

TRACE-GRPO: Teacher-Refined Advantage Credit Estimation for Long-Horizon Agentic RL

Yuchen Liu^{1,3}, Tianchu Yao³, Deshu Chen^{1,2}, Yinghao Zhang^{1,4}, Xigui Li^{1,2},
Zehui Ling¹, Xin Guo², Yuan Cheng^{1,2}, Chao Qu^{1,2,*}, Lintao Ma^{3,*}, Yuan Qi^{1,2,*}

¹Fudan University, Shanghai, China

²Shanghai Academy of Artificial Intelligence for Science, Shanghai, China

³Ant Group

⁴Shanghai Innovation Institute, Shanghai, China

liuyuchen23@m.fudan.edu.cn {quchao,qiyuan}@fudan.edu.cn lintaoma@gmail.com

*Corresponding authors.

Abstract

Group-Relative Policy Optimization (GRPO) is widely used to post-train LLM agents, but its trajectory-level advantage is ill-suited to long-horizon credit assignment. Reward-homogeneous rollout groups yield zero advantage; otherwise, one scalar uniformly scales every token, obscuring which decisions determine the outcome. We propose **TRACE-GRPO** (Teacher-Refined Advantage Credit Estimation for GRPO), which converts teacher-LLM feedback into hierarchical credit. Teacher scores provide a sample-level fallback for zero-variance groups, reweight trajectory advantage at the turn level, and redistribute turn credit at the token level using the policy’s log-probability shift under a teacher critique. This yields dense training-time credit without token-level annotations or learned critics. We also introduce **CAPA** (Critique-Aware Paged Attention), which reuses trajectory KV caches and reduces the isolated critique-conditioned forward from roughly $2\times$ a vanilla GRPO step to approximately 5% of a training step. On ALFWorld and ScienceWorld with Qwen2.5-7B-Instruct, TRACE-GRPO achieves 96.3% and 73.4% success, respectively, outperforming strong agent RL baselines. Ablations show cumulative gains from the three credit levels, and comparison with supervised fine-tuning on teacher-generated successes confirms that critique-guided policy optimization provides benefits beyond imitation. Code is available at <https://github.com/Lau-Jonathan/TRACE-GRPO>.

1 Introduction

Large language model (LLM) agents are increasingly used in interactive, long-horizon environments where success depends on a sequence of grounded decisions rather than a single response (Yao et al., 2023; Shridhar et al., 2021;

Wang et al., 2022). Reinforcement learning can post-train such agents, but sparse and delayed rewards make credit assignment difficult. Group-Relative Policy Optimization (GRPO) (Shao et al., 2024) normalizes rewards within sampled rollout groups without an auxiliary value function. Its trajectory-level scalar advantage, however, creates two problems in long-horizon tasks. Under reward homogeneity, the group-normalized advantage is zero even when a rollout contains local mistakes. When reward variance exists, the same scalar is applied to every response token, so erroneous actions receive the same update magnitude as punctuation, boilerplate, and formatting tokens. The update therefore does not distinguish incidental positions from outcome-relevant decisions.

Prior work adds denser supervision to mitigate sparse rewards. Process reward models and step-wise evaluators score intermediate reasoning (Uesato et al., 2022; Lightman et al., 2024), while LLM feedback methods use natural-language critiques to explain failures or guide revision (Bai et al., 2022; Madaan et al., 2023; Shinn et al., 2023; Wang et al., 2026a; Song et al., 2026). These methods show that teacher feedback contains information missing from environment rewards, but it is often reduced to trajectory- or turn-level scalars (Wang et al., 2026a; Song et al., 2026). This reduction loses the specific object errors, missing preconditions, and invalid actions named in the critique. Direct token labels or learned token critics could retain this information, but they are costly to obtain, difficult to define for open-ended agent responses, and introduce additional modeling assumptions.

We propose TRACE-GRPO, which uses natural-language teacher feedback to refine GRPO advantages at the sample, turn, and token levels. Sample-level fallback scores recover updates for zero-

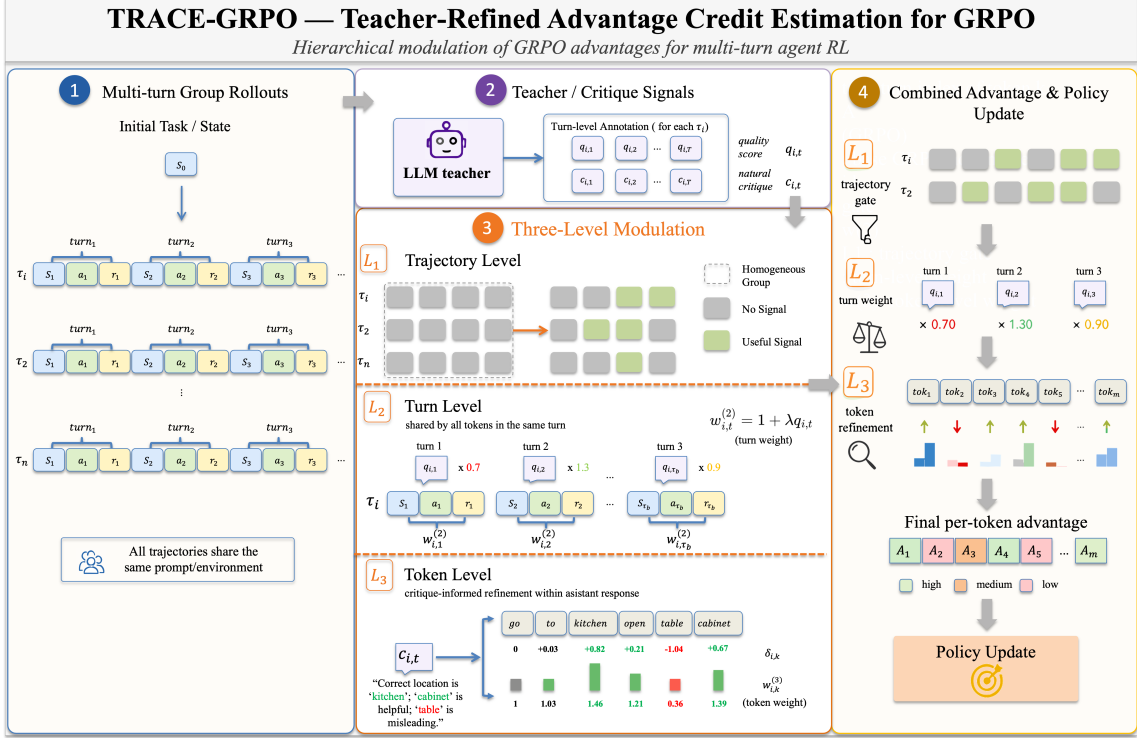


Figure 1: Overview of TRACE-GRPO. A teacher LLM provides turn-level quality scores and natural-language critiques. TRACE-GRPO converts this feedback into sample-level fallback advantages, turn-level reweighting, and token-level redistribution through the critique-induced log-probability shift δ .

variance groups, turn-level judgments reweight individual decisions, and critique-induced log-probability shifts redistribute credit across realized tokens without token labels or learned token critics. We also develop CAPA (Critique-Aware Paged Attention), which reuses trajectory prefix KV caches and reduces the isolated Level-3 conditioned forward to approximately 5% of a training step. With Qwen2.5-7B-Instruct, TRACE-GRPO achieves 96.3% success on ALFWorld (+5.5 pt over GiGPO; +18.7 pt over GRPO) and 73.4% on ScienceWorld (+24.3 pt over GRPO). We further analyze the affected token categories, optimization dynamics, and performance by task difficulty. Our contributions are:

1. We propose **TRACE-GRPO**, a hierarchical advantage estimator that integrates teacher feedback at the sample, turn, and token levels. Its token-level signal uses critique-induced log-probability shifts to extract dense credit from natural-language critiques without token labels or a learned token critic.
2. We introduce **CAPA**, an efficient training-time implementation that reuses KV caches and paged attention to make critique-conditioned

token scoring practical.

3. We evaluate TRACE-GRPO on ALFWorld and ScienceWorld and analyze token-level shifts, optimization dynamics, and performance by task difficulty.

2 Related Work

GRPO and sub-trajectory credit assignment. GRPO (Shao et al., 2024) replaces a learned value function with group-normalized rewards, simplifying large-scale RL for reasoning and agentic LLMs. Recent variants improve stability through normalization, filtering, or baseline estimation, including DR-GRPO (Liu et al., 2025) and StarPO-S (Wang et al., 2025), but still rely on trajectory-level advantages. Other methods refine credit assignment within trajectories: GiGPO (Feng et al., 2026) uses shared-prefix rollout structure to derive per-state advantages, and ArChER (Zhou et al., 2024) uses a hierarchical actor-critic formulation. BEACON (Wang et al., 2026b) instead decomposes long-horizon returns around environment-defined milestones. GiGPO provides the closest value-free comparison because it shares GRPO as the parent optimizer and uses the same benchmark pro-

tokens. Methods based on learned critics, extra branch rollouts, or task-specific milestone detectors use different supervision or compute. In this context, TRACE-GRPO preserves GRPO’s value-free structure while using teacher critiques and policy distribution shifts to refine credit at the sample, turn, and token levels.

Teacher feedback and token-level credit. RLHF (Ouyang et al., 2022) and RLAIIF (Bai et al., 2022) obtain preference or reward signals from human or AI feedback, typically at the response level. Process-reward and textual-feedback methods provide denser supervision by scoring intermediate steps or turns (Wang et al., 2026a), but often compress critiques into scalar rewards over coarse segments (Song et al., 2026). This loses localization information in the teacher’s text, such as the wrong object, missing precondition, or invalid action. TRACE-GRPO keeps scalar teacher scores for sample- and turn-level modulation, while using critique-induced log-probability shifts to obtain token-level credit without token labels or a learned token critic.

3 Method

3.1 Problem Setup

We consider multi-turn agent RL. At turn t , the policy π_θ generates an action a_t , represented as a sequence of tokens, conditioned on the history

$$h_t = (o_1, a_1, o_2, \dots, o_{t-1}, a_{t-1}, o_t)$$

where o_t is the environment observation. The environment returns the next observation o_{t+1} and a terminal reward R at the end of the episode. A trajectory τ is the resulting interleaved sequence of observations and generated actions.

GRPO (Shao et al., 2024) samples N trajectories for each prompt and normalizes rewards within the rollout group:

$$A_{\text{GRPO}}(\tau_i) = \frac{R(\tau_i) - \mu_G}{\sigma_G + \varepsilon_{\text{std}}} \quad (1)$$

for $i \in \{1, \dots, N\}$, where μ_G and σ_G are the group mean and standard deviation, and $\varepsilon_{\text{std}} > 0$ is a numerical-stability constant. The resulting trajectory-level advantage is broadcast to every generated response token:

$$A(i, k) = A_{\text{GRPO}}(\tau_i) m(i, k)$$

where $m(i, k) \in \{0, 1\}$ masks assistant-generated response tokens. This trajectory-level design induces two credit-assignment failures in long-horizon agents. First, under reward homogeneity, group-normalized advantages collapse to zero, paralyzing the policy gradient. Second, within reward-variant groups, a single advantage scalar is indiscriminately applied to all constituent tokens, blinding the model to the decision responsible for the outcome. To bridge this gap, TRACE-GRPO retains GRPO’s value-free architecture while refining advantages across three structural granularities: sample, turn, and token.

