HARNESS-AWARE DISTILLATION FOR SMALL LANGUAGE MODEL AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Language model agents are deployed with a harness, the software around the model that manages its context, tools, and feedback. When such an agent is distilled into a smaller one, the harness stays in place, so the student mainly needs the teacher-specific abilities that the harness cannot provide, such as acting correctly on harness information. Standard distillation, however, imitates the teacher's full outputs and treats the harness as part of the input. We propose Harness-Aware Distillation (HAD), which focuses distillation on what the teacher adds beyond the harness. HAD complements on-policy distillation with two components: an action preference that contrasts the same teacher's actions with and without the harness information, scored after the student's own reasoning, and a validity check that drops preference pairs whose preferred action contradicts the harness records. We show that the contrast gives the student information that imitating the teacher alone cannot provide, and HAD needs no task rewards, success labels, or future information. Across multiple long-horizon agent benchmarks and models, HAD outperforms on-policy distillation baselines with the same harness, and on one benchmark its student even surpasses the larger teacher. Our analysis shows that HAD enters fewer unproductive loops and recovers from errors more often than the baselines, and suggests that it adaptively keeps learnable feedback in its weights while reading state information from the harness.

1 INTRODUCTION

Language model agents solve long-horizon tasks through repeated interactions with an environment and its tools (Yao et al., 2023; Yang et al., 2024a). They are usually deployed inside a *harness*, the software around the model that manages its context, tools, and feedback (Wang et al., 2025; Lee et al., 2026). A harness can, for example, track the environment state, list the available actions, and report when the agent repeats an action without effect. Changing only the harness, with the model fixed, can improve agent performance substantially (Yang et al., 2024a; Lee et al., 2026), so practical deployments assume a harness-equipped agent.

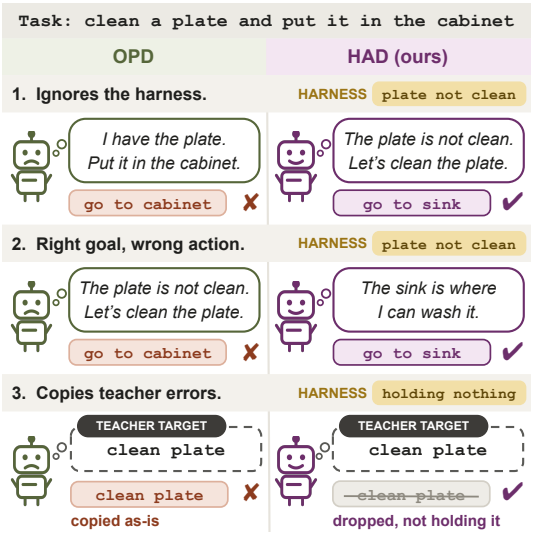


Figure 1: OPD versus HAD.

At the same time, the high cost of large models gives a strong incentive to distill their capabilities into smaller agents (Belcak et al., 2025; Zeng et al., 2024). On-policy distillation (OPD) is a common choice, as it provides teacher supervision at the states the student itself visits (Agarwal et al., 2024; Lu & Thinking Machines Lab, 2025; Wang et al., 2026; Zhou et al., 2026), but existing methods do not explicitly account for the harness. Distilling a harness-equipped agent is a problem between two systems that share the same harness. The harness stays fixed when the teacher is replaced by a smaller model, so the student does not need to learn what the harness

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

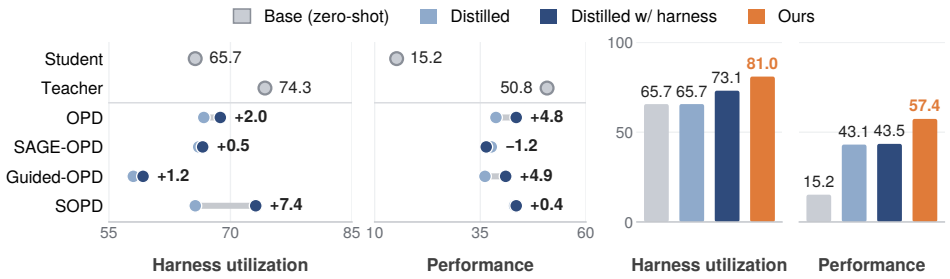


Figure 2: Harness utilization and performance of distilled students on ALFWorld. **Left:** Each OPD method distilled without (light) and with (dark) the harness. **Right:** The zero-shot student, a distilled student trained without and with the harness, and HAD. Adding the harness to distillation improves harness utilization (65.7%→73.1%) but not performance (43.1%→43.5%), while HAD improves both (65.7%→81.0%; 43.1%→57.4%).

already provides. What it needs is the teacher's own ability to act on what the harness reports. A natural approach is to run OPD with the same harness for the teacher and the student, yet this approach does not reliably transfer the ability. As Figure 2 shows, adding the harness to OPD changes the student's performance inconsistently across methods. The agreement between the student's actions and the harness reports, which we call harness utilization, also improves only slightly, and for one method it stays below that of the untrained student.

We attribute this to three failures in the supervision that OPD provides, illustrated in Figure 1. First, *OPD leaves the teacher's use of the harness implicit*. Table 2 shows that part of the teacher's performance comes from the harness, and keeping this benefit requires the student to learn when harness information should change its decision. OPD supervises every step in the same way, so it does not mark the few steps where the harness changes the teacher's action, and it never shows what the teacher would choose without the harness. Second, *OPD does not separate what the harness provides from what the teacher adds*. The harness supplies information and feedback, but deciding how to act on them is an ability of the teacher that the harness cannot provide. In the middle row of Figure 1, the student reads that the plate is not clean, yet goes to the cabinet instead of the sink. Third, *OPD accepts teacher targets that contradict the harness records*. Teachers can be less reliable at student-visited states (Wang et al., 2026; Li et al., 2026; Zhou et al., 2026), and the harness can reveal such errors, for example a target that requires an object the agent does not hold, but OPD does not check teacher targets against these records.

Our key insight is that the shared harness can itself guide distillation, as it shows which teacher decisions depend on it and its records reveal teacher targets that contradict observed facts. We propose **Harness-Aware Distillation (HAD)**, which adds two components to response distillation, our term for on-policy distillation of the teacher's full response with the harness enabled, as illustrated in Figure 3. *Harness-aware action preference learning* queries the same teacher with and without the harness information and trains the student to prefer the action chosen with the harness, scored after the student's own reasoning. This contrast is active only at states where the harness changes the teacher's action, and as we show in Section 3.2, it carries information that imitating the teacher's response alone does not provide. *Harness-aware action validity filtering* drops preference pairs whose preferred action contradicts the harness records, while response distillation is unchanged. HAD uses no task rewards, success labels, or future information.

Our contributions are as follows:

- We view the distillation of a harness-equipped agent as distillation between two systems that share the harness, so the student should learn the teacher-specific abilities that the harness cannot provide. We show that on-policy distillation with the shared harness does not reliably teach the student what the teacher adds beyond the harness.
- We propose HAD, which queries the same teacher with and without the harness and teaches the student to prefer the action chosen with the harness. Since the teacher can still make mistakes, a validity check drops pairs whose preferred action contradicts the harness records. We show in Section 3.2 that the contrast adds information that imitating the teacher alone cannot provide.

- HAD outperforms on-policy distillation baselines with the same harness across multiple long-horizon agent benchmarks and models. Our analysis in Section 4.3 shows that HAD learns to recover from stalls, unlike the baselines, and suggests that it keeps learnable feedback in its weights while reading state information from the harness.

2 Related Work

Distilling language model agents. Language model agents combine model predictions with external actions and feedback (Yao et al., 2023; Shinn et al., 2023). Building on knowledge distillation (Hinton et al., 2015; Kim & Rush, 2016), agent distillation transfers reasoning and tool-use behavior to smaller models (Chen et al., 2023; Kang et al., 2025). Structured supervision further distinguishes between reasoning and action spans (Liu et al., 2026). To reduce the mismatch between teacher-generated training data and student behavior, on-policy distillation supervises student-generated sequences using teacher feedback (Agarwal et al., 2024; Gu et al., 2024). In multi-turn settings, this paradigm has been extended with step-level supervision (Sun et al., 2026), temporal curricula (Wang et al., 2026), and selective teacher intervention (Zhou et al., 2026).

Learning with external context and agent harnesses. External context and agent harnesses can support inference and provide supervision during training. Context distillation and harness self-distillation aim to internalize contextual knowledge or harness-augmented reasoning into model weights (Ye et al., 2026; Zhao et al., 2026). Other work studies the interplay between harness design and post-training, including training compact agents to operate across varying harness configurations (Kim et al., 2026). External guidance can also shape the supervision signal itself: for example, rollout returns can determine the direction of distillation between skill-conditioned and unconditioned views (Tu et al., 2026). Our focus is instead on explicitly supervising how smaller agents use information provided by the harness, while retaining the harness at deployment time.

Contrastive supervision and preference-based distillation. Preference learning trains models to favor one candidate over another, using either a reference-based objective or a reference-free objective combined with imitation (Rafailov et al., 2023; Xu et al., 2024). These approaches also differ in how they construct preference pairs. Prior work constructs synthetic pairs through contrasting prompts (Yang et al., 2024b), compares answers generated with and without retrieval using answer correctness (Yan et al., 2025), or forms preferences between teacher and student generations (Li et al., 2024; Yu et al., 2026). Our method constructs action preferences from the same teacher with and without harness information at student-visited states. We treat the harness-conditioned action as preferred and exclude pairs when that action contradicts harness records. This supervision requires neither task rewards nor success labels.

