback is sufficient for failure recovery during RL training.
- **Feedback-augmented GRPO:** for all-failed groups, GPT-5 provides external natural-language feedback for a selected failed trajectory OpenAI (2025). The policy then generates a refined

response conditioned on the original problem, the failed response, and the feedback. If the refined response passes verification, it replaces the original failed trajectory and receives a positive reward weight. Policy shaping is then applied to the verified refined trajectory.

Evaluation datasets. I evaluate transfer performance on five mathematical reasoning validation sets that are not used for training: `math` Hendrycks et al. (2021a), `amc23` Mathematical Association of America (2023); Yang et al. (2024), `aime` Mathematical Association of America (2023); Yang et al. (2024), `olympiad_bench` He et al. (2024), `minerva` Lewkowycz et al. (2022a). These datasets cover different levels of difficulty. The `math` validation set is based on competition-style mathematical reasoning tasks Hendrycks et al. (2021b). The `minerva` validation set measures broader quantitative reasoning ability Lewkowycz et al. (2022b). The `olympiad_bench` validation set contains more challenging olympiad-level problems He et al. (2024). The `amc` and `aime` validation sets are competition-style subsets used to test transfer to harder mathematical reasoning. This evaluation setup tests whether feedback-guided recovery learned from `GSM8K` transfers beyond the training distribution.

Evaluation protocol. I report validation accuracy at checkpoints 400, 800, and 1200. Each value is the percentage of validation problems whose final answer is verified as correct. All methods are evaluated under matched checkpoint and decoding settings. The main comparison uses checkpoint-level validation accuracy, while the final-checkpoint comparison summarizes average accuracy across the five validation sets at step 1200.

Failure-recovery diagnostics. In addition to validation accuracy, I report recovery-oriented diagnostics that directly measure whether feedback recovers all-failed groups. I use three metrics:

- **All-failed group rate:** the percentage of sampled training groups where all initial trajectories receive zero reward.
- **Valid refinement rate:** among attempted refinements, the percentage of refined trajectories that pass the verifier.
- **Question recovery rate:** among all-failed questions, the percentage for which at least one feedback-guided refinement becomes correct.

These metrics are important because final validation accuracy alone does not directly show whether the method actually recovers failed trajectories during training.

Policy shaping ablation. To isolate the effect of policy shaping, I compare feedback-augmented GRPO with and without the shaped correction loss. Both variants use the same GPT-5 feedback and verifier-gated replacement mechanism. The only difference is whether verified refined trajectories receive the shaped token-level objective. This ablation tests whether the improvement comes only from inserting corrected trajectories into the group, or whether the shaped objective further helps the model internalize low-probability but verified-correct reasoning patterns.

4.1 Quantitative Evaluation

Table 1 reports validation accuracy at checkpoints 400, 800, and 1200 after training on `GSM8K`. Feedback-augmented GRPO is consistently competitive across checkpoints and achieves the best or tied-best result for most dataset-step pairs. At the final checkpoint, it obtains the best result on all five validation sets.

To summarize final performance, I compute the average accuracy across the five validation sets at step 1200. Self-feedback obtains an average accuracy of 58.55%, standard GRPO obtains 60.42%, and FB-GRPO obtains 62.36%. Thus, FB-GRPO improves over standard GRPO by 1.94 percentage points on average and over self-feedback by 3.81 percentage points.

Table 2 reports failure-recovery diagnostics on all-failed GRPO groups. These diagnostics directly measure whether the proposed feedback mechanism can turn otherwise uninformative failed groups into useful training examples.

The all-failed group rate is 27.4%, showing that all-failed groups are a substantial part of the training process rather than a rare edge case. Self-feedback produces verifier-approved refinements for only

Table 1: Validation accuracy at checkpoints 400, 800, and 1200.

Validation Set	Step	Self-FB	GRPO	FB-GRPO
`math`	400	83.80%	91.00%	**92.35%**
`math`	800	83.80%	91.15%	**91.20%**
`math`	1200	90.00%	90.24%	**90.34%**
`amc`	400	65.06%	65.06%	**70.48%**
`amc`	800	63.86%	69.88%	**69.88%**
`amc`	1200	69.28%	69.88%	**72.29%**
`aime`	400	31.67%	36.67%	**36.67%**
`aime`	800	30.00%	50.00%	**50.00%**
`aime`	1200	26.67%	33.33%	**36.67%**
`olympiad_bench`	400	56.16%	57.50%	**61.11%**
`olympiad_bench`	800	59.01%	60.33%	**61.56%**
`olympiad_bench`	1200	59.76%	61.22%	**63.59%**
`minerva`	400	42.65%	48.16%	**48.53%**
`minerva`	800	46.69%	47.74%	**48.16%**
`minerva`	1200	47.06%	47.43%	**48.90%**

Table 2: Failure-recovery diagnostics on all-failed GRPO groups.