3.2 Teacher Annotation Protocol

After each rollout, a frozen teacher LLM T annotates the trajectory using the prompt in Appendix I. The teacher is instructed to label only turns whose decisions are grounded in observable trajectory evidence (state transitions, score changes, parser messages, or logically necessary preconditions). Routine or ambiguous turns are omitted. We enforce a strict JSON schema and a fixed score/confidence vocabulary; malformed output becomes an abstention. Thus $q_{b,t}$ is an evidence-grounded training signal, not a claim of a unique causal label: in a long-horizon environment, delayed effects make such labels generally unavailable from the environment itself.

Teacher requirements and selection. TRACE does not require task-specific teacher training: we use frozen, general-purpose instruction models without benchmark-specific teacher fine-tuning. A suitable teacher should follow the annotation schema, ground judgments in environment evidence, abstain when the evidence is insufficient, and provide decision-specific rationales. In practice, candidate teachers can be screened on a small held-out trajectory set for schema validity, outcome agreement, confidence and abstention behavior, and rationale specificity. These diagnostics assess suitability but do not make the teacher’s turn labels causal ground truth.

For trajectory τ_b , the teacher returns a sparse set of labeled turns:

$$T(\tau_b) = \{(d_{b,t}, c_{b,t}, r_{b,t})\}_{t \in \mathcal{T}_b} \quad (2)$$

where $\mathcal{T}_b$ is the set of annotated turns, $d_{b,t} \in \{-1.0, -0.7, -0.3, 0.3, 0.7, 1.0\}$ is the decision-quality score, $c_{b,t} \in \{0.5, 0.75, 1.0\}$ is the teacher confidence, and $r_{b,t}$ is a concise rationale. Neutral

turns are omitted rather than assigned a zero label, which keeps teacher output focused on outcome-relevant decisions. We define the signed turn signal

$$q_{b,t} = d_{b,t} c_{b,t} \in [-1, 1] \quad (3)$$

for $t \in \mathcal{T}_b$. For unlabeled turns, we set $q_{b,t} = 0$, corresponding to teacher abstention. The rationale $r_{b,t}$ is used only as a training-time conditioning signal for Level 3. The teacher also returns a trajectory-level success judgment for diagnostics, but this flag is not used in the policy-gradient objective.

3.3 Level 1: Zero-Variance Fallback

Level 1 addresses reward-homogeneous rollout groups. When the group reward variance is below the numerical threshold, GRPO provides no learning signal. Before applying a fallback, we compute the mean signed teacher score $\bar{q}_G$ over the group. A group-consistency gate g_G is enabled only when $\bar{q}_G > 0$ for an all-success group or $\bar{q}_G < 0$ for an all-failure group; otherwise $g_G = 0$. We replace the zero trajectory advantage with a teacher-derived turn signal only when this gate passes:

$$A^{(1)}(b, k) = \begin{cases} \lambda q_{b, \iota_b(k)} m(b, k) & g_G = 1 \\ A_{\text{GRPO}}(\tau_b) m(b, k) & \text{otherwise} \end{cases} \quad (4)$$

where $\iota_b(k)$ maps token k to its containing turn, and $\lambda > 0$ controls the fallback scale. If the teacher abstains on a turn, then $q_{b, \iota_b(k)} = 0$ and Level 1 assigns no advantage to tokens in that turn. Thus, Level 1 recovers a learning signal only where the teacher identifies outcome-relevant decisions and the aggregate teacher direction agrees with the environment outcome. A conflicting fallback is rejected, so the group retains vanilla GRPO’s zero update rather than receiving an outcome-opposed signal.

3.4 Level 2: Turn-Level Critique Modulation

Level 2 reweights the update at turn granularity according to whether the teacher agrees with the GRPO direction. Let

$$s_b = \text{sgn}(A_{\text{GRPO}}(\tau_b)) \in \{-1, 0, +1\}$$

$$a_{b,k} = s_b q_{b, \iota_b(k)}$$

A positive $a_{b,k}$ means the teacher’s assessment of the turn agrees with the trajectory-level update direction; a negative value means the teacher dissents. The turn-level weight and advantage are

$$w_{\text{crit}}(b, k) = \text{clip}(1 + \alpha a_{b,k}, 0, 1 + \alpha) \quad (5)$$

$$A^{(2)}(b, k) = A^{(1)}(b, k) w_{\text{crit}}(b, k) \quad (6)$$

where $\alpha \geq 0$ controls modulation strength. Agreement amplifies the update, while dissent dampens it. The lower clamp prevents the teacher signal from reversing the sign of the GRPO update. For zero-variance groups, $s_b = 0$, so $w_{\text{crit}}(b, k) = 1$ and the Level 1 fallback passes through unchanged. Tokens associated with environment-detected format errors skip this reweighting and use $w_{\text{crit}}(b, k) = 1$, preserving the environment penalty independent of teacher leniency.

3.5 Level 3: Critique-Conditioned Token-Level Reweight

3.5.1 Definition

Levels 1 and 2 operate at turn granularity. Level 3 further redistributes credit within each annotated response by measuring how the policy’s own token probabilities change when conditioned on the teacher rationale. For turn t and response token $y_{b,t,j}$, we compute

$$\delta_{b,t,j} = \text{sg} \left[\log \pi_{\theta}(y_{b,t,j} \mid h_{b,t}, r_{b,t}, y_{b,t,<j}) - \log \pi_{\theta}(y_{b,t,j} \mid h_{b,t}, y_{b,t,<j}) \right] \quad (7)$$

where $h_{b,t}$ is the turn history and $j = 1, \dots, L_{\text{resp}}(b, t)$. The operator $\text{sg}[\cdot]$ denotes stop-gradient. In implementation, δ is computed once under the rollout policy and treated as a fixed credit-assignment signal during the policy update. For unlabeled turns, we set the Level 3 weight to one and do not compute δ .

The magnitude $|\delta_{b,t,j}|$ measures how strongly the rationale affects the likelihood of the realized token. Its sign indicates whether the rationale increases or decreases that likelihood. To align token reweighting with the current update direction, denote

$$u_{b,k} = \text{sgn}(A^{(2)}(b, k))$$

For token k in turn $\iota_b(k)$ and within-turn position $j_b(k)$, the token weight is

$$w_{b,k} = 1 + \alpha_3 u_{b,k} |q_{b, \iota_b(k)}| \text{clip}(\delta_{b, \iota_b(k), j_b(k)}, -\kappa, \kappa) \quad (8)$$

where $\alpha_3 \geq 0$ is the token-level modulation strength and $\kappa > 0$ clips the log-probability shift. We choose $\alpha_3 \kappa < 1$, which ensures $w_{b,k} > 0$ for all response tokens and prevents sign flips of the underlying advantage.

Assuming the trajectory contains at least one response token, the Level 3 advantage is

$$A^{(3)}(b, k) = A^{(2)}(b, k) \frac{w_{b,k}}{\bar{w}_\delta(b)} \quad (9)$$

where

$$\bar{w}_\delta(b) = \frac{\sum_k w_{b,k} m(b, k)}{\sum_k m(b, k)} \quad (10)$$

The row-mean normalization makes Level 3 a credit-redistribution operator: it changes where the update is concentrated within a trajectory, while limiting changes to the trajectory-level scale.

3.5.2 Properties

Level 3 is designed as a controlled token-level redistribution rather than an unconstrained rescaling of the trajectory advantage. Let $N_b = \sum_k m(b, k) > 0$, and let Cov_m and Var_m denote covariance and variance under the uniform distribution over masked response tokens. Writing $A_k = A^{(2)}(b, k)$ and $w_k = w_{b,k}$, the row-mean normalization in Eq. (9) gives

$$\begin{aligned} & \sum_k A^{(3)}(b, k) m(b, k) \\ &= \sum_k A_k m(b, k) + \frac{N_b \text{Cov}_m(A, w)}{\bar{w}_\delta(b)} \end{aligned} \quad (11)$$

Thus, Level 3 exactly preserves the masked trajectory total whenever $\text{Cov}_m(A, w) = 0$. This includes the case where $A^{(2)}(b, \cdot)$ is constant over response tokens, which covers broadcast advantages and trajectories whose turn-level multiplier is uniform.

When the covariance is nonzero, the deviation is bounded. Let $a = \alpha_3 \kappa$. Since $|w_k - 1| \leq a$ and $\bar{w}_\delta(b) \geq 1 - a > 0$, we have

$$\begin{aligned} & \left| \sum_k A^{(3)} m - \sum_k A^{(2)} m \right| \\ & \leq \frac{N_b a}{1 - a} \cdot \sqrt{\text{Var}_m(A^{(2)})} \end{aligned} \quad (12)$$

The derivation is provided in Appendix A. In our experiments, $a = 0.3$, so Level 3 remains close to pure redistribution while allowing token-specific amplification and damping.

Soft fallback. If the teacher abstains on all turns, then $q_{b,t} = 0$, $w_{b,k} = 1$, and $A^{(3)} = A^{(2)}$. More generally, if $w_{b,k}$ is constant over the response-token mask, the normalization in Eq. (9) cancels exactly. Level 3 therefore degrades gracefully when the critique contains no token-specific information.

Ablation chain. Setting $\alpha_3 = 0$ recovers the L1+L2 variant. Setting the Level 2 coefficient to zero further recovers L1. Setting the Level 1 fallback coefficient to zero recovers vanilla GRPO, including its zero gradient on reward-homogeneous rollout groups. Each level can therefore be disabled independently.

Policy update. All teacher-derived quantities, including q , δ , and the resulting advantages, are detached during optimization. We replace the GRPO advantage in the standard clipped policy-gradient loss with $A^{(3)}$:

$$\begin{aligned} \mathcal{L}_{\text{PG}} = & -\mathbb{E}_{b,k} \left[m(b, k) \right. \\ & \left. \min(\rho_{b,k} A^{(3)}(b, k), \rho_{b,k}^\epsilon A^{(3)}(b, k)) \right] \end{aligned} \quad (13)$$

where $\rho_{b,k}$ is the policy ratio, $\pi_\theta(y_{b,k} | h_{b,k}) / \pi_{\text{old}}(y_{b,k} | h_{b,k})$, and $\rho_{b,k}^\epsilon$ denotes its clipped value, $\text{clip}(\rho_{b,k}, 1 - \epsilon, 1 + \epsilon)$. The KL penalty and optimizer details follow Appendix B.

3.6 CAPA: Efficient Critique-Conditioned Forwards

Naively computing the critique-conditioned term in Eq. (7) requires one extra full forward per annotated turn. CAPA (Critique-Aware Paged Attention) reduces this cost by reusing the trajectory prefix KV cache from the standard log-probability pass. Since annotated turns share nested prefixes, CAPA treats the trajectory cache as a paged store (Kwon et al., 2023): shared prefix pages are reused, while each turn appends a private stub containing its rationale and response tokens. A single batched FlashAttention-2 paged call (Dao, 2024) then evaluates all critique-conditioned continuations using per-turn `block_table` and `cu_seqlens` indices (Figure 2).

CAPA matches the naive conditioned forward within an absolute tolerance ($\text{atol} = 10^{-5}$) in FP32. With BF16 and FlashAttention-2, the maximum observed numerical drift is below $\text{atol} = 4 \times 10^{-3}$, well below typical critique-induced shifts. On Qwen2.5-7B-Instruct with 8 GPUs, CAPA reduces Level-3 forward cost by $12.7\times$, from 178 to 14 seconds per step. The resulting isolated Level-3 forward accounts for approximately 5% of a training step; this figure excludes the separate asynchronous teacher-annotation loop, whose end-to-end cost is reported in Appendix E. Implementation details and FSDP integration are in Appendix H.