3 Method

3.1 Harness-Aware Distillation

We consider a multi-turn agent interacting with an environment over a horizon T. For a task $u \sim \mathcal{D}$, an external harness augments each environment observation o_t with supplementary information h_t, computed deterministically from the environment state and interaction history. HAD makes no assumption about what h_t contains, and only requires that it is computed deterministically from the environment state and the interaction history. Writing $\tilde{o}_t = (o_t, h_t)$, we define the *harness-induced history* $\tilde{x}_t$ and its *stripped history* x_t as

$$\tilde{x}_t = (u, \tilde{o}_0, \tilde{y}_0, \ldots, \tilde{y}_{t-1}, \tilde{o}_t), \qquad x_t = (u, o_0, \tilde{y}_0, \ldots, \tilde{y}_{t-1}, o_t). \tag{1}$$

The agent conditions on $\tilde{x}_t$ to generate a response $\tilde{y}_t = (\tilde{z}_t, \tilde{a}_t)$ consisting of a reasoning trace and an action. The environment executes $\tilde{a}_t$ and returns the next observation o_{t+1}. Stripping removes only $h_0, \ldots, h_t$, not their influence on the realized observations and responses; x_t is therefore not a separately generated harness-free trajectory.

Harness-equipped response distillation. Let π_ϕ be a teacher and π_θ a student. With the harness enabled for both the student and the teacher, we collect student rollouts matching deployment and

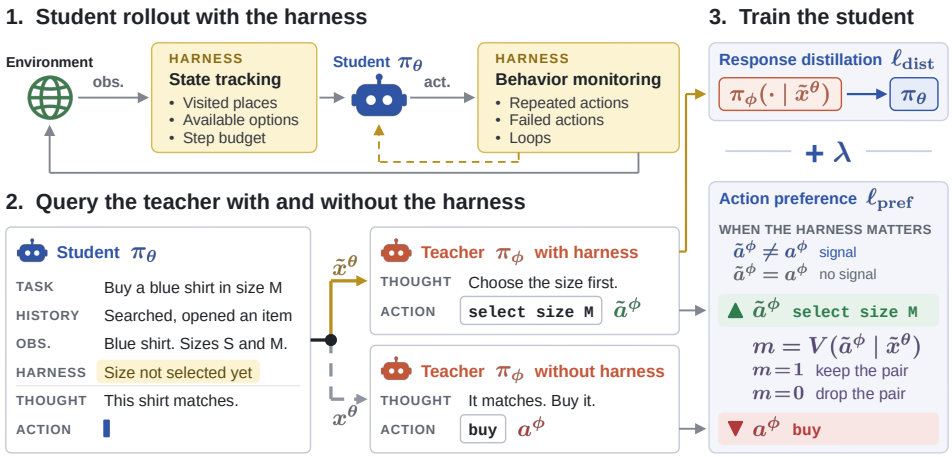


Figure 3: Overview of HAD. (1) The student acts through the harness, which tracks the environment state and monitors the student's behavior. (2) At each visited state, the same teacher is queried with and without the harness information. (3) The student imitates the teacher's response given the harness and learns to prefer the action chosen with the harness over the action chosen without it, which gives a signal only when the two actions differ. The validity mask drops pairs whose preferred action contradicts the harness records.

query the teacher at each visited history.

$$\tilde{y}_t^\theta = (\tilde{z}_t^\theta, \tilde{a}_t^\theta) \sim \pi_\theta(\cdot \mid \tilde{x}_t^\theta), \qquad \tilde{y}_t^\phi = (\tilde{z}_t^\phi, \tilde{a}_t^\phi) \sim \pi_\phi(\cdot \mid \tilde{x}_t^\theta). \tag{2}$$

We distill the teacher's complete response using its mean token negative log-likelihood:

$$\ell_{\text{dist},t}(\theta) = -\frac{1}{|\tilde{y}_t^\phi|} \sum_{j=1}^{|\tilde{y}_t^\phi|} \log \pi_\theta\left(\tilde{y}_{t,j}^\phi \mid \tilde{x}_t^\theta, \tilde{y}_{t,<j}^\phi\right). \tag{3}$$

Harness-aware action preference learning. Distilling at student-visited states addresses the distribution shift of training on teacher trajectories, but does not explicitly supervise harness-induced behavioral differences. We therefore query the same teacher on the stripped history:

$$y_t^\phi = (z_t^\phi, a_t^\phi) \sim \pi_\phi(\cdot \mid x_t^\theta). \tag{4}$$

Given the visited history, the two teacher responses, with and without the harness, are independent of each other and of the student's current reasoning. Using access to the explicit harness information as a quality proxy, we prefer $\tilde{a}_t^\phi$ over a_t^ϕ and score both under the same student reasoning $\tilde{z}_t^\theta$:

$$\Delta_t(\theta) = \frac{1}{|\tilde{a}_t^\phi|} \log \pi_\theta\left(\tilde{a}_t^\phi \mid \tilde{x}_t^\theta, \tilde{z}_t^\theta\right) - \frac{1}{|a_t^\phi|} \log \pi_\theta\left(a_t^\phi \mid \tilde{x}_t^\theta, \tilde{z}_t^\theta\right). \tag{5}$$

Each score averages log-likelihood over action tokens only, excluding the reasoning prefix. We minimize the *reference-free* pairwise logistic loss (see Appendix A.4 for its regularization bias):

$$\ell_{\text{pref},t}(\theta) = -\log \sigma(\beta \Delta_t(\theta)), \tag{6}$$

where σ is the sigmoid and $\beta > 0$ controls the margin scale. Unlike response distillation, this term supervises actions under student rather than teacher reasoning.

Harness-aware action validity filtering. We set $m_t = V(\tilde{a}_t^\phi \mid \tilde{x}_t^\theta)$, where the validity checker $V(a \mid \tilde{x}) \in \{0, 1\}$ evaluates actions using the harness-induced history. Specifically, we judge the action invalid if it conflicts with the harness-induced environment observation. The mask applies only to the harness-aware preference learning term, leaving the harness-equipped distillation term unchanged. See Appendix A.1 for details on how V is implemented in practical settings.

Final objective. Let ν denote the distribution of collected training records. Our proposed Harness-Aware Distillation (HAD) combines response distillation with validity-filtered action preferences:

$$\mathcal{L}_{\mathrm{HAD}}(\theta;\lambda) = \mathbb{E}_{\nu}[\ell_{\mathrm{dist},t}(\theta) + \lambda m_t \ell_{\mathrm{pref},t}(\theta)]\,. \tag{7}$$

We adjust λ at each update using minibatch gradient-norm balancing of the two loss terms (Appendix A.1). The ablations in Section 4.5 show gains over harness-equipped distillation alone ($\lambda = 0$), unfiltered preferences ($m_t \equiv 1$), full-response preferences, positive-only supervision, and a teacher reference model. Full-response preferences compare $\tilde{y}_t^{\phi}$ with y_t^{ϕ} under the same augmented history, rather than comparing actions under student reasoning. Positive-only supervision imitates $\tilde{a}_t^{\phi}$ under the student reasoning prefix without contrasting it with a_t^{ϕ}. See Appendix A.2 for more detailed definitions; corresponding ablation results are provided in Section 4.5 and Appendix D.3.

3.2 What Harness Awareness Adds to Distillation

Our proposed harness-aware preference learning makes *harness awareness* explicit by contrasting the same teacher's actions with and without explicit harness information. For a fixed harness-induced history $\tilde{x}^{\theta}$ and its stripped history x^{θ}, let $P_+(a) = \sum_z \pi_{\phi}(z, a \mid \tilde{x}^{\theta})$ and $P_-(a) = \sum_z \pi_{\phi}(z, a \mid x^{\theta})$ be the teacher's action marginals. Here, $+$ denotes the teacher with explicit harness information, while $-$ denotes the same teacher with that information stripped from the history. Assuming that harness information generally helps the teacher choose better actions, we form each preference pair by labeling the harness-equipped action from P_+ positive and the stripped-history action from P_- negative. The analysis below characterizes the auxiliary preference term; formal assumptions and derivations appear in Appendix A.3 and Appendix A.5.

What supervision is added? Response distillation already supervises harness-equipped behavior, but it does not distinguish actions that are generally likely under the teacher from those that become more likely specifically because of the harness. Our additional supervision reveals how the teacher's action probabilities differ without the harness information. For independent draws from strictly positive laws on a common finite action set, the isolated unfiltered preference loss $\ell_{\mathrm{pref,t}}$ has optimal unrestricted scores $\log(P_+(a)/P_-(a)) + \kappa$, with arbitrary constant κ as in Proposition A.1. Thus, our harness-aware preference term does not simply reinforce actions with high $P_+(a)$; it emphasizes actions with high $P_+(a)$ relative to $P_-(a)$. Response distillation alone cannot identify this relative change, since its risk depends only on the positive response law as mentioned in Observation A.1.

Why filter positive preference labels? Harness access does not guarantee that the positive action $a \sim P_+$ is valid. When it is impossible to know which action is correct, the best we can do is exclude actions that are clearly invalid because they conflict with the harness-induced environment observation. We thus use the validity checker V, with $V(a \mid \tilde{x}^{\theta}) = 1$ denoting acceptance. If the positive action is rejected, we drop only its preference loss, leaving response distillation unchanged. With independent teacher sampling and nonzero acceptance, the masked risk is the acceptance probability times the preference risk with P_+ conditioned on acceptance and P_- unchanged (cf. Observation A.2). However, filtering cannot remove cases where the positive action is valid yet incorrect; this limitation is inherent to any distillation framework that follows the incomplete teacher.