Method	All-Failed Groups	Valid Refinements	Questions Recovered
Self-feedback	27.4%	7.8%	13.6%
Feedback-augmented GRPO	27.4%	24.9%	38.2%

7.8% of attempted refinements and recovers 13.6% of all-failed questions. In contrast, feedback-augmented GRPO produces valid refinements for 24.9% of attempts and recovers 38.2% of all-failed questions. These results directly support the central motivation of the project: all-failed groups are not necessarily useless, but they require informative correction signals to become useful training examples.

Table 3 isolates the contribution of policy shaping at step 1200. The evaluation is conducted on the same five validation sets used in the main quantitative evaluation. The setting with policy shaping corresponds to the main FB-GRPO method reported in Table 1.

Feedback-based recovery without policy shaping already improves the average accuracy from 60.42% to 61.38%, showing that verifier-gated replacement alone provides useful additional training signal. Adding policy shaping further improves the average accuracy to 62.36%. The improvement appears across all five validation sets, with larger gains on `amc`, `aime`, and `olympiad_bench`. This suggests that verified corrections are useful as replacement trajectories, and that the shaped token-level objective helps the model better internalize corrected reasoning patterns that are initially unlikely under the policy.

4.2 Qualitative Analysis

The quantitative results reveal a consistent pattern across performance, recovery, and ablation metrics. First, the checkpoint-level results in Table 1 show that FB-GRPO does not merely improve one isolated benchmark. Instead, it achieves the best final-checkpoint accuracy on all five validation sets. This consistency is important because the model is trained only on `GSM8K`, while the evaluation sets cover different levels and styles of mathematical reasoning. Therefore, the improvement suggests that feedback-guided recovery helps the policy learn reasoning patterns that transfer beyond the training distribution.

Second, the gains are larger on relatively harder benchmarks such as `amc`, `aime`, and `olympiad_bench`. This trend matches the motivation of the method. On easier or already saturated benchmarks such as `math`, standard GRPO can already obtain many successful trajectories, so the additional value of recovering all-failed groups is limited. In contrast, harder benchmarks are more likely to expose weaknesses in the policy's reasoning ability. The stronger gains on these

Table 3: Ablation of policy shaping at step 1200. The last row is the main FB-GRPO method.

Method	math	amc	aime	olymp.	minerva	Avg.
Standard GRPO	90.24%	69.88%	33.33%	61.22%	47.43%	60.42%
FB-GRPO w/o policy shaping	90.28%	71.10%	35.00%	62.20%	48.30%	61.38%
FB-GRPO w/ policy shaping	**90.34%**	**72.29%**	**36.67%**	**63.59%**	**48.90%**	**62.36%**

benchmarks indicate that external feedback is most useful when ordinary outcome-reward GRPO has difficulty obtaining informative reward variation.

Third, Table 2 explains why external feedback improves over self-feedback. Both methods face the same all-failed group rate, but their recovery quality is very different. Self-feedback has a low valid refinement rate, suggesting that the policy model often cannot reliably locate or correct its own reasoning errors. External feedback, by contrast, produces substantially more verifier-approved refinements and recovers more all-failed questions. This indicates that the benefit of FB-GRPO comes not simply from adding another refinement step, but from adding a more reliable error diagnosis signal.

Fourth, Table 3 shows that feedback-based replacement and policy shaping make complementary contributions. The no-shaping variant already improves over standard GRPO, which means that converting all-failed groups into contrastive groups is useful by itself. However, the full method improves further when policy shaping is added. This suggests that corrected trajectories may contain reasoning tokens that are initially low-probability under the current policy. The shaped objective gives these verified-correct tokens a stronger and more stable learning signal, helping the model internalize corrected reasoning patterns rather than only treating the refined trajectory as another positive sample.

Together, these results support the main mechanism of the method. All-failed groups occur frequently, external feedback can recover a meaningful fraction of them, and policy shaping helps convert those verified corrections into transferable improvements. The quantitative gains are therefore not just empirical improvements in final accuracy; they are aligned with the intended training dynamics of feedback-guided recovery.

4.3 Feedback-Guided Refinement

Figure 2 shows an example of feedback-guided refinement. The initial response uses an incorrect geometric relation and obtains the wrong answer. The external CoT critique identifies the key error: the solution should use the inradius relation for the axial-section triangle. Conditioned on this critique, the refined response revises the derivation and obtains the correct final answer.

This example illustrates why natural-language feedback is useful. A scalar reward only says that the answer is wrong. The critique explains why the reasoning is wrong and provides a concrete direction for repair. The verifier then ensures that only correct refinements are used as positive training signal.

5 Discussion

External feedback is more reliable than self-feedback. Self-feedback is weaker than GRPO-based methods across all reported datasets. The recovery diagnostics also show that self-feedback has substantially lower valid refinement and question recovery rates. This suggests that the model does not reliably diagnose its own reasoning errors during training. External feedback provides a more informative correction signal.

Failed trajectories can become useful. All-failed groups are not necessarily useless. A failed trajectory may contain partial reasoning structure. With external critique, the model can revise this structure into a correct solution. Verifier-gated replacement then turns the original all-zero group into a contrastive training example.