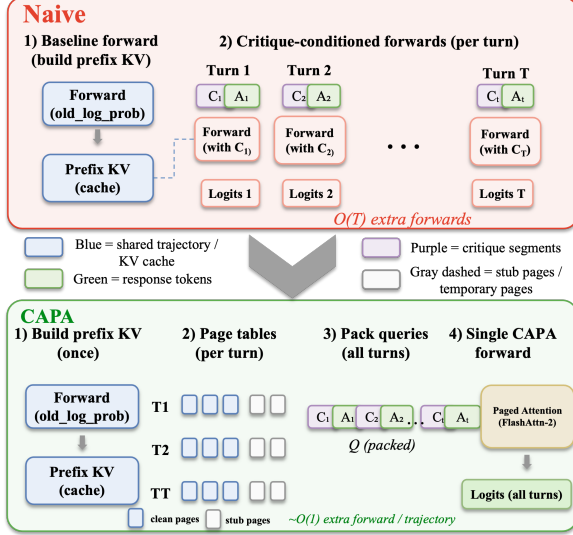


Figure 2: CAPA architecture. The trajectory KV cache is partitioned into shared prefix pages. Each annotated turn appends a private stub containing its rationale and response tokens. Per-turn block_table indices concatenate shared pages with the stub page, allowing all critique-conditioned continuations to be evaluated in one paged-attention call. Due to space constraints, a simplified illustration is shown here; the full, detailed architecture diagram is provided in Appendix H.

4 Experiments

4.1 Setup

We evaluate TRACE-GRPO using Qwen2.5-7B-Instruct on two long-horizon agent benchmarks, ALFWorld (Shridhar et al., 2021) and ScienceWorld (Wang et al., 2022), strictly following the GiGPO (Feng et al., 2026) protocol. Training is conducted on $8 \times H100$ GPUs with a frozen DeepSeek-V3.1 teacher guiding the turn-level critique derivation. Full implementation details (including hyperparameters, rollout settings, and dataset construction) are deferred to Appendix B. We compare TRACE-GRPO against leading closed-source models, prompting agents, and strong reinforcement learning baselines (including PPO (Schulman et al., 2017), RLOO (Ahmadian et al., 2024), vanilla GRPO (Shao et al., 2024), and GiGPO (Feng et al., 2026)). To isolate the gains of RL optimization from mere teacher imitation, we also evaluate **Student-SFT** and **Teacher-Prompt** controls. Detailed descriptions of all baselines are deferred to Appendix B.

4.2 Main Results

As shown in Table 1, TRACE-GRPO achieves **96.3%** overall success on ALFWorld, improv-

ing over GiGPO+std (90.8%) by +5.5 points and over vanilla GRPO by +18.7 points. The largest gains over GiGPO+std occur on decision-intensive subtasks, including Look (+17.3 points), Pick2 (+14.9 points), Heat (+12.0 points), and Cool (+10.7 points). On ScienceWorld, TRACE-GRPO obtains a score of **70.6** and a success rate of **73.4%**, exceeding vanilla GRPO by +8.8 score points and +24.3 success points. Relative to GiGPO+std, the score gain is small (70.6 vs. 69.2), whereas the success-rate gain is substantially larger (+20.0 points). Overall, the results indicate that converting teacher feedback into hierarchical credit signals improves long-horizon agent training beyond standard group-relative reward normalization.

Evaluation uncertainty. For each fixed final checkpoint, we perform three complete evaluation passes over every held-out task at temperature 0.4. Standard deviations across the three passes quantify decoding-time variability, not training-seed variability. We additionally average the three outcomes for each task and bootstrap tasks to obtain 95% evaluation intervals. TRACE obtains 96.3 ± 0.8 ALFWorld success, 70.6 ± 1.3 ScienceWorld score, and 73.4 ± 1.6 ScienceWorld success. Appendix C reports intervals for all methods and ALFWorld subtask cells. The intervals support the large overall success gains, while the 70.6 versus 69.2 ScienceWorld score difference remains a small numerical improvement.

SFT vs. RL. Teacher-Prompt obtains 57.1% ALFWorld success, and Student-SFT obtains 61.4%, compared with TRACE-GRPO at 96.3%. The same pattern appears on ScienceWorld, where Teacher-Prompt and Student-SFT remain below the RL-trained agents. Because Student-SFT uses only successful teacher trajectories, its lower results show that success-only imitation does not reproduce TRACE-GRPO’s gains. TRACE-GRPO instead learns from the student’s on-policy successes and failures and uses the teacher critiques to refine their advantages.

Different student, teacher, and task family. We further train on the τ^2 -Bench Telecom tool-use environment (Barres et al., 2025) using Ling-2.6-flash (104B total, 7.4B active parameters) (Li et al., 2026) as the student and Claude Sonnet 4.5 as the teacher. This simultaneously changes the student scale and architecture, teacher family, and environment. After 100 updates, full TRACE improves

Type	Method	ALFWorld							ScienceWorld	
		Pick	Look	Clean	Heat	Cool	Pick2	All	Score	Succ
Closed-source prompting baselines										
Closed-src	GPT-4o (ReAct) (Hurst et al., 2024)	75.3	60.8	31.2	56.7	21.6	49.8	48.0	54.3	45.4
Closed-src	Gemini-2.5-Pro (ReAct) (Comanici et al., 2025)	92.8	63.3	62.1	69.0	26.6	58.7	60.3	42.5	35.9
Teacher-only control										
Teacher	DeepSeek-V3.1 Prompt	82.86	69.23	51.85	31.25	36.00	58.33	57.14	13.05	21.88
Base model: Qwen2.5-7B-Instruct										
Prompt	Direct Prompt (Yang et al., 2024)	33.4	21.6	19.3	6.9	2.8	3.2	14.8	11.4	4.2
Prompt	ReAct (Yao et al., 2023)	48.5	35.4	34.3	13.2	18.2	17.6	31.2	17.4	7.8
Prompt	Reflexion (Shinn et al., 2023)	62.0	41.6	44.9	30.9	36.3	23.8	42.7	23.4	11.7
SFT	Student-SFT (Qwen2.5-7B-Instruct ← DeepSeek-V3.1)	62.86	53.85	59.26	75.00	48.00	70.83	61.43	19.09	14.06
RL	PPO (critic) (Schulman et al., 2017)	92.3	64.0	92.5	89.5	80.3	68.8	80.4	37.1	24.0
RL	RLOO (Ahmadian et al., 2024)	87.6	78.2	87.3	81.3	71.9	48.9	75.5	–	–
RL	GRPO (Shao et al., 2024)	90.8	66.1	89.3	74.7	72.5	64.7	77.6	61.8	49.1
RL	GiGPO+std (Feng et al., 2026)	97.7	82.7	98.8	83.7	89.3	79.2	90.8	69.2	53.4
RL	GiGPO-std (Feng et al., 2026)	91.8	88.6	95.9	90.2	86.5	85.2	90.2	–	–
TRACE-GRPO ablations										
Ours	TRACE-GRPO (L1 only)	91.5	67.8	89.7	76.0	74.1	66.5	78.6	62.0	49.5
Ours	TRACE-GRPO (L1+L2)	94.5	91.7	92.0	87.5	92.4	83.6	90.3	50.8	80.5
Ours	TRACE-GRPO (L1+L2+L3)	95.8 ^{+5.0}	100.0 ^{+33.9}	93.6 ^{+4.3}	95.7 ^{+21.0}	100.0 ^{+27.5}	94.1 ^{+29.4}	96.3 ^{+18.7}	70.6 ^{+8.8}	73.4 ^{+24.3}

Table 1: Main results on ALFWorld and ScienceWorld using Qwen2.5-7B-Instruct as the student backbone. ALFWorld reports per-subtask and overall success (%); ScienceWorld reports task score and success (%). Bold indicates the best result; gains in the final row are absolute improvements over vanilla GRPO; N/A denotes unavailable matched-protocol results.

all five environment-grounded accuracy and efficiency measures. At update 75, L1+L2 improves mean reward by 3.6% over GRPO, while adding L3 raises reward per token by a further 3.4%, shortens responses by 4.5%, and reduces environment steps by 4.2%. The three-way comparison separates the L1+L2 gain from the additional effect of L3. Figure 4 reports the complete matched training curves together with the update-100 endpoints and relative changes. Full protocol and teacher-agreement statistics are in Appendix F.

4.3 Teacher Feedback Analysis

Figure 8 in Appendix D analyzes DeepSeek-V3.1 feedback during ALFWorld training. Teacher–environment outcome agreement is already $93.9 \pm 2.6\%$ in iterations 0–30 and rises to $96.8 \pm 1.5\%$ thereafter. The high initial value is expected because this diagnostic asks only whether the teacher correctly reads the explicit terminal outcome; it does not certify every turn label. Early feedback is also more conservative: 21.4% of labels use the lowest allowed confidence, versus 8.7% later. Negative labels dominate early training, become majority positive by iteration 100, and the dominant aspect shifts from format errors to action choice. Malformed JSON (0.8% in this run) automatically abstains and falls back to the underlying GRPO signal. The changing label polarity and aspect distribution show that the annotations vary with the student’s trajectories rather than repeating the ter-

minal outcome.

4.4 Ablation Study

The bottom rows of Table 1 isolate the three levels of TRACE-GRPO. On ALFWorld, the gains over vanilla GRPO increase from +1.0 point with L1 to +12.7 points with L1+L2 and +18.7 points with L1+L2+L3, showing progressively larger benefits under sparse binary rewards. On ScienceWorld, L1 has little effect on success (49.5% vs. 49.1% for GRPO), as dense task scores reduce zero-variance groups. L2 raises success to 80.5%, while L3 increases the dense task score from 50.8 to 70.6 and remains far above GRPO in success (73.4% vs. 49.1%). The two metrics separate their effects: L2 primarily raises binary success, whereas L3 raises dense task progress and improves policy efficiency.

Where L3 helps. To locate L3’s benefit, we stratify the 128-task split by GRPO per-template score (Figure 6a). On the hardest tertile ($n = 44$), L3 raises score from 20 for L1+L2 to **78** (+58). On the medium ($n = 45$) and easiest ($n = 39$) tertiles, however, score changes from 63 to 51 and from 78 to 75, respectively. L3 also raises the overall dense score from 50.8 to 70.6 but lowers binary success from 80.5% to 73.4%. Our interpretation is that precise critique localization is most valuable on tasks beyond the student’s existing competence; on easier tasks, residual judge noise may over-concentrate an otherwise adequate update. This is an interpretation of the stratified pattern, not a

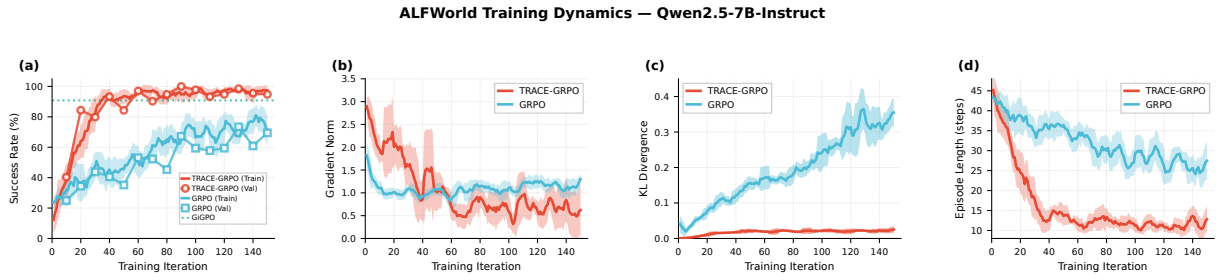


Figure 3: ALFWorld training dynamics with Qwen2.5-7B-Instruct on $8\times H100$ over 150 iterations. **(a)** Train and validation success rates. The dashed line marks the GiGPO reported performance of 90.8%. **(b)** Gradient norm. **(c)** KL divergence from the reference policy. **(d)** Mean episode length. Shaded regions denote $\pm 1\sigma$ within a sliding window.