3.3 Why Action-Only Preference under Student Reasoning

Our preference term is designed with two considerations: what to compare and under which reasoning context to compare it. We motivate these choices in the following paragraphs, with further formal analysis in Appendices A.6 and A.7.

Why action-only? Our goal is to supervise the action difference induced by the harness, rather than differences between complete teacher responses. When the teacher is queried with and without the harness, its reasoning can differ substantially in content or style even when the resulting action is the same. Comparing complete responses can therefore spend much of the preference signal on distinguishing such reasoning differences, rather than focusing on the action change induced by the harness. This effect is further amplified because reasoning traces are typically much longer than actions, allowing them to dominate the full-response comparison and weaken the action-level signal. Indeed, our empirical ablation in Table 15 shows that the full-response variant performs worse. Appendix A.6 further formalizes this distinction: full-response preferences combine the desired action

contrast with an additional teacher-reasoning contrast, which can alter the induced action ordering even when the teacher action marginals are unchanged.

Why student-generated reasoning? Having isolated the action-level comparison, we score the teacher actions with and without the harness under the student's own sampled reasoning. This matches the deployed student's decision context: at inference time, it chooses an action after its own reasoning, not after the reasoning underlying either teacher action. Accordingly, the preference term does not condition action supervision on the teacher's no-harness reasoning; the harness-equipped teacher's reasoning is instead learned separately through response distillation. Thus, the auxiliary preference term focuses on action choices under the student's own reasoning, while response distillation provides supervision for the desired teacher reasoning. Appendix A.7 further formalizes that student-prefix conditioning preserves P_+ and P_- at student-visited reasoning contexts, while response distillation complements the auxiliary objective with reasoning-dependent supervision.

4 Experiment

4.1 Experimental setup

Models and datasets. We evaluate on ALFWorld (Shridhar et al., 2021), WebShop (Yao et al., 2022), and ScienceWorld (Wang et al., 2022). Using models from the Qwen3 family (Yang et al., 2025), we distill 1.7B from the 8B teacher on ALFWorld, and 0.6B from the 30B-A3B teacher on WebShop and ScienceWorld. To assess generalization across model families, we additionally distill Gemma-4-E2B from Gemma-4-12B on ALFWorld (Gemma Team et al., 2026). All evaluations use temperature 0.4 with thinking disabled; Appendix B.1 gives further details.

Harness. We set the harness to be task-agnostic, sharing the same code and templates across environments without task-specific knowledge such as subgoals, and it only adds text to the observation without selecting or blocking actions. Its components follow functions common in agent harnesses (Wang et al., 2025) and fall into state tracking and behavior monitoring, as listed in Table 1. State tracking is shown at every step, and behavior monitoring only when its condition fires. Every method uses the same harness for training and evaluation, and Appendix B.2 gives the templates, thresholds, validity check, and example pages.

Table 1: Harness components.

Component	Computed from
State Tracking	
Available actions	Environment action list
Situation	Location and inventory
History	Visit history
Budget	Step counter
Behavior Monitoring	
Failure	Actions at the same state
Stall	Recent states

Metrics. We report the success rate (SR) on all environments, the percentage of episodes that complete the task within the step limit, and the score on WebShop and ScienceWorld, given by the mean final reward scaled to 100 and the mean final task score, respectively. We also report the average number of turns in successful episodes, and harness utilization (HU), which measures how often the agent's next action agrees with what the harness reports. For each harness component, we compute the fraction of steps where the next action is consistent with its report and average these fractions over the components. For example, the next action should be among the available actions and should differ from an action reported as failed. We use this metric only as a diagnostic of how the agent responds to the harness, and Appendix B.3 gives the full rules and counting procedure.

Baselines. We compare HAD with four OPD methods, all trained and evaluated with the same harness. OPD (Agarwal et al., 2024) matches the teacher with the token-level reverse KL divergence on the student's own responses, SAGE-OPD (Zhou et al., 2026) reweights each turn by how strongly the teacher judges that it needs correction, Guided-OPD (Li et al., 2026) lets the teacher take over a decreasing fraction of turns, and SOPD (Sun et al., 2026) imitates a full step written by the teacher at each state the student visits. All methods share the same training loop, task pool, and number of updates, and we also report the zero-shot teacher and student. Appendix B.4 gives implementation details and differences from the original papers.

Table 2: Main results. Each cell reports performance **with harness** / without harness. All distillation methods are trained with the harness, and the teacher and the student are evaluated zero-shot. Bold marks the best student in each column.

	ALFWorld (1.7B ← 8B)		WebShop (0.6B ← 30B)		ScienceWorld (0.6B ← 30B)	
Method	Unseen SR	Seen SR	Score	SR	Score	SR
Teacher	51.49 / 38.06	50.00 / 37.86	49.96 / 41.41	18.80 / 13.40	20.26 / 18.47	20.13 / 15.44
Student	9.70 / 2.99	20.71 / 6.43	20.95 / 4.29	9.80 / 1.40	−16.47 / −16.97	3.36 / 3.36
OPD	47.01 / 31.34	40.00 / 31.43	41.59 / 39.05	19.00 / 16.60	−0.82 / −0.64	8.05 / 7.38
SAGE-OPD	31.34 / 25.37	41.43 / 34.29	42.21 / 39.15	17.00 / 14.00	0.77 / 1.41	6.04 / 6.71
Guided-OPD	46.27 / 33.58	35.71 / 27.14	43.89 / 40.09	16.60 / 14.40	−2.52 / −4.17	6.04 / 6.04
SOPD	45.52 / 26.87	41.43 / 27.86	39.97 / 37.64	13.80 / 13.60	−8.45 / −8.00	6.71 / 5.37
HAD (ours)	**63.43** / 40.30	**51.43** / 37.86	**47.19** / 41.63	**19.60** / 15.00	**4.17** / 4.37	**9.40** / 8.72



Figure 4: An unseen ALFWorld task where heating works only on the object in hand. The baselines loop on actions with no effect until the step limit, while HAD leaves its loop after the stall report and heats the cup once the harness reports that it holds it.

Hyperparameter setting. All methods use full fine-tuning with AdamW, a cosine learning rate schedule with 15 warmup updates, gradient clipping at 1.0, weight decay 0.01, and an effective batch size of 60. The learning rate is 10^{-5} for all methods. Each round rolls out 16 tasks with the current student, and each collected step is used once within two policy versions. We train for 250 updates on ALFWorld and 150 updates on WebShop and ScienceWorld. For HAD, we use $\beta = 0.5$ and set λ at each update so that the gradient norm of the preference term is $\rho = 0.5$ times that of response distillation. Appendix A.1 and Appendix B.4 list all the details.

4.2 Main Experiment

Quantitative analysis. HAD achieves the best performance in every environment in Table 2. With the harness, it reaches 63.4% on unseen and 51.4% on seen ALFWorld tasks, against 47.0% and 41.4% for the best baseline, and it even exceeds its 8B teacher. The harness also matters more for HAD than for the baselines, consistent with our design that trains the student on the decisions the harness changes. Removing the harness at evaluation lowers HAD from 63.4% to 40.3% on unseen tasks, the largest drop among all methods, yet HAD still outperforms the majority of baselines. Appendices B.3 and D.1 further show that its actions are the most consistent with the harness and that it takes a similar number of turns in successful episodes as the other methods.

Qualitative analysis. Figure 4 shows an unseen ALFWorld task that asks the agent to heat a cup and put it in a cabinet, where heating works only on the object the agent holds. Every student puts the cup in the microwave and then tries to heat it with an empty hand, and the baselines keep repeating actions with no effect until the step limit. HAD falls into a similar loop but leaves it with the help of

Table 3: Prevention (episodes without stalls or repeated failed actions) and recovery (stalls escaped) rates (%) on ALFWorld.

	Prevention	Recovery
Teacher	64.6	47.0
Student	20.4	46.8
OPD	64.2	42.9
SAGE-OPD	60.2	32.4
Guided-OPD	60.6	18.8
SOPD	62.0	50.0
HAD (ours)	**75.9**	**59.7**

Table 4: Success rate of HAD on ALFWorld with one harness category hidden or harness-rejected actions blocked.

Harness at evaluation	Unseen	Seen
Full harness	63.43	51.43
Hidden		
Behavior monitoring	55.97	45.00
State tracking	37.31	40.71
Blocked		
Rejected actions	55.22	46.43

the harness. After the harness flags that five steps produced nothing new, HAD takes the cup back, and once the harness shows that it holds the cup, HAD heats it and completes the task. Appendix C gives additional examples.

4.3 Analysis

Error trajectory recovery. Long-horizon agents often fall into loops of actions that have no effect, and escaping such error states is a known difficulty (Wang et al., 2025; Zhou et al., 2026). We ask whether HAD avoids these states, which we call prevention, and whether it leaves a stall once stuck, which we call recovery, counting a stall as escaped when the agent reaches a new state. *HAD improves both prevention and recovery over every other model*, as shown in Table 3. It avoids stalls and repeated failed actions in 75.9% of its episodes, against at most 64.6% for the other models. Standard distillation does not teach recovery, as no baseline clearly improves over the untrained student at 46.8%, whereas HAD escapes 59.7% of its stalls. Every stall we count is flagged by the harness, which suggests that HAD learns to act on the harness to leave a stall, and the states where the two teacher actions differ are spread over the whole episode, as shown in Appendix D.4.