Policy shaping strengthens correction learning. Verifier-approved corrections may be rare under the current policy. Policy shaping gives these corrected tokens a special bounded learning signal.

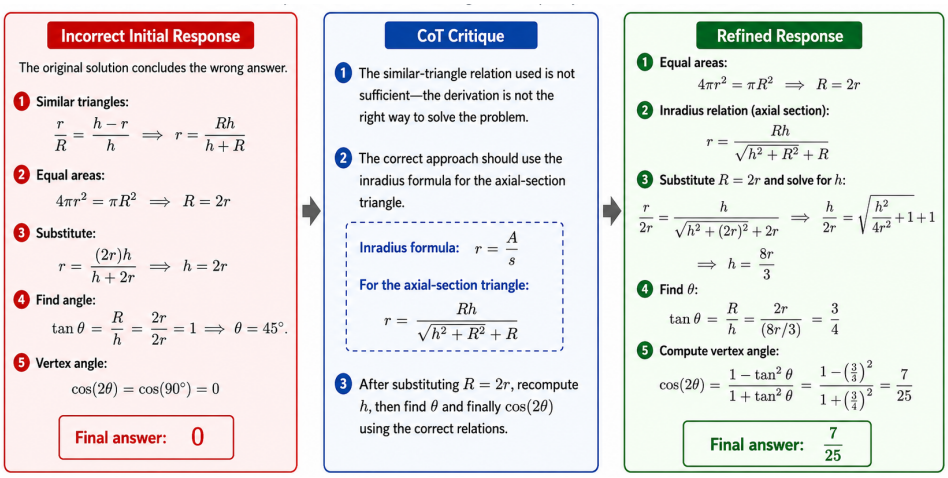

Figure 2: Example of feedback-guided refinement. External CoT critique identifies the key geometric error in the initial response and guides the policy to revise the solution using the correct inradius relation.

The ablation shows that policy shaping provides an additional gain over feedback-based replacement alone. This supports the intuition that the model benefits from a token-level objective that emphasizes low-probability but verified-correct reasoning patterns.

Limitations. The method depends on the quality and cost of external feedback. If the critique is wrong or unhelpful, the refinement may fail. The automatic verifier also plays an important role, because noisy accepted corrections could harm training. In addition, verifier-gated correction is most useful when the failed trajectory contains recoverable reasoning structure. If the original response has no useful partial reasoning, feedback-based recovery may be less effective.

6 Conclusion

This project presents Feedback-Augmented GRPO with Policy Shaping, a method for reusing failed reasoning trajectories during GRPO post-training. The method detects all-failed groups, uses external natural-language feedback to revise failed trajectories, accepts only verifier-approved corrections, and applies policy shaping to strengthen learning from corrected trajectories.

Experiments show that FB-GRPO consistently improves over standard GRPO and self-feedback across all five validation sets at the final checkpoint. Failure-recovery diagnostics show that external feedback recovers many more all-failed questions than self-feedback. A policy-shaping ablation further shows that shaped optimization improves over feedback-based replacement alone. The main takeaway is that failure-only samples should not always be discarded. With feedback, verification, and shaped optimization, all-failed groups can become useful training signal for mathematical reasoning.

7 Team Contributions

- **Chenyue Li:** I am the sole member of this project. I designed the project, implemented the GRPO training and logging pipeline, set up `GSM8K` training, implemented the feedback-based recovery mechanism, ran experiments comparing standard GRPO, self-feedback, and feedback-augmented GRPO, analyzed the quantitative and qualitative results, and wrote the final report.

Changes from Proposal The final project keeps the original objective of converting failed GRPO trajectories into useful training signal through external natural-language feedback. The main methodological extension is policy shaping for verifier-approved corrected trajectories, which strengthens the feedback-based recovery pipeline by helping the model learn low-probability but verified-correct reasoning patterns more effectively. I also expanded the empirical analysis by reporting final-checkpoint

transfer performance on five validation sets, failure-recovery diagnostics, a policy-shaping ablation, and qualitative examples of feedback-guided correction.

References

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073* (2022).

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

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948* (2025).

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

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021a. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874* (2021).

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. Measuring mathematical problem solving with the MATH dataset. *arXiv preprint arXiv:2103.03874* (2021).

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, et al. 2025. Training language models to self-correct via reinforcement learning. In *The Thirteenth International Conference on Learning Representations*.

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

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. 2022b. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems* 35 (2022), 3843–3857.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems* 36 (2023).

Mathematical Association of America. 2023. American Mathematics Competitions. `https://maa.org/student-programs/amc/`. Accessed: 2026-06-08.

OpenAI. 2025. GPT-5 is here. `https://openai.com/gpt-5/`.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems* 35 (2022), 27730–27744.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, YK Li, Y Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300* (2024).

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. HybridFlow: A flexible and efficient RLHF framework. *arXiv preprint arXiv:2409.19256* (2024).

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems* 36 (2023).

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

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. 2024. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122* (2024).