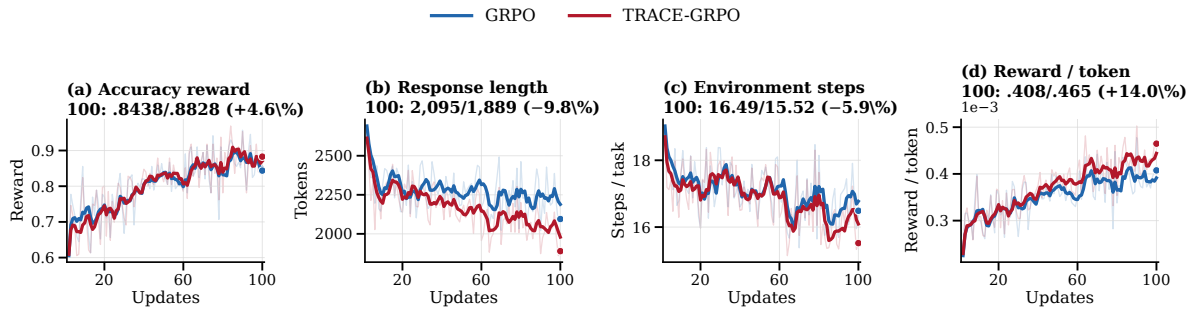


Figure 4: τ^2 -Bench Telecom training curves over the matched first 100 updates. Faint lines show individual updates and solid lines show five-update trailing means. The second line above each panel gives the raw update-100 endpoints as GRPO/TRACE, followed by the relative change; mean reward also rises from 0.8538 to 0.8780 (+2.8%). Higher is better in (a,d), whereas lower is better in (b,c).

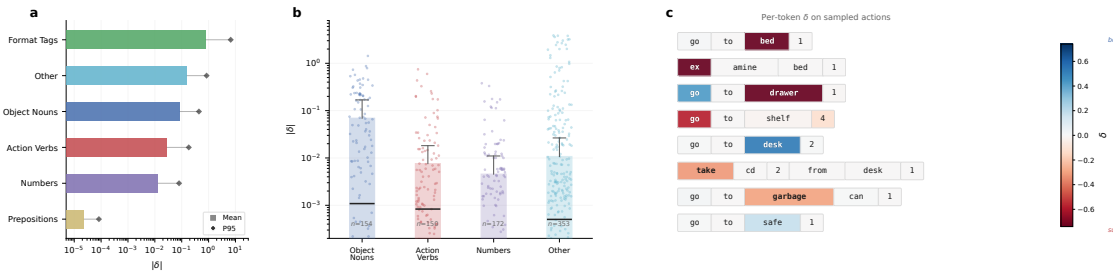


Figure 5: L3 token-level δ modulation analysis on ALFWorld (step 150, $n = 2,289$ tokens). **(a)** Mean and P95 of $|\delta|$ by semantic token category. **(b)** Per-token $|\delta|$ distribution for content categories. **(c)** Qualitative examples with token-level δ overlaid on sampled actions. Red indicates suppression and blue indicates boosting.

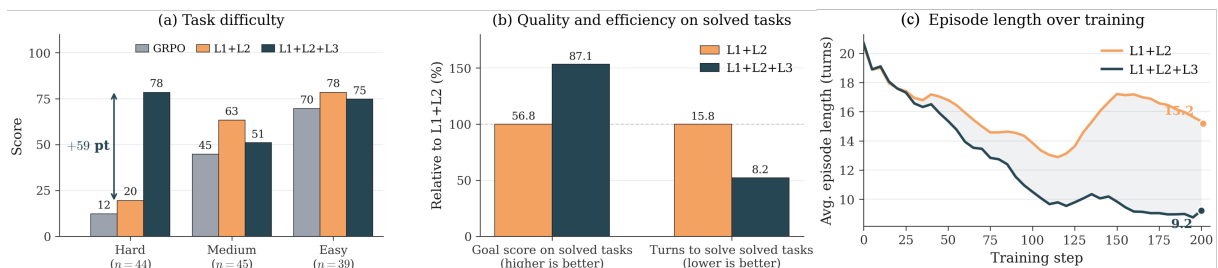


Figure 6: ScienceWorld L3 ablation and efficiency. **(a–b)** Task score, goal score, and trajectory length under L3 ablations. **(c)** Mean validation episode length; L1+L2+L3 converges to 9.2 turns, compared with 15.2 for L1+L2.

causal claim about teacher noise. Among tasks solved by both L1+L2 and L1+L2+L3 (Figure 6b),

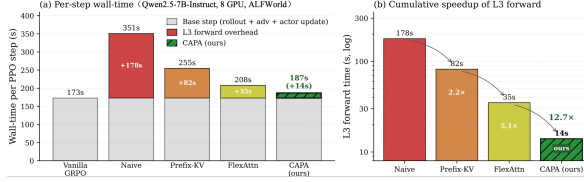


Figure 7: Level-3 critique-conditioned forward cost. (a) Per-step wall time, including the base GRPO step and Level-3 forward overhead. (b) Cumulative Level-3 speedup on a log scale.

L3 achieves a higher goal score (87.1 vs. 56.8) with fewer turns (8.2 vs. 15.8). Over the full validation pool, L3 also shortens episodes to 9.2 turns, compared with 15.2 for L1+L2 (Figure 6). L3’s measured gain is therefore concentrated in difficult-task progress and efficiency rather than longer interaction sequences.

4.5 Training Dynamics

Figure 3 compares TRACE-GRPO with our reproduced GRPO over 150 ALFWorld iterations. TRACE-GRPO reaches 96.3% validation success, compared with 69.5% for GRPO. At convergence, it reduces gradient norm by 76% (2.93 to 0.72), yields 13.6× lower KL divergence from the reference policy (0.028 vs. 0.381), and shortens mean episode length by 64% (29.4 to 10.6 steps). The final checkpoint thus combines higher success with lower reference-policy divergence and shorter trajectories.

4.6 Interpretability of Token-Level Shifts

To examine what L3 modulates, we analyze the step-150 checkpoint on 20 ALFWorld valid-unseen trajectories, covering 302 action turns and 2,289 response tokens. For each token, we compute $\delta = \log \pi(a | h, c) - \log \pi(a | h)$ and group tokens by semantic category. Figure 5 shows that δ concentrates on decision-critical positions. Format boundary tokens have the largest mean $|\delta|$ (0.76), reflecting whether the critique changes the model’s confidence in producing an action response. Among content tokens, object nouns receive the strongest modulation (mean $|\delta| = 0.084$, P95 = 0.44), followed by action verbs (mean $|\delta| = 0.028$, P95 = 0.18). In contrast, prepositions have near-zero shifts ($< 10^{-4}$), indicating that L3 does not uniformly rescale all tokens. Qualitative examples further show that location nouns such as *bed*, *drawer*, and *desk* are strongly suppressed when implicated by the critique, while function words remain largely

unchanged. We interpret δ as a policy-internal sensitivity to an evidence-grounded critique, not as a causal-token estimator.

4.7 Efficiency

CAPA reuses trajectory-prefix KV caches and batches critique-conditioned continuations with paged attention. On Qwen2.5-7B-Instruct with 8-GPU FSDP, it reduces the Level-3 forward from 178 to 14 seconds per step (12.7×), leaving this isolated forward at approximately 5% of a training step. This excludes teacher inference. Figure 7 shows the engineering progression, while the complete end-to-end accounting remains in Appendix E.

5 Conclusion

We have presented TRACE-GRPO, a value-free method that refines GRPO advantages with teacher critiques at the sample, turn, and token levels. Its token-level component reweights realized tokens using critique-induced log-probability shifts without token labels or learned critics, and CAPA makes the additional conditioned forward practical. On ALFWorld, ScienceWorld, and τ^2 -Bench with a larger MoE student, TRACE-GRPO improves over standard GRPO. The ablations separate the effects of L1–L3, including L3’s larger score gain on the hardest ScienceWorld tasks. Student-SFT does not reproduce these gains. Future work will test additional teacher families and domains.

Limitations

TRACE-GRPO has several limitations. First, its benefit depends on a teacher that can follow the annotation schema, read environment evidence, abstain under ambiguity, and produce critiques specific enough to affect the relevant policy decisions. Outcome agreement and confidence calibration test only parts of these requirements; neither proves that every turn judgment is correct. Although the environment reward, Level-1 group-consistency gate, sign-preserving Level 2, positive normalized Level-3 weights, and format-error bypass constrain the teacher’s influence, they cannot eliminate systematic teacher bias.

Second, a student may inherit a teacher’s preferred strategy or phrasing rather than only environment-grounded behavior. The lower Teacher-Prompt and Student-SFT results make surface imitation an incomplete explanation, but do

not rule out this risk. Future work should compare multiple teacher families, smaller local judges, and disagreement-aware ensembles, and should audit whether equivalent actions receive equivalent credit across paraphrases.

Third, our evaluation remains text-based. The additional τ^2 -Bench result extends beyond ALF-World and ScienceWorld to larger-model tool use, but does not establish generality for multimodal control, web/code agents, or mathematical reasoning. Finally, CAPA makes the isolated critique-conditioned forward small, but teacher annotation still incurs latency and monetary cost. Actor-critic extensions and asynchronous or self-critique pipelines may be preferable in highly stochastic domains or when external teachers are unavailable.

Ethical Considerations

Our experiments use synthetic, text-based benchmark environments and do not collect data from human participants or use personally identifying information. Teacher feedback is used only during training; the deployed student does not require access to the teacher. Nevertheless, a teacher can encode systematic preferences or produce incorrect critiques, so we retain environment-grounded rewards, constrain teacher influence with consistency and sign-preserving safeguards, and describe the remaining bias risk in the Limitations section. The public artifact excludes API credentials, infrastructure logs, and proprietary model weights.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant Nos. 82394432 and 92249302) and the Shanghai Municipal Science and Technology Major Project (Grant No. 2023SHZDZX02).

References

- Arash Ahmadian, Chris Cremer, Matthias Gall , Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet  st n, and Sara Hooker. 2024. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12248–12267.
- 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*.
- Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. 2025. τ^2 -Bench: Evaluating conversational agents in a dual-control environment. *arXiv preprint arXiv:2506.07982*.
- Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Naveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.
- Tri Dao. 2024. Flashattention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, volume 2024, pages 35549–35562.
- Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. 2026. Group-in-group policy optimization for llm agent training. *Advances in Neural Information Processing Systems*, 38:46375–46408.
- Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.
- Ang Li, Ben Liu, Bin Han, Bin Hu, Bin Jing, Binbin Hu, Bing Li, Cai Chen, Caizhi Tang, Changxin Tian, et al. 2026. Ling and Ring 2.6 technical report: Efficient and instant agentic intelligence at trillion-parameter scale. *arXiv preprint arXiv:2606.15079*.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let’s verify step by step. In *International Conference on Learning Representations*, volume 2024, pages 39578–39601.
- Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025. [Understanding r1-zero-like training: A critical perspective](#). In *Second Conference on Language Modeling*.
- 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*, 36:46534–46594.