Harness internalization. A harness is internalized when the model learns to produce, from its own weights, the behavior that the harness would otherwise supply, so that it acts correctly even when that part of the harness is removed (Zhao et al., 2026). HAD is not designed to internalize the harness, but our results suggest that it may internalize part of the harness while still relying on the rest. Without the harness, HAD reaches 40.3% on unseen and 37.9% on seen tasks in Table 2, above every baseline without the harness and on par with or above the teacher without the harness. Table 4 shows that this reliance differs across harness categories. Hiding state tracking lowers the success rate of HAD from 63.4% to 37.3% on unseen tasks and from 51.4% to 40.7% on seen tasks, whereas hiding behavior monitoring lowers it to 56.0% and 45.0%, although behavior monitoring also appears on fewer steps. Blocking the actions that the harness rejects also lowers the success rate on both splits, which suggests that the harness is more useful to HAD as information to act on than as a hard constraint. These results are consistent with HAD internalizing part of the feedback from behavior monitoring, while it still reads state information, such as the places visited and the objects held, from the harness. State tracking thus appears more essential at deployment.

4.4 Cross-model generalization

Table 5 repeats the ALFWorld evaluation with the Gemma-4 family; E2B students are distilled from the 12B teacher, under the same recipe and harness. HAD again achieves the highest success rate among the students, with 37.3% on unseen and 36.4% on seen tasks against 32.1% and 34.3% for the strongest baseline. Its harness utilization also exceeds even that of the teacher, as detailed in Appendix D.2. This agreement across model families suggests that the gain of HAD comes from the harness contrast rather than from properties of a specific model family.

Table 5: Gemma-4 results.

Method	Unseen SR	Seen SR
Teacher	51.49 / 43.28	49.29 / 42.14
Student	14.93 / 8.21	18.57 / 15.71
OPD	26.87 / 18.66	31.43 / 29.29
SAGE-OPD	24.63 / 20.90	28.57 / 27.14
Guided-OPD	29.10 / 18.66	32.86 / 24.29
SOPD	32.09 / 29.85	34.29 / 25.71
HAD (ours)	**37.31** / 23.88	**36.43** / 30.00

4.5 Ablation study

Component ablation. Table 6 removes each component of HAD in turn on ALFWorld. Removing the validity check ($m_t \equiv 1$) lowers the SR from 57.4% to 50.8%, since teacher actions that contradict the harness records then enter the preference loss. Removing the negative, so that the student only imitates the action chosen with the harness after its own reasoning, lowers the SR to 52.4% and harness utilization from 81.0% to 74.4%, close to the 73.1% of response distillation alone. The contrast with the action chosen without the harness therefore accounts for most of the gain in harness utilization (HU), while the validity check mainly protects task performance. Removing the preference term leaves response distillation alone ($\lambda = 0$), and the SR drops to 43.5%. Each component thus contributes to HAD in a complementary manner.

Table 6: Component ablations.

Variant	SR	HU
HAD (ours)	**57.4**	**81.0**
w/o validity ($m_t \equiv 1$)	50.8	78.9
w/o negative (pos-only)	52.4	74.4
w/o preference ($\lambda = 0$)	43.5	73.1

Teacher budget. In the main results, every method is trained for the same number of updates, but HAD queries the teacher twice per state, once with and once without the harness, while the baselines query it once. To separate the effect of this extra teacher budget, Table 7 also trains OPD and SOPD with twice the teacher budget. Each baseline sees twice as many student trajectories, so that the teacher is queried as often as in HAD, which also gives the baselines twice as many updates and favors them in this comparison. Even with this advantage, the success rate averaged over seen and unseen tasks only rises from 43.5% to 46.7% for OPD and from 43.5% to 46.8% for SOPD, far below the 57.4% of HAD. The gain of HAD is therefore not explained by the additional teacher queries.

Table 7: Teacher budget ablations.

Method	Unseen SR	Seen SR
OPD	47.0	40.0
OPD, 2×	47.8	45.7
SOPD	45.5	41.4
SOPD, 2×	50.8	42.9
HAD (ours)	**63.4**	**51.4**

Design choices. HAD scores only the action after the student's own reasoning and uses no reference model, following Section 3.3. As shown in Table 15, both alternatives to these choices are less effective than HAD (see Appendix D.3). Comparing full responses instead of actions lets the loss drop by telling apart the teacher's reasoning styles with and without the harness, and the long reasoning dilutes the signal from the short action. Scoring only the action after the student's own reasoning avoids both problems and matches how the student chooses its action at deployment. Using the teacher as a reference model does not help either, since the teacher's probabilities after the student's reasoning are not a reliable prior at the states the student visits. In Table 16, the success rate peaks at $\rho = 0.5$, which suggests that a smaller weight gives too little signal from the harness contrast while a larger weight may pull the small student away from the task.

5 Conclusion

We have presented Harness-Aware Distillation (HAD), a distillation framework for small language model agents that are deployed with a harness. HAD starts from the view that distilling such an agent is a problem between two systems that share the same harness, so the student mainly needs the teacher's ability to act on what the harness reports. It complements on-policy distillation with an action preference that contrasts the same teacher's actions with and without the harness information, and with a validity check that drops pairs whose preferred action contradicts the harness records. Across multiple long-horizon agent tasks, our method achieves the best performance among on-policy distillation methods equipped with the same harness.

Future work and limitations. We evaluated HAD with students of up to 2B parameters and teachers of up to 30B parameters in text-based environments; extending it to larger models and tool-calling or software agents remains future work. Our harness is also designed by hand, and harnesses that are learned or optimized automatically may interact with HAD differently. More broadly, HAD reflects a principle that distillation into a system with a fixed component should focus on what that component cannot provide, which may also apply to agents that rely on retrieval or external tools.

AI USE STATEMENT

We used large language models to assist in checking the correctness of mathematical derivations and proofs for mathematical claims presented in this paper. They were used as a supplementary verification tool rather than as a source of mathematical claims or proofs; the authors reviewed and independently verified the final mathematical arguments. We also used large language models to assist with editing the text for clarity and readability and with preparing figures. The authors designed the research, developed the methodology, ran all experiments, analyzed and interpreted the results, and take full responsibility for the final content of this paper.

ETHICS STATEMENT

This work distills existing open-weight language models in simulated text environments and involves no human subjects or personal data. All benchmarks and models we use are publicly available under their respective licenses. Our method is a general distillation technique, and its societal impact is mediated through the agents it is applied to, without introducing new risks beyond those of the underlying models.

REPRODUCIBILITY STATEMENT

Appendix B.4 lists all hyperparameters and the implementation details, and Appendix B.1 describes the environments and their versions. Appendix B.2 gives the harness templates, and Appendix C shows the prompts used by the agents. We provide the code, the harness, and the training scripts as anonymous supplementary material.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations (ICLR)*, 2024.

Peter Belcak, Greg Heinrich, Shizhe Diao, Yonggan Fu, Xin Dong, Saurav Muralidharan, Yingyan Celine Lin, and Pavlo Molchanov. Small language models are the future of agentic ai, 2025. URL https://arxiv.org/abs/2506.02153v3.

Baian Chen, Chang Shu, Ehsan Shareghi, Nigel Collier, Karthik Narasimhan, and Shunyu Yao. FireAct: Toward language agent fine-tuning, 2023. URL https://arxiv.org/abs/2310.05915v1.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, Mayank Chaturvedi, Aditya Chawla, Victor Cotruta, Alice Coucke, Phil Culliton, Robert Dadashi, Lucas Dixon, Mohamed Elhawaty, Utku Evci, Clément Farabet, Johan Ferret, Filippo Galgani, Sertan Girgin, Jean-Bastien Grill, Maarten Grootendorst, Jiaxian Guo, Cassidy Hardin, Yanzhang He, Steven M. Hernandez, Omri Homburger, Léonard Hussenot, Juyeong Ji, Armand Joulin, Aishwarya Kamath, Parnian Kassraie, Olivier Lacombe, Preethi Lahoti, Gaël Liu, Gus Martins, Luciano Martins, Tatiana Matejovicova, Ramona Merhej, Nikola Momchev, Sneha Mondal, Ryan Mullins, Sindhu Raghuram Panyam, Shreya Pathak, Sarah Perrin, André Susano Pinto, Etienne Pot, Angéline Pouget, Alexandre Ramé, Sabela Ramos, Douglas Reid, David Rim, Morgane Rivière, Karsten Roth, Louis Rouillard, Omar Sanseviero, Pier Giuseppe Sessa, Shane Settle, Danila Sinopalnikov, Sara Smoot, Piotr Stanczyk, Andreas Steiner, Lawrence Stewart, Ilya Tolstikhin, Michael Tschannen, Anton Tsitsulin, Nino Vieillard, Renjie Wu, Pingmei Xu, Haichuan Yang, Edouard Yvinec, Biao Zhang, Li Zhang, Joe Zou, Nicolas Aagnes, Abdelrahman Abdelhamed, Jakub Adamek, Shivani Agrawal, Shubham Agrawal, Ibrahim Alabdulmohsin, Jean Baptiste Alayrac, Uri Alon, Chandramouli Amarnath, Ankesh Anand, Chrysovalantis Anastasiou, Setareh Ariafar, François-Xavier Aubet, Kyriakos Axiotis, Federico Barbero, Joelle Barral, Alexei Bendebury, Urs Bergmann, Stanley Bileschi, Kat Black, Mathieu Blondel, Sebastian Borgeaud, Arthur Bražinskas, Ryan Burnell, Robert Busa-Fekete, Mu Cai, Daniele Calandriello, Glenn Cameron, Charlotte Caucheteux, Rahma Chaabouni, Garima Chadha, Jetha Chan, Blake Jianhang Chen, Jesse Chen, Lin Chen, Xu Chen, Derek Cheng, Tzu hsiang Chien, Nikolai Chinaev, Yi Chou, Zhaohui Chu, Benjamin Coleman, Pooja Consul, Sam Conway-Rahman, Scott Crowell, Dylan Cutler, Vivek Dani, Samira Daruki, Anil Das, Daniel Deutsch, Nishanth Dikkala, Li Ding, Qiuhan Ding, Shenil Dodhia, Konstantin Donhauser, Tulsee Doshi, Anca Dragan, Alex Druinsky, Sahil Dua, Zoltan Egyed, Danielle Eisenbud, Daniel Eppens, Cindy Fan, Bahare Fatemi, Yassir Fathullah, Vlad Feinberg, Milen Ferev, Sebastian Flennerhag, Takumi Fujimoto, João Gabriel Oliveira, Isaac Galatzer-Levy, João Gante, Simon Geisler, Soham Ghosal, Antonious M. Girgis, Tamara von Glehn, Alec Go, Alhaad Gokhale, Alex Grills, Yiming Gu, Mayank Gupta, Pramod Gupta, Guru Guruganesh, Raia Hadsell, Hamza Harkous, Jitendra Harlalka, Demis Hassabis, Anja Hauth, Joe Heyward, Arian Hosseini, Chih-Yang Hsia, I-Hung Hsu, Xiaopeng Huang, Yangsibo Huang, Kevin Hui, Adrian Hutter, Te I, Fotis Iliopoulos, Advait Jain, Ganesh Jawahar, Ziwei Ji, Qilin Jin, Melvin Johnson, Kandarp Joshi, Arun Kandoor, Wang-Cheng Kang, Koray Kavukcuoglu, Mehran Kazemi, Kathleen Kenealy, Amr Khalifa, Phoebe Kirk, Ivan Korotkov, Suraj Kothawade, Vitaly Kovalev, Neel Kovelamudi, Adam Kraft, Ravin Kumar, Vivek Kumar, Harish Kuppam, Justin Lannin, Chen-Yu Lee, Seungji Lee, Dmitry Lepikhin, Alon Levkovitch, Dongdong Li, Qiujia Li, Valentin Liévin, Ethan Lin, Ziqian Lin, Casper Liu, Tianlin Liu, Tianqi Liu, Xin Liu, Ivan Lobov, Mayank Lunayach, Min Ma, Gagan Madan, Andrii Maksai, Eric Malmi, Michal Matuszak, Daniel McDuff, Gaurav Menghani, Maciej Mikuła, Daniil Mirylenka, Karolis Misiunas, Vedant Misra, Andreea Mitran, Kareem Mohamed, Maksim Mukha, Eric Noland, James O'Donnell, Brendan O'Donoghue, Kate Olszewska, Bernett Orlando, Wanqiong Pan, Rina Panigrahy, Unnati Parekh, Nicolas Perez-Nieves, Chunjong Park, Eric Paskie, Liqian Peng, Bryce Petrini, Slav Petrov, Jonas Pfeiffer, Bilal Piot, Martyna Plomecka, Siim Poder, Octavio Ponce, Arijit Pramanik, David Racz, Anish Rajan, Michelle Ramanovich, Anand Rao, Marvin Ritter, Vitor Rodrigues, Evan Rosen, Mikołaj Rybiński, Noveen Sachdeva, Michaël E. Sander, Rohit Sathyanarayana, Sagar Savla, Samuel Schmidgall, Tal Schuster, George Scrivener, Benoit Seguin, Andrew Sellergren, Aliaksei Severyn, Izhak Shafran, Dhruv Shah, Bobak Shahriari, Yuan Shang-

guan, Ashish Shenoy, Pradeep Shenoy, Rakesh Shivanna, Pauline Sho, Lucas Spangher, Wojciech Stokowiec, Tim Strother, Yao Su, Yinghao Sun, Mukund Sundararajan, Andrea Tacchetti, Mor Hazan Taege, Pouya Tafti, Jean Tarbouriech, Chetan Tekur, Shantanu Thakoor, Rahul Thapa, Madeleine Traverse, Lenart Treven, Tao Tu, Chien Te Tung, Çağlar Ünlü, Petar Veličković, Malini Pooni Venkat, Sagar Gubbi Venkatesh, Vidya Venkiteswaran, Francesco Visin, Alex Vitvitskyi, Kiran Vodrahalli, Weiyi Wang, Xin Wang, Tris Warkentin, Jan Wassenberg, John Wieting, Cindy Wu, Lechao Xiao, Hao Xu, Yuhui Xu, Fuzhao Xue, Arun Yadav, Jun Yan, Antoine Yang, Lin Yang, Ming-Hsuan Yang, Ziyu Ying, Jae Hyeon Yoo, Morteza Zadimoghaddam, Sajjad Zafar, Fred Zhang, Jiageng Zhang, Jianyi Zhang, Xiaofan Zhang, Chao Zhao, David Zhou, and Chen Zou. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770v2.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531v1.

Minki Kang, Jongwon Jeong, Seanie Lee, Jaewoong Cho, and Sung Ju Hwang. Distilling LLM agent into small models with retrieval and code tools. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

Kyungmin Kim, Youngbin Choi, Seoyeon Lee, Suhyeon Jun, Dongwoo Kim, and Sangdon Park. The interplay of harness design and post-training in LLM agents, 2026. URL https://arxiv.org/abs/2606.25447v1.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

Yoonho Lee, Roshen Sanjay Nair, Qizheng Zhang, Kangwook Lee, Omar Khattab, and Chelsea Finn. Meta-Harness: End-to-end optimization of model harnesses. In *Conference on Language Modeling (COLM)*, 2026.

Gengsheng Li, Mao Zheng, Mingyang Song, Ruiqi Liu, Tianyu Yang, Jie Sun, Qiyong Zhong, Haiyun Guo, Junfeng Fang, Dan Zhang, and Jinqiao Wang. On-policy distillation with curriculum turn-level guidance for multi-turn agents, 2026. URL https://arxiv.org/abs/2606.15912v2.

Yixing Li, Yuxian Gu, Li Dong, Dequan Wang, Yu Cheng, and Furu Wei. Direct preference knowledge distillation for large language models, 2024. URL https://arxiv.org/abs/2406.19774v2.

Jun Liu, Zhenglun Kong, Peiyan Dong, Changdi Yang, Tianqin Li, Yanyue Xie, Yifan Gong, Xuan Shen, Pu Zhao, Hao Tang, Geng Yuan, Wei Niu, Wenbin Zhang, Xue Lin, Dong Huang, and Yanzhi Wang. Structured agent distillation for large language model agents. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2026.

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Cote, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. ALFWorld: Aligning text and embodied environments for interactive learning. In *International Conference on Learning Representations (ICLR)*, 2021.

Changhui Sun, Lanbo Liu, Hang Lei, Tong Ling, Jiahang Xie, Zhiyong Zheng, Yujia Wang, Hao Liu, Feng Xiao, Lu Liu, Yanlong Du, Zifeng Cheng, Ziwei Jiang, and Qing Gu. Step-level on-policy distillation: Interpolating between on-policy distillation and supervised fine-tuning, 2026. URL https://arxiv.org/abs/2608.16333v1.

Songjun Tu, Chengdong Xu, Qichao Zhang, Yiwen Ma, Yaocheng Zhang, Linjing Li, Dong Li, Xiangyuan Lan, and Dongbin Zhao. UCOB: Learning to utilize and evolve agentic skills via credit-aware on-policy bidirectional self-distillation, 2026. URL https://arxiv.org/abs/2606.29502v2.

Jiaqi Wang, Wenhao Zhang, Weijie Shi, Yaliang Li, and James Cheng. Exploring temporal curriculum in on-policy distillation for multi-turn autonomous agents. In *Conference on Language Modeling (COLM)*, 2026.

Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. ScienceWorld: Is your agent smarter than a 5th grader? In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

Xingyao Wang, Boxuan Li, Yufan Song, Frank F. Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, Hoang H. Tran, Fuqiang Li, Ren Ma, Mingzhang Zheng, Bill Qian, Daniel Shao, Niklas Muennighoff, Yizhe Zhang, Binyuan Hui, Junyang Lin, Robert Brennan, Hao Peng, Heng Ji, and Graham Neubig. OpenHands: An open platform for AI software developers as generalist agents. In *International Conference on Learning Representations (ICLR)*, 2025.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *International Conference on Machine Learning (ICML)*, 2024.

Shi-Qi Yan, Quan Liu, and Zhen-Hua Ling. RPO: Retrieval preference optimization for robust retrieval-augmented generation. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388v1.

John Yang, Carlos Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-agent: Agent-computer interfaces enable automated software engineering. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024a.

Kevin Yang, Dan Klein, Asli Celikyilmaz, Nanyun Peng, and Yuandong Tian. RLCD: Reinforcement learning from contrastive distillation for LM alignment. In *International Conference on Learning Representations (ICLR)*, 2024b.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. WebShop: Towards scalable real-world web interaction with grounded language agents. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models, 2026. URL https://arxiv.org/abs/2602.12275v2.

Xin Yu, Liuchen Liao, Yiwen Zhang, Yingchen Yu, Lingzhou Xue, and Qinzhen Guo. Preference-based self-distillation: Beyond KL matching via reward regularization, 2026. URL https://arxiv.org/abs/2605.05040v2.

Aohan Zeng, Mingdao Liu, Rui Lu, Bowen Wang, Xiao Liu, Yuxiao Dong, and Jie Tang. AgentTuning: Enabling generalized agent abilities for LLMs. In *Findings of the Association for Computational Linguistics: ACL*, 2024.

Zhengyang Zhao, Lu Ma, and Wentao Zhang. Training with harnesses: On-policy harness self-distillation for complex reasoning, 2026. URL https://arxiv.org/abs/2605.08741v1.