- Long Ouyang, Jeffrey 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*, 35:27730–27744.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- 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*, 36:8634–8652.
- Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. 2021. [ALFWorld: Aligning text and embodied environments for interactive learning](#). In *International Conference on Learning Representations*.
- Yuda Song, Lili Chen, Fahim Tajwar, Rémi Munos, Deepak Pathak, Drew Bagnell, Aarti Singh, and Andrea Zanette. 2026. [Expanding the capabilities of reinforcement learning via text feedback](#). In *Forty-third International Conference on Machine Learning*.
- Jonathan Uesato, Nate Kushman, Ramana Kumar, H Francis Song, Noah Yamamoto Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. 2022. Solving math word problems with process-based and outcome-based feedback.
- Hanyang Wang, Lu Wang, Chaoyun Zhang, Tianjun Mao, Si Qin, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. 2026a. [Text2grad: Reinforcement learning from natural language feedback](#). In *The Fourteenth International Conference on Learning Representations*.
- Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. 2022. Scienceworld: Is your agent smarter than a 5th grader? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11279–11298.
- Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, et al. 2025. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*.
- Zixuan Wang, Yuchen Yan, Hongxing Li, Teng Pan, Dingming Li, Ruiqing Zhang, Weiming Lu, Jun Xiao, Yueting Zhuang, and Yongliang Shen. 2026b. [Milestone-guided policy learning for long-horizon language agents](#). In *Forty-third International Conference on Machine Learning*.
- 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, and 22 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.
- Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023. [React: Synergizing reasoning and acting in language models](#). In *The Eleventh International Conference on Learning Representations*.
- Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. 2024. [Archer: Training language model agents via hierarchical multi-turn RL](#). In *Forty-first International Conference on Machine Learning*.

A Proof of Level-3 Redistribution Bound

Let $A_k = A^{(2)}(b, k)$, $w_k = w_{b,k}$, and let expectations, variances, and covariances be taken uniformly over masked response tokens. Since $\bar{w}_\delta = \mathbb{E}_m[w]$,

$$\begin{aligned} \sum_k A^{(3)}(b, k)m(b, k) &= \frac{N_b \mathbb{E}_m[Aw]}{\mathbb{E}_m[w]} \\ &= \sum_k A_k m(b, k) + \frac{N_b \text{Cov}_m(A, w)}{\bar{w}_\delta} \end{aligned}$$

If $\text{Cov}_m(A, w) = 0$, the masked total is preserved exactly. Otherwise, with $a = \alpha_3 \kappa$, the clipped weight satisfies $|w_k - 1| \leq a$ and $\bar{w}_\delta \geq 1 - a > 0$. Therefore,

$$\begin{aligned} &\left| \sum_k A^{(3)}m - \sum_k A^{(2)}m \right| \\ &\leq \frac{N_b}{\bar{w}_\delta} \sqrt{\text{Var}_m(A)} \sqrt{\text{Var}_m(w)} \\ &\leq \frac{N_b a}{1 - a} \sqrt{\text{Var}_m(A^{(2)})} \end{aligned}$$

B Implementation Details

Hardware and software. For the main ALFWorld and ScienceWorld experiments, training uses 8×NVIDIA H100 (80 GB) on a single node, with FSDP for the actor and reference, vLLM (FlashAttention-2 backend) for rollout, and Ray for orchestration. We use ver1-agent (GiGPO (Feng et al., 2026) fork) as the trainer scaffold and Qwen2.5-7B-Instruct as the base policy. The teacher (DeepSeek-V3.1) is queried over an OpenAI-style HTTP API with up to 32 concurrent workers per training step.

Hyperparameters. Table 2 lists the hyperparameters used for both benchmarks. The evaluation settings for both ALFWorld and ScienceWorld strictly follow the GiGPO (Feng et al., 2026) protocol where applicable. TRACE-GRPO-specific hyperparameters (λ , α , α_3 , κ , σ_ϵ) were fixed early in development on a small ALFWorld sweep and reused on ScienceWorld without further tuning.

Teacher annotation. At each training step, the DeepSeek-V3.1 teacher is queried at temperature 0 with up to 4096 judge tokens per call and a 240 s timeout. The schema returns, for each emitted turn, a decision quality $\in \{-1.0, -0.7, -0.3, 0.3, 0.7, 1.0\}$, confidence $\in \{0.5, 0.75, 1.0\}$, a categorical aspect, and a free-form rationale; we set q_t to their product and use

	ALFWorld	ScienceWorld
<i>Optimization</i>		
Learning rate	1×10^{-6}	1×10^{-6}
KL loss coefficient	0.01	0.01
KL penalty type	low-var KL	low-var KL
Entropy coefficient	0	0.001
PPO mini-batch size	16	16
PPO micro-batch / GPU	2	2–4
Invalid-action penalty	0.1	0.1
<i>Rollout</i>		
Group size N	8	8
Train batch size	16	16
Val batch size	128	128
Max env steps	50	30
Max prompt length	8192	7000
Max response length	24576	16384
Train rollout temp.	1.0	1.0
Eval rollout temp.	0.4	0.4
<i>TRACE-GRPO</i>		
λ (L1 fallback)	0.2	0.2
σ_ϵ (L1 trigger)	10^{-3}	10^{-3}
α (L2 turn-level)	0.5	0.5
α_3 (L3 token-level)	0.3	0.3
κ (L3 clamp)	1.0	1.0
<i>Trainer</i>		
Total iterations	150	200
Eval frequency	every 5 iters	every 5 iters
Checkpoint frequency	every 20 iters	every 20 iters

Table 2: Hyperparameters for TRACE-GRPO on ALFWorld and ScienceWorld.

omission as abstention. Format-error turns flagged by the environment are pinned to $q_t = 0$ and bypass L2/L3 modulation (§3.4).

Evaluation protocol. ALFWorld evaluation uses the GiGPO protocol: 1 trajectory per task on the 134-game valid-unseen split in each pass, eval rollout temperature 0.4, max 50 environment steps, success counted as the binary won flag returned by the textworld engine. ScienceWorld uses 128 tasks stratified across the 29 task types in the held-out pool, 1 trajectory per task per pass, and max 30 steps. Task score is the mean of $\max(0, \text{score})$, where $\text{score} \in [-100, 100]$ is the engine’s progress signal. For final statistics, we repeat each complete evaluation three times as described in Appendix C.

C Evaluation Uncertainty

For each fixed GRPO, GiGPO, and TRACE checkpoint, we run three complete evaluation passes. Each pass evaluates every held-out task once at temperature 0.4 and yields a benchmark-level metric. The reported standard deviation is computed across these three complete passes and therefore measures stochastic decoding variability for a fixed

checkpoint, not training-run variability.

For the 95% interval, we first average a task’s three rollout outcomes to obtain one value for that task. We then repeatedly draw, with replacement, the same number of tasks as in the benchmark and recompute the task mean. The 2.5th and 97.5th percentiles of these resampled means form the interval. Keeping the repeated outcomes together at the task level avoids treating three evaluations of the same task as three distinct tasks.

D Teacher Safeguards and Level-3 Sensitivity

The teacher prompt requires observable evidence, a fixed schema, a discrete score/confidence vocabulary, and abstention below confidence 0.5. Invalid JSON is assigned $q_t = 0$. Environment format errors are also pinned to $q_t = 0$ and bypass L2/L3 so teacher leniency cannot weaken the environment penalty. At L1, the group-consistency gate in §3.3 rejects a fallback whose aggregate direction conflicts with the common environment outcome. L2 cannot reverse an environment-derived advantage, and L3 uses positive weights with row-mean normalization.

As an early-training check, disabling L2/L3 for the first 30 iterations reaches 95.6% ALFWorld success versus 96.3% without the warm start. Thus the lower early confidence does not create a material dependence on early fine-grained modulation.

We vary one Level-3 parameter at a time around the default using the same Qwen2.5-7B initialization, 150-iteration budget, and all other hyperparameters. The default run is shared by the two sweeps and the existing L1+L2 checkpoint provides the $\alpha_3 = 0$ reference, so four additional runs are required. All nonzero settings satisfy $\alpha_3\kappa < 1$.

All non-default Level-3 settings obtain 94.8–95.8% success, within 1.5 points of the default and above the 90.3% no-L3 reference. The useful region is therefore local but not a narrow isolated configuration.

E End-to-End Training Cost

Table 6 profiles ScienceWorld on 8 H100 GPUs over 36 GRPO steps and 148 TRACE steps. Phase rows are measured wall-clock durations and are not additive: TRACE dispatches a teacher request when each trajectory finishes, up to 32 concurrent workers, so most teacher inference overlaps the remaining rollout work. The total row is the observed

end-to-end optimizer-step time.

Each TRACE step issues 128 teacher requests; mean and maximum request latency are 18.3 and 56.4 seconds, respectively, with 32-way concurrency. The profile contains no failed request or malformed response and averages 1.4 labeled turns per trajectory (median 0). These values explain both cost terms: 67.9 seconds is the teacher loop’s elapsed wall time, while 13.6 seconds is the isolated CAPA forward after annotations are available. The approximately 5% characterization refers only to the latter quantity as a share of a training step, not to the entire teacher-assisted pipeline.

F Larger-Student Tool-Use Evaluation

We use the Telecom domain of τ^2 -Bench (Barres et al., 2025), which requires multi-turn user interaction, executable tools, observation use, and domain-policy compliance. The student is Ling-2.6-flash, an instruct MoE model with 104B total and 7.4B active parameters (Li et al., 2026); the teacher is Claude Sonnet 4.5. This configuration differs from the main Qwen2.5-7B/DeepSeek-V3.1 experiments in student scale and architecture, teacher family, and task family.

At update 100, full TRACE attains 89.8% evaluation success versus 86.7% for GRPO; on the 45-task paired subset the values are 93.4% and 91.4%. Averaged over the final ten training updates, TRACE improves reward per token by 10.8%, shortens responses by 9.2%, and uses 4.1% fewer environment steps. Failed TRACE trajectories also use 10.3% fewer characters, so the efficiency gain is not explained by abandoning more tasks.

To probe teacher-family dependence, we compare Sonnet 4.5 with Opus 4.6 on ten real trajectories. Eight produce valid matched pairs (two Sonnet calls receive HTTP 429 errors). Across matched per-turn scores, sign agreement is 86.4%, Pearson correlation is 0.61, and mean absolute difference is 0.26. Mean per-sample latency is 18.2 seconds for Sonnet and 20.0 seconds for Opus under eight-way concurrency. The teachers are not identical, but usually agree on the positive/negative direction consumed by TRACE.

G SFT and Teacher-Prompt Baselines

SFT data construction. For each benchmark we collect successful trajectories produced by the DeepSeek-V3.1 teacher under the same agent-rollout prompt template used in RL. We sample

Method	ALFWorld Succ.	ScienceWorld Score	ScienceWorld Succ.
GRPO	77.6 $\pm$ 2.8 [72.0, 82.6]	61.8 $\pm$ 2.5 [57.1, 66.3]	49.1 $\pm$ 3.5 [43.4, 54.8]
GiGPO	90.8 $\pm$ 1.3 [87.0, 93.8]	69.2 $\pm$ 1.8 [65.2, 73.0]	53.4 $\pm$ 3.0 [47.8, 59.0]
TRACE-GRPO	96.3 $\pm$ 0.8 [94.1, 98.0]	70.6 $\pm$ 1.3 [67.4, 73.8]	73.4 $\pm$ 1.6 [69.0, 77.6]

Table 3: Repeated evaluation of fixed checkpoints. Values are mean $\pm$ standard deviation across three complete passes; brackets are task-level bootstrap 95% intervals.