Yuhang Zhou, Lizhu Zhang, Yifan Wu, Mingyi Wang, Bo Peng, Jiayi Liu, Xiangjun Fan, and Zhuokai Zhao. SAGE-OPD: Selective agent-guided intervention for multi-turn on-policy distillation, 2026. URL https://arxiv.org/abs/2606.19659v1.

A Additional Supervision from Harness Awareness

A.1 Practical Implementation

Scaling preference margins (β). The margin Δ_t compares mean log-likelihoods over action tokens, so its scale does not grow with the length of the compared actions. The coefficient β then sets how quickly the logistic loss saturates in this margin. With a small β, the loss is close to linear in Δ_t and every retained pair contributes a similar gradient, whereas a large β concentrates the gradient on pairs whose margin is still small or negative. Since $\lambda_{\mathcal{B}}$ rescales the preference gradient to a fixed fraction of the distillation gradient, β mainly changes how this gradient is distributed across pairs rather than its overall size. We use $\beta = 0.5$ in all environments, selected together with ρ on a validation split held out from the training tasks. When the two teacher actions are identical, $\Delta_t = 0$ for every θ, so the pair contributes a constant loss and no gradient.

Filtering invalid actions (m_t). The checker V judges the positive teacher action $\tilde{a}_t^{\phi}$ only against the records that the harness computes at the visited state $\tilde{x}_t^{\theta}$, and uses no task rewards or future information. At a high level, it rejects an action that the harness records contradict, that is, an action the harness itself would report as impossible or ineffective at this state. Examples include an action outside the available actions that the harness lists, an action that uses an object the agent does not hold, and an action that repeats one reported as having had no effect at the same state. The checks for each environment are listed in Appendix B.2 and Appendix B.4. Negative actions are not checked, since lowering the likelihood of an invalid action does not harm the student. A rejected pair is removed from the preference term only, and the teacher response at that state still enters $\ell_{\mathrm{dist},t}$. The checker is therefore conservative, as it removes actions that contradict the records but cannot detect actions that are valid yet do not advance the task (Section 3.2).

Balancing distillation and preference losses (λ). In practice, we compute the mini-batch losses

$$D_{\mathcal{B}} = \frac{1}{n}\sum_{t\in\mathcal{B}} \ell_{\mathrm{dist},t}, \qquad P_{\mathcal{B}} = \frac{1}{n_V}\sum_{t\in\mathcal{B}} m_t \ell_{\mathrm{pref},t}, \qquad n_V = \sum_{t\in\mathcal{B}} m_t, \tag{8}$$

for a minibatch $\mathcal{B}$ of n records, where $P_{\mathcal{B}} = 0$ when $n_V = 0$, and perform backpropagation on the combined loss $D_{\mathcal{B}} + \lambda_{\mathcal{B}} P_{\mathcal{B}}$. Rather than using a fixed value of $\lambda_{\mathcal{B}}$, we introduce a per-mini-batch heuristic that scales the newly introduced preference loss-gradient so that its contribution to the parameter update is approximately half that of the original distillation loss-gradient:

$$\lambda_{\mathcal{B}} = \begin{cases} \mathrm{stopgrad}\left[\rho \dfrac{\|\nabla_\theta D_{\mathcal{B}}\|_2}{\|\nabla_\theta P_{\mathcal{B}}\|_2}\right], & \nabla_\theta P_{\mathcal{B}} \neq 0, \\ 0, & \nabla_\theta P_{\mathcal{B}} = 0, \end{cases} \qquad \rho = 0.5. \tag{9}$$

A.2 Ablation Variants

Response distillation alone and no filtering. Setting $\lambda = 0$ removes the auxiliary term, whereas setting $m_t \equiv 1$ retains every positive preference candidate.

Positive-only action supervision. Replace $\ell_{\mathrm{pref},t}$ by

$$\ell_{\mathrm{pos},t} = -\frac{1}{|\tilde{a}_t^{\phi}|} \log \pi_\theta(\tilde{a}_t^{\phi} \mid \tilde{x}_t^{\theta}, \tilde{z}_t^{\theta}), \tag{10}$$

retaining the positive teacher actions, student reasoning prefixes, length normalization, and validity masks. This preserves direct student-prefix supervision without the stripped-history comparison.

Full-response preferences. Replace Δ_t and $\ell_{\mathrm{pref},t}$ by

$$\Delta_{\mathrm{full},t} = \frac{1}{|\tilde{y}_t^{\phi}|} \log \pi_\theta(\tilde{y}_t^{\phi} \mid \tilde{x}_t^{\theta}) - \frac{1}{|y_t^{\phi}|} \log \pi_\theta(y_t^{\phi} \mid \tilde{x}_t^{\theta}), \tag{11}$$

$$\ell_{\mathrm{full},t} = -\log \sigma(\beta \Delta_{\mathrm{full},t}). \tag{12}$$

Each target contains its own teacher reasoning and action, and both are scored under the same augmented history. For $y = (z, a)$, autoregressive factorization gives

$$\frac{\log \pi_\theta(z, a \mid \tilde{x})}{|z| + |a|} = \frac{\log \pi_\theta(z \mid \tilde{x}) + \log \pi_\theta(a \mid \tilde{x}, z)}{|z| + |a|}. \tag{13}$$

Thus this variant changes both the scored span and the action-conditioning prefix, rather than merely adding reasoning supervision to an otherwise identical comparison.

A.3 What Harness-Aware Action Preference Learning Adds

We fix a realized harness-induced history $\tilde{x}^\theta$, its stripped counterpart x^θ, and the student reasoning prefix $z_s = \tilde{z}^\theta$. Write the teacher response laws and action marginals as

$$J_+(z, a) = \pi_\phi(z, a \mid \tilde{x}^\theta), \quad P_+(a) = \textstyle\sum_z J_+(z, a), \tag{14}$$

$$J_-(z, a) = \pi_\phi(z, a \mid x^\theta), \quad P_-(a) = \textstyle\sum_z J_-(z, a). \tag{15}$$

Teacher draws are independent of each other and of current student reasoning conditional on the history. Let $Q(z, a) = Q_Z(z)Q_A(a \mid z)$ be a student response law at $\tilde{x}^\theta$, where z includes the boundary before the action. Set $L(a) = |a| > 0$ and $D(z, a) = |(z, a)| > 0$, and take student likelihoods to be positive wherever they are scored. The response-distillation risk is

$$\mathcal{D}(Q) = -\sum_{z,a} \frac{J_+(z, a)}{D(z, a)} \log Q(z, a). \tag{16}$$

It depends on the full positive response law J_+, but not on the negative teacher law. Zero-weight imitation terms are interpreted as zero, including at zero student probabilities.

Definition A.1 (Unfiltered preference risk)**.** For probability laws p_+ and p_- on a common finite, nonempty alphabet $\mathcal{S}$, independent draws $\xi^+ \sim p_+$ and $\xi^- \sim p_-$, and scores $g : \mathcal{S} \to \mathbb{R}$, define

$$\mathcal{R}_{p_+,p_-}(g) = \mathbb{E}[-\log \sigma(g(\xi^+) - g(\xi^-))]. \tag{17}$$

For action comparisons, define the unweighted, unmasked preference risk and the action-preference score by

$$\mathcal{R}(f) = \mathcal{R}_{P_+,P_-}(f), \qquad f_Q(a) = \frac{\beta}{L(a)} \log Q_A(a \mid z_s), \quad \beta > 0. \tag{18}$$

Proposition A.1 (Bayes-optimal action-preference scores)**.** *Let P_+ and P_- be strictly positive probability laws on a finite, nonempty action set $\mathcal{A}$, with independent sampling. The minimizers of $\mathcal{R}$ over all $f : \mathcal{A} \to \mathbb{R}$ are exactly*

$$f^*(a) = s(a) + \kappa, \qquad s(a) = \log \frac{P_+(a)}{P_-(a)}, \quad \kappa \in \mathbb{R}. \tag{19}$$

Proof. Write $w_{ab} = P_+(a)P_-(b)$, $g(a, b) = f(a) - f(b)$, and

$$\mu(a, b) = \frac{w_{ab} + w_{ba}}{2}, \qquad \eta(a, b) = \frac{w_{ab}}{w_{ab} + w_{ba}}. \tag{20}$$

Symmetrizing the risk gives

$$\mathcal{R}(f) = \sum_{a,b} \mu(a, b)\, \mathrm{CE}(\eta(a, b), \sigma(g(a, b))), \tag{21}$$

where $\mathrm{CE}(p, v) = -p \log v - (1 - p) \log(1 - v)$. The weights μ sum to one, and full support makes every weight positive and every $\eta(a, b)$ strictly between zero and one. Each cross-entropy is uniquely minimized at its target probability, with log-odds

$$\log \frac{\eta(a, b)}{1 - \eta(a, b)} = s(a) - s(b). \tag{22}$$

Thus $f = s$ minimizes all terms simultaneously, and equality requires $f(a) - f(b) = s(a) - s(b)$ for every pair. Equivalently, $f - s$ is constant. □

The argument also yields the exact excess-risk identity

$$\mathcal{R}(f) - \mathcal{R}(f^*) = \mathbb{E}_{(a,b)\sim\mu} D_{\mathrm{KL}}(\mathrm{Ber}(\eta(a,b)) \,\|\, \mathrm{Ber}(\sigma(f(a) - f(b)))) \,. \tag{23}$$

This calibrates scores to teacher-input provenance, not to measured action utility.

Observation A.1 (Contrast is not identified by response imitation)**.** The positive response law J_+ determines $\mathcal{D}$, but does not generally determine the optimal action-preference score differences, which also depend on P_-.

For example, fix $J_+(z_0, a_1) = J_+(z_0, a_2) = 1/2$ and consider two negative action laws,

$$P_-^{(1)} = (3/4, 1/4), \qquad P_-^{(2)} = (1/4, 3/4). \tag{24}$$

The distillation risks agree for every Q, whereas Proposition A.1 gives

$$f_1^*(a_1) - f_1^*(a_2) = -\log 3, \qquad f_2^*(a_1) - f_2^*(a_2) = \log 3. \tag{25}$$

The additional query therefore identifies a contrast absent from the positive response law alone. This does not preclude useful harness-dependent behavior learned through response imitation.

A.4 Regularization Bias and Reference-Relative Preferences

Regularization bias in isolation. Under the assumptions of Proposition A.1, take $V \equiv 1$ and a common action length L. When $P_+ = P_-$, the optimal scores are constant, so the isolated preference optimum over normalized action laws is uniform even if the common teacher law is not. For example, if $P_+ = P_- = (0.9, 0.1)$ over two actions, the positive–negative pairs (a_1, a_2) and (a_2, a_1) each occur with probability 0.09. These symmetric comparisons yield the preference-only optimum $(0.5, 0.5)$, rather than the teacher's $(0.9, 0.1)$. Thus reference-free preferences can flatten action probabilities even without a harness-induced action contrast.

A reference-relative variant. To address this bias, one can introduce a fixed reference policy π_{ref} that assigns positive probability to every scored action, define

$$\Delta_{\mathrm{rel},t}(\theta) = \frac{1}{|\tilde{a}_t^\phi|} \log \frac{\pi_\theta(\tilde{a}_t^\phi \mid \tilde{x}_t^\theta, \tilde{z}_t^\theta)}{\pi_{\mathrm{ref}}(\tilde{a}_t^\phi \mid \tilde{x}_t^\theta, \tilde{z}_t^\theta)} - \frac{1}{|a_t^\phi|} \log \frac{\pi_\theta(a_t^\phi \mid \tilde{x}_t^\theta, \tilde{z}_t^\theta)}{\pi_{\mathrm{ref}}(a_t^\phi \mid \tilde{x}_t^\theta, \tilde{z}_t^\theta)}, \tag{26}$$

$$\ell_{\mathrm{rel},t}(\theta) = -\log \sigma(\beta \Delta_{\mathrm{rel},t}(\theta)) \,, \tag{27}$$

and replace $\ell_{\mathrm{pref},t}$ by $\ell_{\mathrm{rel},t}$ in Equation 7, leaving all other components unchanged. Under the assumptions above, with student and reference action laws normalized on $\mathcal{A}$, $P_+ = P_-$ now yields the reference law as the isolated optimum.

However, choosing the reference policy is nontrivial: a teacher reference evaluates continuations conditioned on student-generated reasoning, rather than under the marginal P_+, while a frozen student may itself make poor use of the harness. Thus, neither provides a canonical prior over the decision contexts visited by the student. Consistent with this concern, an ablation that sets $\pi_{\mathrm{ref}} = \pi_\phi$ yields worse empirical performance (cf. Appendix D.3). For equal-length actions, reference-free scoring is equivalent to the uniform reference $\pi_{\mathrm{ref}} \equiv 1/|\mathcal{A}|$. Note that HAD also retains direct teacher supervision through response distillation, with the preference gradient balanced against that signal. The flattening bias therefore remains an auxiliary regularization trade-off; the isolated uniform optimum does not imply a uniform solution for the combined objective.

A.5 What Harness-Aware Validity Filtering Adds

Definition A.2 (Filtered preference risk)**.** For a fixed checker $V : \mathcal{A} \to \{0, 1\}$ and independent teacher actions, define

$$\mathcal{R}_V(f) = \sum_{a,b} V(a) P_+(a) P_-(b) \log\Big(1 + e^{f(b) - f(a)}\Big) \,. \tag{28}$$

Observation A.2 (Positive-action filtering)**.** Under the full-support and independence assumptions of Proposition A.1, fix a checker V and let $\gamma = \mathbb{E}_{P_+}[V(a)]$. The filtered risk in Definition A.2 satisfies, for $\gamma > 0$,

$$\mathcal{R}_V(f) = \gamma \mathcal{R}_{P_+^V, P_-}(f), \qquad P_+^V(a) = \frac{V(a) P_+(a)}{\gamma}. \tag{29}$$

For $\gamma = 0$, the risk is identically zero. For each accepted–rejected pair, only the comparison favoring the accepted action remains, and its contribution strictly decreases as the accepted action's score margin increases.

Proof. Substituting $V(a)P_+(a) = \gamma P_+^V(a)$ into the filtered risk gives the identity; when $\gamma = 0$, all positive-candidate weights vanish. If $V(a) = 1$ and $V(b) = 0$, the oriented comparison weights are $w_{ab}^V = P_+(a)P_-(b) > 0$ and $w_{ba}^V = 0$. The retained comparison contributes

$$w_{ab}^V \log\left(1 + e^{f(b)-f(a)}\right), \tag{30}$$

whose derivative with respect to $f(a) - f(b)$ is strictly negative. □

Without filtering, a rejected action can have a larger P_+/P_- ratio than an accepted action. Validity filtering addresses this failure of harness access as a preference proxy, without requiring checks on negative candidates during training.

A.6 Why Action-Only Preference Rather Than Full-Response Preference

Section 3.3 motivates a conditional action comparison at the student's own reasoning prefix. Full-response preferences instead score each candidate together with its teacher reasoning, changing both the scored span and the action-conditioning context. We show that this substitution can change the preferred actions even when teacher action marginals stay fixed. Appendix A.7 then formalizes how the conditional comparison complements response distillation. Throughout, teacher reasoning still generates the candidates and the complete positive response remains a distillation target.

Definition A.3 (Full-response preference risk)**.** Let J_+ and J_- be probability laws on a common finite, nonempty response set $\mathcal{Y}$, whose action projection is $\mathcal{A}$, with independent teacher draws. Definition A.1 gives the unweighted, unmasked full-response risk and score

$$\mathcal{R}_{\text{full}}(F) = \mathcal{R}_{J_+,J_-}(F), \qquad F_Q(z,a) = \frac{\beta}{D(z,a)} \log Q(z,a). \tag{31}$$

This is the population counterpart of Equation 12 with $V \equiv 1$.

Corollary A.1 (Full-response contrast and conditional action odds)**.** *Under Definition A.3, assume J_+ and J_- are strictly positive on $\mathcal{Y}$. All minimizers of $\mathcal{R}_{\text{full}}$ over unrestricted scores are*

$$F^*(z,a) = \log \frac{J_+(z,a)}{J_-(z,a)} + \kappa = \underbrace{\log \frac{P_+(a)}{P_-(a)}}_{\textit{action contrast}} + \underbrace{\log \frac{J_+(z \mid a)}{J_-(z \mid a)}}_{\textit{conditional reasoning contrast}} + \kappa, \quad \kappa \in \mathbb{R}. \tag{32}$$

If additionally $\mathcal{Y} = \mathcal{Z} \times \mathcal{A}$ and all actions have length L, write $d_z = |z| + L$ and $r_z(a) = \log[J_+(z \mid a)/J_-(z \mid a)]$. Let q^*_{act} and Q^*_{full} be the isolated unfiltered optima over positive laws normalized on $\mathcal{A}$ and $\mathcal{Y}$, respectively. Then, for every z, a, b,

$$\frac{\beta}{L} \log \frac{q^*_{\text{act}}(a)}{q^*_{\text{act}}(b)} = s(a) - s(b), \tag{33}$$

$$\frac{\beta}{d_z} \log \frac{Q^*_{\text{full}}(a \mid z)}{Q^*_{\text{full}}(b \mid z)} = s(a) - s(b) + r_z(a) - r_z(b). \tag{34}$$

Proof. Applying Proposition A.1 to the response alphabet and factoring $J_\pm(z,a) = P_\pm(a)J_\pm(z \mid a)$ gives the joint score target. For positive student laws normalized on $\mathcal{Y}$, Equation 49 gives

$$Q^*_{\text{full}}(z,a) = \exp\left[\frac{D(z,a)}{\beta}\left(\log \frac{J_+(z,a)}{J_-(z,a)} + \kappa_{\text{full}}\right)\right], \tag{35}$$

with the unique normalizing constant characterized in Section A.8. For equal action lengths, Proposition A.1 gives $(\beta/L)\log q^*_{\text{act}}(a) = s(a) + \kappa_{\text{act}}$, while the joint optimum gives $(\beta/d_z)\log Q^*_{\text{full}}(z,a) = s(a) + r_z(a) + \kappa_{\text{full}}$. Subtracting scores at a shared prefix cancels the normalizing constants and the reasoning marginal $Q^*_{\text{full},Z}(z)$, proving Equation 34. □

These identities concern isolated preference optima, not optima of the combined objective. Because both length factors are positive, an opposite action ordering requires the additional term $r_z(a)-r_z(b)$ to change the sign of the action contrast. A reasoning contrast constant across actions at the given prefix cancels instead.

Reasoning-based source discrimination. For any action score f, its lift $F_f(z,a) = f(a)$ satisfies $\mathcal{R}_{\text{full}}(F_f) = \mathcal{R}(f)$ after marginalizing reasoning. Taking $f = s$ therefore yields

$$\min_F \mathcal{R}_{\text{full}}(F) \le \min_f \mathcal{R}(f). \tag{36}$$

Equality holds exactly when the lifted action optimum is also a full-response optimum, requiring $\log(J_+(z \mid a)/J_-(z \mid a))$ to be a single constant on $\mathcal{Y}$. Normalization of each conditional law forces that constant to be zero, so equality is equivalent to $J_+(\cdot \mid a) = J_-(\cdot \mid a)$ for every action. If $P_+ = P_-$, the optimal action risk is $\log 2$, but the optimal full-response risk is strictly smaller whenever the conditional reasoning laws differ. For example, $J_\pm(z,a) = P(a)H_\pm(z)$ with distinct, full-support H_+ and H_- yields a full-response target depending only on reasoning. A lower full-response risk can therefore reflect reasoning-based source discrimination without an action-marginal contrast. This alone need not change the conditional action ordering: the relevant question is whether the additional reasoning contrast differs across actions.