Method	Pick	Look	Clean	Heat	Cool	Pick2
GRPO	90.8 $\pm$ 5.1 [82.0,96.0]	66.1 $\pm$ 6.7 [48.0,81.0]	89.3 $\pm$ 5.4 [79.5,95.4]	74.7 $\pm$ 6.9 [59.0,86.5]	72.5 $\pm$ 5.4 [58.5,83.5]	64.7 $\pm$ 7.3 [49.8,77.2]
GiGPO	97.7 $\pm$ 1.6 [93.0,99.5]	82.7 $\pm$ 7.9 [66.0,92.6]	98.8 $\pm$ 1.6 [95.0,99.8]	83.7 $\pm$ 7.2 [68.0,93.0]	89.3 $\pm$ 8.2 [75.0,96.0]	79.2 $\pm$ 6.6 [64.0,89.0]
TRACE-GRPO	95.8 $\pm$ 1.8 [90.0,98.5]	100.0 $\pm$ 0.8 [94.0,100.0]	93.6 $\pm$ 2.4 [87.0,97.2]	95.7 $\pm$ 2.1 [88.5,98.8]	100.0 $\pm$ 0.9 [94.0,100.0]	94.1 $\pm$ 2.5 [87.5,97.8]

Table 4: ALFWorld subtask statistics under the protocol of Table 3. Wider intervals reflect the smaller number of tasks in each category.

α_3	κ	ALFWorld Succ.
0 (L1+L2)	–	90.3
0.15	1.0	94.8
0.30	1.0	96.3
0.45	1.0	95.5
0.30	0.5	95.1
0.30	1.5	95.8

Table 5: Local sensitivity of the Level-3 parameters. The default was selected on ALFWorld and transferred to ScienceWorld without retuning.

Phase	GRPO	TRACE
Rollout	93.1 s	77.9 s
Old log-probability / KL	51.6 s	35.0 s
Advantage estimator	0.0 s	81.7 s
Actor update	99.4 s	66.5 s
Teacher inference loop	–	67.9 s
CAPA L3 forward	–	13.6 s
Total optimizer step	252.8 s	273.5 s

Table 6: Full training-time profile, including the teacher loop. Because teacher requests overlap rollout, phase rows do not sum to the total.

$K = 8$ rollouts per training task, retain only trajectories with $\text{won} = 1$ (ALFWorld) or $\text{final score} \geq 50$ (ScienceWorld), and deduplicate them by trajectory hash. The two pools are merged into a multi-turn SFT parquet keyed by messages, yielding roughly $5\times$ the training-pool size of the RL set after deduplication.

SFT training. We fine-tune Qwen2.5-7B-Instruct with FSDP for 3 epochs at learning rate 1×10^{-4} (LoRA rank 16, $\alpha = 32$), per-device batch 1 with gradient accumulation 16, max sequence length 2048. Training uses the same ver1 multi-turn SFT trainer as the RL stack so the chat template and tokenization are identical to inference time. Evaluation reuses the same protocol as the

Method (update 75)	Reward	Tokens	Steps	R/token
GRPO	0.8477	2,173	16.50	0.000390
TRACE L1+L2	0.8784	2,106	16.57	0.000417
TRACE L1+L2+L3	0.8672	2,011	15.87	0.000431

Table 7: Controlled ablation on τ^2 -Bench Telecom.

RL runs (Appendix B).

Teacher-Prompt baseline. Teacher-Prompt evaluates DeepSeek-V3.1 directly on each benchmark’s held-out split with no fine-tuning, using the same system prompt, action template, and environment as the student. Generation is deterministic ($T = 0$) with a 4096-token budget per turn and the same step limits as RL evaluation.

H CAPA Implementation Details

This appendix gives the full implementation of CAPA, the optimization from §3.6 that brings the critique-conditioned $\delta_{b,t,j}$ in Eq. (7) from one extra full forward per annotated turn down to a single packed forward per trajectory. The key observation is that the standard PPO log-probability pass already runs a full forward over the trajectory x ; CAPA re-uses that pass’s KV cache as the read-only *shared prefix* for all N annotated turns and only adds a packed forward over the new critique + response tokens. We instantiate this idea on top of vLLM’s paged-KV abstraction (Kwon et al., 2023) and FlashAttention-2’s varlen paged kernel (Dao, 2024).

H.1 Notation and turn layout

A trajectory is a token sequence $x = [s_0, a_1, s_1, \dots, s_T, a_T]$ of length L_T . For each annotated turn $t \in \{1, \dots, N\}$ we tokenize the teacher’s rationale to $\mathbf{c}_t = (c_{t,1}, \dots, c_{t,|c_t|})$.

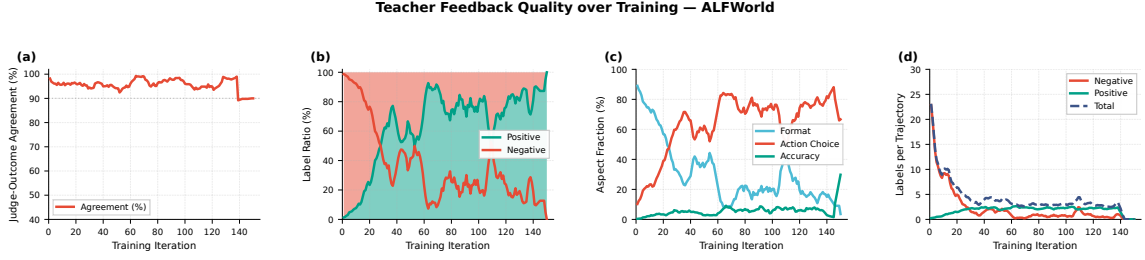


Figure 8: DeepSeek-V3.1 feedback over ALFWorld training. (a) Teacher outcome agreement. (b) Positive and negative label ratio. (c) Dominant feedback aspect. (d) Mean labels per trajectory.

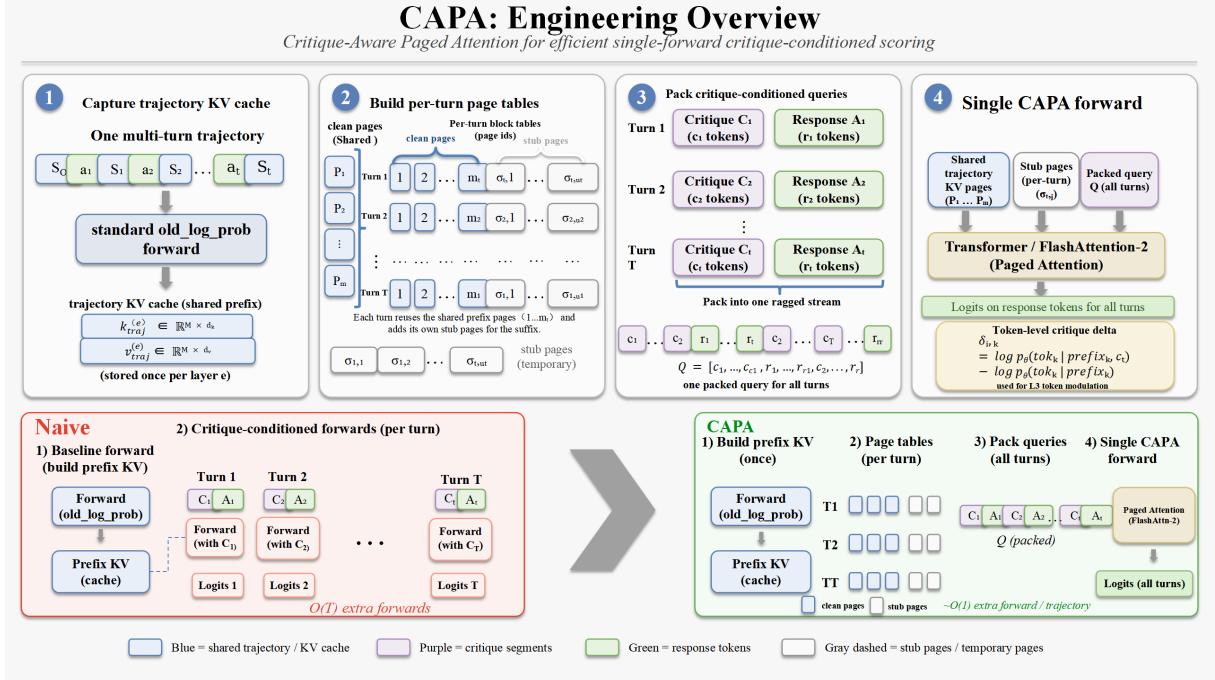


Figure 9: Detailed CAPA architecture. The trajectory KV cache is partitioned into shared prefix pages. Each annotated turn appends a private stub containing its rationale and response tokens. Per-turn block_table indices concatenate shared pages with the stub page, allowing all critique-conditioned continuations to be evaluated in one paged-attention call.

The conditioned input that the reference (slow) implementation feeds the model for turn t is

$$\left[x_{0:\text{ctx_end}_t}, \right. \\ \langle \text{teacher_note} \rangle c_t \langle / \text{teacher_note} \rangle, \\ \left. x_{\text{resp_start}_t:\text{resp_end}_t} \right],$$

where $\text{ctx_end}_t = \text{resp_start}_t$ is the position right after the last environment observation visible to turn t . We write $s_t = |c_t| + |r_t|$ for the number of new tokens turn t contributes (critique + response). Per-token log-probabilities of r_t under this conditioned input are recovered from the model’s output logits at the response slice of the packed forward.

H.2 Page table layout

CAPA pages each layer’s trajectory KV cache into fixed $P=256$ -token pages (the alignment FA2’s

paged kernel expects). Let $n_\tau = \lceil L_\tau / P \rceil$ be the number of trajectory pages. Algorithm 1 (KVTOPAGES) reshapes one layer’s cache from $(1, h_{kv}, L_\tau, d_{\text{head}})$ into $(n_\tau, P, h_{kv}, d_{\text{head}})$.

For each annotated turn t the reads partition into *shared prefix pages* (referred to as “clean pages” below) and *stub pages*:

- **Clean pages.** The first $n_t^c = \lfloor \text{ctx_end}_t / P \rfloor$ trajectory pages are fully occupied by tokens that turn t is allowed to attend to (every token in them lies strictly before ctx_end_t). They are pointed to directly via the page table, with no copying.
- **Stub pages.** The trajectory tail $\tau_t = \text{ctx_end}_t \bmod P$ tokens cannot be read from the clean cache: the same physical page also contains tokens belonging to *later* turns, and FA2’s

Algorithm 1 KVTOPAGES: layer-wise KV $\rightarrow$ paged tensor.

Require: $K, V \in \mathbb{R}^{1 \times h_{kv} \times L_\tau \times d_{head}}$, page size P
1: $\Delta \leftarrow (P - L_\tau \bmod P) \bmod P$; $L' \leftarrow L_\tau + \Delta$
2: Right-pad K, V with zeros along axis 2 to length L'
3: $n_\tau \leftarrow L'/P$
4: $K' \leftarrow K.view(1, h_{kv}, n_\tau, P, d_{head})$
5: $K_{pages} \leftarrow permute(K', (2, 3, 1, 4))$
6: Apply the same view and permutation to obtain V_{pages}
7: **return** $(K_{pages}, V_{pages}, L_\tau)$

Algorithm 2 BUILDPAGE TABLES: per-turn FA2 metadata.

Require: turn layouts $\{(ctx_end_t, s_t, |c_t|, |r_t|)\}_{t=1}^N, n_\tau, P$
1: $stub_cursor \leftarrow 0$
2: **for** $t = 1$ **to** N **do**
3: $n_t^c \leftarrow \lfloor ctx_end_t / P \rfloor$; $\tau_t \leftarrow ctx_end_t \bmod P$
4: $n_t^s \leftarrow \lceil (\tau_t + s_t) / P \rceil$
5: Record $(n_t^c, \tau_t, n_t^s, s_t^{start} = stub_cursor)$
6: $stub_cursor \leftarrow stub_cursor + n_t^s$
7: $cache_seqLens[t] \leftarrow ctx_end_t + s_t$
8: **end for**
9: $W \leftarrow \max_t(n_t^c + n_t^s)$
10: Allocate $block_table \in \mathbb{Z}^{N \times W}$ (zero-init)
11: **for** $t = 1$ **to** N **do**
12: $block_table[t, 0 : n_t^c] \leftarrow [0, 1, \dots, n_t^c - 1]$
13: $a \leftarrow n_\tau + s_t^{start}$
14: $block_table[t, n_t^c : n_t^c + n_t^s] \leftarrow [a, a + 1, \dots, a + n_t^s - 1]$
15: **end for**
16: $cu_seqLens \leftarrow [0, cumsum(s_t)_{t=1}^N]$
17: **return** $(block_table, cache_seqLens, cu_seqLens, \{(n_t^c, \tau_t, n_t^s, s_t^{start})\})$

causal mask is computed at query position granularity, not page granularity, so reading the partial page directly would leak future-turn keys. Stub pages give turn t a private copy of the tail history followed by its own K_{new}/V_{new} .

Turn t allocates $n_t^s = \lceil (\tau_t + s_t) / P \rceil$ stub pages. Algorithm 2 (BUILD PAGE TABLES) builds the per-turn block table and ragged Q/K length vectors. The block table for turn t lists, in order, the n_t^c clean page indices followed by n_t^s stub page indices. Stub pages live in a single tensor of length $n_s = \sum_{t=1}^N n_t^s$ appended after the trajectory pages, so a turn's absolute page index is $n_\tau + s_t^{start} + p$ for the p -th stub page of turn t .

H.3 Packed forward and FlashAttention-2 dispatch

CAPA reuses the per-layer KV that the PPO log-probability pass has already produced. Concretely, the standard log-prob pass calls $\pi_\theta(x, use_cache=True)$ once on the trajectory; we detach and clone the returned $\{(K^\ell, V^\ell)\}_{\ell=1}^L$ on the local device (the `detach().clone()` re-

Algorithm 3 SCATTERINTOSTUBPAGES: in-place fill of private stubs.

Require: $K_{stub}, V_{stub} \in \mathbb{R}^{n_s \times P \times h_{kv} \times d_{head}}$; flat $K_{traj}, V_{traj} \in \mathbb{R}^{L_\tau \times h_{kv} \times d_{head}}$; new $K_{new}, V_{new} \in \mathbb{R}^{(\sum_t s_t) \times h_{kv} \times d_{head}}$; turn metadata; $cu_seqLens$
1: **for** $t = 1$ **to** N **do**
2: $K_{tail} \leftarrow K_{traj}[ctx_end_t - \tau_t : ctx_end_t]$ (analog. V_{tail})
3: $K_{new,t} \leftarrow K_{new}[cu_seqLens[t] : cu_seqLens[t+1]]$ (analog. $V_{new,t}$)
4: $K_{seg} \leftarrow [K_{tail}; K_{new,t}]$ (length $\tau_t + s_t$)
5: **for** $p = 0$ **to** $n_t^s - 1$ **do**
6: $\ell_{lo} \leftarrow pP$; $\ell_{hi} \leftarrow \min((p+1)P, \tau_t + s_t)$
7: $K_{stub}[s_t^{start} + p, 0 : \ell_{hi} - \ell_{lo}] \leftarrow K_{seg}[\ell_{lo} : \ell_{hi}]$
8: (analog. for V_{stub}); trailing slots stay zero
9: **end for**
10: **end for**

moves the cached tensors from the autograd graph and retains independent storage for the subsequent packed call). No additional full forward is paid by CAPA itself.

The conditioned forward then proceeds in one pass (Algorithm 4). Each layer's cache is paged via Algorithm 1; Algorithm 2 runs once and the page table is shared across layers. The packed query is the concatenation $\mathbf{q}_{packed} = [c_1 r_1 c_2 r_2 \dots c_N r_N]$ and per-turn position IDs are $pos_t = (ctx_end_t, \dots, ctx_end_t + s_t - 1)$, so RoPE applied during the packed forward matches the absolute trajectory frame the trajectory KV was captured in. We register a custom attention backend `ALL_ATTENTION_FUNCTIONS["capa"]` and set the model's `_attn_implementation` to "capa" for the duration of the packed forward, restoring the previous setting in a `try/finally` block.

Inside the backend each layer receives (Q, K_{new}, V_{new}) of the new packed sequence plus that layer's $(K_{traj}^\ell, V_{traj}^\ell)$ pages via the `capa_payload` dictionary. The backend allocates $(K_{stub}^\ell, V_{stub}^\ell)$ and runs Algorithm 3. On CUDA fp16/bf16 it dispatches to `flash_attn_varlen_func` (Dao, 2024) with the page table:

```
flash_attn_varlen_func(
    q=q_packed, k=K_concat, v=V_concat,
    cu_seqLens_q=cu_seqLens,
    cu_seqLens_k=cu_k,
    max_seqLen_q=max_q,
    max_seqLen_k=max_k,
    causal=True, block_table=block_table)
```

where $K_{concat} = [K_{traj, pages}^\ell; K_{stub}^\ell]$ along the page axis, and $cu_k = [0, cumsum(cache_seqLens)]$. CPU/fp32 falls back to an eager reference (Algorithm 5) used in unit tests for indexing-equivalence

Algorithm 4 CAPA-FORWARD: end-to-end critique-conditioned forward.

Require: Trained model π_θ , trajectory x of length L_τ , prompt length L_p , annotated turns $\{(\text{resp_start}_t, \text{resp_end}_t, \mathbf{c}_t)\}_{t=1}^N$, page size P

- 1: **(Reuse)** retrieve $\{(K^\ell, V^\ell)\}_{\ell=1}^L$ from the standard PPO log-prob pass; `detach().clone()`
- 2: Build per-turn layout $\{(\text{ctx_end}_t, s_t, |\mathbf{c}_t|, |r_t|)\}_{t=1}^N$; build packed query $\mathbf{q}_{\text{packed}}$ and per-turn position IDs
- 3: (block_table, cache_seqlens, cu_seqlens, stubs) $\leftarrow$ BUILDPAGE TABLES($\cdot$)
- 4: **for** $\ell = 1, \dots, L$ **do**
- 5: $(K_{\text{pages}}^\ell, V_{\text{pages}}^\ell, \cdot) \leftarrow \text{KVToPAGES}(K^\ell, V^\ell, P)$
- 6: Stash flat K^ℓ, V^ℓ for stub-page tail copies
- 7: **end for**
- 8: Set π_θ 's attention implementation to `capa`; populate `capa_payload`
- 9: $\text{out}_{\text{cond}} \leftarrow \pi_\theta(\mathbf{q}_{\text{packed}}, \text{position_ids}, \text{capa_payload})$
- 10: Restore prior attention implementation
- 11: **for** $t = 1, \dots, N$ **do**
- 12: Locate response slice $[\text{resp_lo}_t, \text{resp_hi}_t)$ in $\text{out}_{\text{cond}}.\text{logits}$
- 13: $\text{cond_logp}_t \leftarrow \text{logsoftmax}(\text{out}_{\text{cond}}.\text{logits}[\text{resp_lo}_t - 1 : \text{resp_hi}_t - 1][\mathbf{q}_{\text{packed}}[\text{resp_lo}_t : \text{resp_hi}_t]])$
- 14: **end for**
- 15: **return** $\{\text{cond_logp}_t\}_{t=1}^N$

Algorithm 5 CAPA-EAGER: reference implementation (CPU/fp32 tests).

Require: Packed Q , paged $K_{\text{concat}}, V_{\text{concat}}$, block_table, cache_seqlens, cu_seqlens, turn layouts; softmax scale $\sigma = 1/\sqrt{d_{\text{head}}}$

- 1: **for** $t = 1$ **to** N **do**
- 2: $T_t \leftarrow \text{cache_seqlens}[t]$; gather first T_t keys/values by walking `block_table[t]` across pages; expand $h_{kv} \rightarrow n_{\text{heads}}$ for GQA
- 3: $S \leftarrow \sigma Q[\text{cu_seqlens}[t] : \text{cu_seqlens}[t+1]] \cdot K^\top$
- 4: Apply absolute-position causal mask: $S_{ij} \leftarrow -\infty$ if $j > \text{ctx_end}_t + i$
- 5: $\text{out}[t] \leftarrow \text{softmax}(S) \cdot V$
- 6: **end for**
- 7: **return** packed out

checking. Because positional encoding is baked into K_{traj}^ℓ via the original log-prob pass and into K_{new}^ℓ via the packed forward's RoPE, no further position bookkeeping is required inside the kernel: FA2's causal mask suffices since each query token's absolute trajectory position is strictly greater than the keys it is allowed to see.

H.4 FSDP integration

The actor is wrapped in PyTorch FullyShardedDataParallel (FSDP) for sharded parameters and gradients. CAPA integrates with FSDP at three points.

Module unwrap. The custom backend needs to inspect the underlying causal-LM

(`model.model.layers`, `model.lm_head`). FSDP and DeepSpeed both wrap the module; we unwrap with

```
root = getattr(model, "module",
    getattr(model,
        "_fsdp_wrapped_module",
        model))
```

inside `CapaCritiqueForward._inner_model`, then access `root.model.layers` for per-layer KV capture.

KV capture under sharding. The PPO log-prob pass runs under FSDP's normal all-gather/forward path, and the returned `past_key_values` contain complete-sequence activation tensors for the local batch in the layer's native dtype. We `detach().clone()` each (K^ℓ, V^ℓ) to remove it from the autograd graph and retain independent storage for the subsequent packed forward. The cached tensors stay on the same device; no cross-device transfer is required.

Packed forward. The packed forward triggers FSDP's all-gather identically to a normal forward; the only difference is that the input length is $\sum_t s_t$ (typically $\leq L_\tau$ in the regime where CAPA is profitable) and the attention backend is the registered `capa` function. We disable HF caching (`use_cache=False`) on the packed forward and pass `attention_mask={"full_attention": None, "sliding_attention": None}` so HF skips its default mask construction; CAPA builds its own mask via the page table plus FA2's `causal=True`. To keep gradient flow correct under FSDP we wrap the entire CAPA pass in `torch.no_grad()`: the log-probability shift $\delta_{b,t,j}$ enters Eq. (8) as a stop-gradient signal — it modulates the advantage but does not contribute its own gradient.

Mixed precision. On bf16/fp16 FSDP (the production setting), CAPA dispatches to FA2's varlen paged kernel with the layer's native dtype. The eager fallback is gated on `query.is_cuda & query.dtype in (fp16, bf16)`; unit tests run on CPU in fp32, exercising the same indexing logic through Algorithm 5.

H.5 Correctness, complexity, and fallbacks

Numerical correctness. We validate CAPA against the per-turn reference forward in two regimes. (i) On CPU/FP32 the maximum absolute difference across the response-token log-probabilities is below $\text{atol} = 10^{-5}$, limited by

softmax order of operations. (ii) On CUDA/BF16 with FA2’s paged kernel the maximum observed drift is below $\text{atol} = 4 \times 10^{-3}$, well below typical critique-induced shifts ($\text{std}(\delta) \approx 0.1$ in our runs). An end-to-end smoke test runs a single PPO mini-batch with CAPA enabled and disabled and confirms the resulting $A^{(3)}$ tensors agree within the same BF16 tolerance.

Complexity. Let L_τ be the trajectory length and $S = \sum_t s_t$ the total new-token count across annotated turns, and let $s_{\max} = \max_t s_t$. The naive baseline runs one additional forward for each annotated turn, with attention cost $\mathcal{O}(\sum_t (\text{ctx_end}_t + s_t)^2)$. CAPA pays no additional trajectory forward because it reuses the trajectory KV from the PPO log-prob pass, and its packed forward has attention cost $\mathcal{O}(\sum_t s_t (\text{ctx_end}_t + s_t)) = \mathcal{O}(SL_\tau + \sum_t s_t^2) = \mathcal{O}(S(L_\tau + s_{\max}))$. In our setting, $s_{\max} = \mathcal{O}(L_\tau)$, so this becomes $\mathcal{O}(SL_\tau)$. In our 7B / 32k-token regime this reduces the isolated Level 3 critique-conditioned forward from 178 s to 14 s, a $12.7\times$ speedup. This isolated forward accounts for approximately 5% of a training step and excludes teacher inference (Figure 7).

I Teacher Annotation Prompt

The box below shows the complete system prompt used to instruct the teacher LLM for trajectory annotation. The prompt is designed to elicit sparse, evidence-grounded labels: the teacher omits routine or neutral turns (no $d = 0$), commits to strong scores when evidence is clear, and grounds every judgment in observable environment feedback.

System Prompt for Teacher Annotation

You are an expert RL credit-assignment judge for long-horizon agent tasks. Output exactly one JSON object — no markdown, no code fences, no narration.

What you are doing

The trajectory has already finished. Your job is to identify turns where the agent’s decision **materially changed the outcome** (positively or negatively), and assign a graded score to each. Use full context for credit assignment: task goal, prior state, executed action, environment feedback, trajectory context, and final outcome. Use the final outcome as a consistency reference, but not as a hard rule — a successful trajectory can still contain bad turns and a failed one can still contain strong decisions. The downstream RL trainer uses these scores as token-level advantage modulators — the strength and polarity of your scores directly shape gradient updates.

Judgment criteria

- Ground every judgment in observable evidence: environment feedback, score changes, error messages, task state transitions, or logical inference from the trajectory

context, etc.

- **Positive turns:** advanced the goal, completed a sub-goal, set up necessary preconditions, recovered from an earlier mistake, gathered critical information, etc.
- **Negative turns:** hindered progress, triggered errors, repeated failed actions, moved away from the goal, wasted steps on irrelevant actions, violated environment protocol, etc.
- You may infer poor strategy even without an explicit env error — e.g. repeating an unsuccessful approach, ignoring relevant observations, choosing an obviously suboptimal path, etc.
- Skip routine/filler turns that had no meaningful impact. Only label turns that genuinely shifted the trajectory’s outcome.

Scoring guidance

- Commit to strong scores when the evidence is clear. ± 0.7 or ± 1.0 produce meaningful gradient signal; weak scores (± 0.3) are near-noise.
- Use ± 0.3 only when the step’s impact is genuinely ambiguous.
- For confidence: 1.0 when env feedback directly confirms the result; 0.75 when the judgment relies on inference; 0.5 when truly uncertain.
- There is no required balance between positive and negative labels. Label what the evidence supports — follow the evidence, not a quota.

Output schema (every field required for emitted turns)

- `trajectory_outcome_recap:` object with `agent_succeeded`.
- `turn_judgments:` list of labeled turns.
- Each labeled turn contains `turn_index`, `decision_quality`, `confidence`, `judgment_tags`, and `rationale`.
- `decision_quality:` one of -1.0, -0.7, -0.3, 0.3, 0.7, 1.0.
- `confidence:` one of 0.5, 0.75, 1.0.

Rules

- Do NOT emit `decision_quality = 0.0`; if a turn is neutral, omit it.
- Do NOT emit `confidence < 0.5`; below that threshold, omit the turn.
- Keep rationale concise — cite specific env feedback or state changes.
- One tag per turn. Allowed tags: `goal_progress`, `accuracy`, `action_choice`, `observation_use`, `planning`, `exploration`, `recovery`, `efficiency`, `tool_or_env_protocol`, `formatting`, `safety_or_constraint`, `other`.
- `agent_succeeded` must reflect the actual final outcome (lying voids the entire labeling).

Final reward formula (FYI)

$q = \text{decision_quality} * \text{confidence}$ per labeled turn, applied as a token-level advantage modulator. Strong verdicts ($|q| \geq 0.7$) move the policy; weak verdicts ($|q| \leq 0.225$) are near-noise.

J Teacher Annotation Case Study

This appendix illustrates the teacher-annotation pipeline in §3 with a concrete trajectory. We first state the sparse JSON schema returned by the DeepSeek-V3.1 teacher and distinguish it from the trainer-side fields derived from that response and the environment. We then walk through one ScienceWorld trajectory turn by turn, showing how the

teacher’s per-turn rationale and signed score q_t provide *decision-level* information that trajectory-level reward cannot.

J.1 Annotation schema

For a trajectory of N turns, the teacher returns one JSON object whose `turn_judgments` list contains only materially consequential turns. Routine, neutral, or below-threshold turns are omitted:

```
{
  "trajectory_outcome_recap": {
    "agent_succeeded": bool,
    "turn_judgments": [{
      "turn_index": int,
      "decision_quality":
        -1.0 | -0.7 | -0.3 |
        0.3 | 0.7 | 1.0,
      "confidence": 0.5 | 0.75 | 1.0,
      "judgment_tags": ["<one allowed
tag>"],
      "rationale": string
    }]
}
```

After schema validation, the trainer converts each emitted judgment into an internal turn record and computes $q_t = \text{decision_quality} \times \text{confidence}$. Level 2 uses the signed value of q_t , whereas Level 3 uses $|q_t|$ together with the rationale-conditioned log-probability shift. For readability, Table 8 reports $\text{sgn}(q_t)$ as a positive/negative label. Omitted turns receive $q_t = 0$ and bypass fine-grained modulation. The trainer derives `response_start/response_end` from the tokenized trajectory, rather than from the teacher, and obtains `has_format_error` from the environment. When that flag is true, we override $q_t \leftarrow 0$ and bypass L2/L3 so teacher leniency cannot weaken the environment penalty (§3.4). Finally, the trainer copies the teacher’s rationale into `critique_text` and wraps it with explicit `<teacher_note>...</teacher_note>` delimiters. This text becomes $r_{b,t}$ in Eq. (7) and can be spliced after the turn prefix by the critique-conditioned forward (Appendix H.3).

J.2 Worked example: “Find a non-living thing and move it to the blue box.”

Table 8 displays a complete 7-turn trajectory from the ScienceWorld training pool (training step 43). The student makes the right high-level moves (turns 0–2), gets stuck for three consecutive turns on a parser ambiguity (turns 3–5), and recovers in turn 6. We chose this trajectory as illustrative because (i) the goal — “find a non-living thing, then move

it to the blue box in the kitchen” — is short and unambiguous; (ii) the teacher emits both positive and negative turn-level judgments with rationales that diagnose the parser failure; and (iii) the final trajectory reward (+92 raw progress, `won=1`) summarizes the episode but does not identify the mid-episode failure that L2/L3 need to penalize.

The annotations in Table 8 provide decision-level information that is absent from GRPO’s broadcast trajectory advantage. Turns 3–5 each receive $A_{\text{GRPO}} > 0$ because the trajectory ultimately wins, but the teacher’s $q_t \in \{-0.7, -1.0, -0.7\}$ sets $w_{\text{crit}} < 1$ and downweights these actions, while turn 6’s $q_t = +1.0$ amplifies the recovery action. At the token level, conditioning on the turn-3 rationale — “Failed to specify inventory cup in move command, triggering ambiguity error” — changes the log probabilities of the realized response move cup to blue box. The resulting $\delta_{b,t,j}$ reweights tokens in this observed action; it does not assign credit to ungenerated counterfactual tokens.

J.3 Why this layout works for L1–L3

Three structural properties of the schema are exploited by the algorithm. (i) *Sign agreement*. $\text{sgn}(q_t)$ disagrees with $\text{sgn}(A_{\text{GRPO}})$ on turns 3–5, so L2’s w_{crit} shrinks below 1 (Eq. 5); on turns 0–2 and 6 it agrees, so $w_{\text{crit}} > 1$. The trajectory’s total advantage is preserved up to a covariance term (Eq. 11), but the credit is reallocated toward the productive turns. (ii) *Score strength $|q_t|$* . Disapproval is strongest at turn 4 (−1.0) after the repeated error, then returns to −0.7 at turn 5. L3 retains these graded magnitudes in the token-level scaling rather than collapsing them to a single bit. (iii) *Targeted rationale*. The free-form text on turn 3 names the precise failure (“failed to specify inventory cup”) instead of restating the outcome. When this rationale is spliced into the critique-conditioned forward (Appendix H.3), the resulting per-token δ is computed over the realized response and allows L3 to reweight the tokens that produced the underspecified action.

Table 8: One ScienceWorld trajectory annotated by the DeepSeek-V3.1 teacher (training step 43, task: *find a non-living thing and move it to the blue box in the kitchen*). Score is the environment’s running progress in $[0, 100]$; Δ is the score change for that turn. The trajectory completes successfully ($won=1$, final score 100), but turns 3–5 stall on a parser ambiguity that trajectory-level reward cannot localize. Teacher rationales are reproduced verbatim with the `<teacher_note>` wrappers stripped.

Turn	Score	Δ	Action	q_t	Label	Teacher rationale
0	8 $\rightarrow$ 58	+50	focus on cup	+1.0	positive	Correctly identified the non-living thing (cup) to focus on, confirmed by +50 score increase from environment.
1	58 $\rightarrow$ 67	+9	pick up cup	+1.0	positive	Successfully picked up the cup, moving it to inventory as required for transportation, confirmed by +9 score increase.
2	67 $\rightarrow$ 83	+16	go kitchen	+1.0	positive	Moved to the kitchen where the blue box is located, essential for task completion, confirmed by +16 score increase.
3	83 $\rightarrow$ 83	0	move cup to blue box	−0.7	negative	Failed to specify inventory cup in move command, triggering ambiguity error and wasting a turn.
4	83 $\rightarrow$ 83	0	move cup to blue box	−1.0	negative	Repeated same ambiguous command format after explicit environment feedback, resulting in unknown action error.
5	83 $\rightarrow$ 83	0	move cup to blue box in kitchen	−0.7	negative	Added redundant location specification instead of addressing the inventory ambiguity, repeating the same error.
6	83 $\rightarrow$ 100	+17	$\emptyset$	+1.0	positive	Correctly resolved the ambiguity by selecting option 0 (inventory cup), successfully completing the task with +17 score increase.