Marginal versus reasoning-conditioned contrasts. Under the same full-support assumptions, Bayes' rule gives

$$s(a) + r_z(a) = \log \frac{J_+(a \mid z)}{J_-(a \mid z)} + c_z, \qquad c_z = \log \frac{J_{+,Z}(z)}{J_{-,Z}(z)}, \tag{37}$$

where $J_{\pm,Z}(z) = \sum_a J_\pm(z,a)$. The prefix-dependent constant cancels in action comparisons. Thus, for equal action lengths, the full-response optimum ranks actions at a shared prefix by a reasoning-conditioned action contrast, whereas the action-only optimum uses the marginal contrast s. The conditional contrast may encode useful reasoning–action compatibility; it is not necessarily a nuisance signal. Our choice of action-only scoring isolates the harness-induced action contrast after marginalizing teacher reasoning, rather than asserting that marginal contrasts always yield better actions.

Observation A.3 (Negative-reasoning invariance)**.** At a fixed history and student prefix, the combined objective $\mathcal{H}_V(Q;\lambda) = \mathcal{D}(Q) + \lambda\mathcal{R}_V(f_Q)$ depends on negative teacher responses only through P_-. Holding J_+, P_-, the checker V, and the weight $\lambda \ge 0$ fixed therefore leaves it unchanged for every admissible Q, including parameterized students and variable action lengths. This follows directly from the loss definitions and concerns negative teacher reasoning, not the student prefix used to score actions.

Proposition A.2 (Full-response preferences need not preserve action ordering)**.** *With J_+ and P_- fixed, changing only negative reasoning can reverse the full-response probability optimum's action ordering both at a shared prefix and marginally over reasoning, even with constant lengths and $V \equiv 1$. The reversed conditional ordering can oppose the action-preference optimum and incur strictly greater action-preference risk.*

Proof. Take two one-token actions a_1, a_2, two one-token reasoning prefixes z_0, z_1, $\beta = 2$, and $V \equiv 1$, with teacher laws

$$J_+ = \begin{pmatrix} 1/3 & 1/4 \\ 1/3 & 1/12 \end{pmatrix}, \qquad J_- = \begin{pmatrix} 1/6 & 1/24 \\ 1/6 & 5/8 \end{pmatrix}. \tag{38}$$

Rows index reasoning prefixes and columns index actions. Both laws have full support, with $P_+ = (2/3, 1/3)$ and $P_- = (1/3, 2/3)$. Equation 49, applied to action scores, gives $q^*_{\text{act}} = (2/3, 1/3)$ and score margin $\log 4$. Full responses have length two, so Equation 35 gives

$$Q^*_{\text{full}} = \begin{pmatrix} 15/76 & 45/76 \\ 15/76 & 1/76 \end{pmatrix}, \qquad Q^*_{\text{full}}(a_1 \mid z_0) = \frac{1}{4}. \tag{39}$$

This favors a_2 both at z_0 and marginally, since $\sum_z Q^*_{\text{full}}(z, a_1) = 15/38 < 1/2$. Its action-score margin at z_0 is $2\log((1/4)/(3/4)) = -\log 9$, rather than $\log 4$, giving strictly greater action risk by

Equation 23. Now replace J_- by $J_-^{(0)}(z,a) = P_-(a)J_+(z \mid a)$, leaving J_+ and P_- unchanged. The joint teacher ratio becomes $P_+(a)/P_-(a)$, independent of reasoning, and

$$Q^*_{\text{full},0}(a_1 \mid z_0) = \sum_z Q^*_{\text{full},0}(z, a_1) = \frac{4}{5} > \frac{1}{2}. \tag{40}$$

Thus negative reasoning alone reverses both full-response orderings while the conditional HAD objective stays fixed. □

The example holds the positive response law fixed, alongside the action marginals, lengths, and checker acceptance, isolating sensitivity to negative reasoning. It compares isolated preference optima; it does not establish the same reversal for the full-response variant after adding response distillation, nor does it order the variants by task utility. More generally, an action-only positive filter leaves $J_+(z \mid a)$ unchanged for accepted actions, so it does not remove their conditional reasoning contrast. Action-only preferences avoid directly contrasting teacher reasoning, but do not eliminate its role in producing the candidate actions or in the positive response-distillation term.

A.7 Action-Only Preferences at Student Reasoning Prefixes

We now formalize the coupled construction in Section 3.3: the student's sampled reasoning supplies the context, and teacher actions supply the preference candidates. The resulting loss supplements imitation of complete teacher responses with supervision of the student's conditional action choice. All probabilities below are conditional on the fixed harness-induced history.

Definition A.4 (Student-decision preference risk and excess risk)**.** Let $\mathcal{Z}$ be a common finite, nonempty reasoning-prefix alphabet. At every prefix, use the same finite action alphabet $\mathcal{A}$, positive action lengths $L(a)$, candidate laws $P_\pm$, and checker V. Let ω be a fixed distribution of collected student prefixes, and draw $z_s \sim \omega$, $(z^+, a^+) \sim J_+$, and $(z^-, a^-) \sim J_-$ independently. For student action conditionals that are positive laws normalized on $\mathcal{A}$, define

$$\begin{aligned}\mathcal{R}_{V,\omega}(Q) &= \mathbb{E}\Big[V(a^+)\log\Big(1 + e^{f_{Q,z_s}(a^-) - f_{Q,z_s}(a^+)}\Big)\Big] \\ &= \sum_z \omega(z)\mathcal{R}_V(f_{Q,z}), \qquad f_{Q,z}(a) = \frac{\beta}{L(a)}\log Q_A(a \mid z).\end{aligned} \tag{41}$$

For positive action laws q normalized on $\mathcal{A}$, write $f_q(a) = \beta \log q(a)/L(a)$ and $r^\star_V = \inf_q \mathcal{R}_V(f_q)$. Define the per-prefix and prefix-averaged excess risks by

$$e_V(Q,z) = \mathcal{R}_V(f_{Q,z}) - r^\star_V, \qquad \mathcal{E}_{V,\omega}(Q) = \sum_z \omega(z) e_V(Q,z) = \mathcal{R}_{V,\omega}(Q) - r^\star_V. \tag{42}$$

Using an infimum allows filtered risks whose best separating margins are attained only in a limit. These excesses concern the isolated action-preference term, not task utility. The prefix law is held fixed during the comparison, as are collected records during differentiation.

Observation A.4 (Response imitation and student-decision supervision)**.** Under Definition A.4, let $Q(z,a) = Q_Z(z)Q_A(a \mid z)$ be strictly positive on $\mathcal{Z} \times \mathcal{A}$. Write $w_+(z,a) = J_+(z,a)/D(z,a)$ and $w_{+,Z}(z) = \sum_a w_+(z,a)$. The response-distillation risk decomposes as

$$\mathcal{D}(Q) = -\sum_z w_{+,Z}(z)\log Q_Z(z) - \sum_{z,a} w_+(z,a)\log Q_A(a \mid z). \tag{43}$$

It therefore scores action conditionals under positive teacher reasoning. In contrast, for fixed ω, $\mathcal{R}_{V,\omega}(Q)$ depends on Q only through $Q_A(\cdot \mid z)$ at prefixes with $\omega(z) > 0$ and is invariant to changes in Q_Z with these action conditionals held fixed. For each retained pair, the preference loss is strictly decreasing as a function of its conditional action-score margin $f_{Q,z_s}(a^+) - f_{Q,z_s}(a^-)$.

Proof. Substituting $\log Q(z,a) = \log Q_Z(z) + \log Q_A(a \mid z)$ into Equation 16 gives the decomposition. Equation 41 contains only action conditionals at prefixes weighted by ω. For a retained pair with margin δ, the loss is $\log(1 + e^{-\delta})$, whose derivative is $-1/(1 + e^{\delta}) < 0$. No teacher- or student-reasoning likelihood appears as a separate score in that margin. □

The two scoring choices describe one conditional comparison: z_s is the context of the action law being compared, not another response span receiving a preference label. This is a property of the collected-data loss, not a claim that preference updates leave the student's reasoning unchanged: neural parameters may be shared, and resampling prefixes would change ω.

Corollary A.2 (Independent prefixes preserve the marginal action target)**.** *Under Definition A.4, whenever $\omega(z) > 0$, the conditional candidate law is $\Pr(a^+ = a, a^- = b \mid z_s = z) = P_+(a)P_-(b)$. Thus the same per-prefix risk $\mathcal{R}_V$ is used at every supervised prefix. If additionally $P_\pm$ have full support, $V \equiv 1$, and all actions have length L, the isolated optimum over independently variable positive normalized action conditionals is*

$$Q_A^*(a \mid z) = \frac{\big(P_+(a)/P_-(a)\big)^{L/\beta}}{\sum_b \big(P_+(b)/P_-(b)\big)^{L/\beta}}, \qquad \omega(z) > 0. \tag{44}$$

In particular, the isolated target is independent of the scoring prefix.

Proof. Independence of the prefix and the two teacher responses gives the product conditional law, so the masked candidate risk at each prefix is $\mathcal{R}_V$. For unfiltered comparisons with full support and equal action lengths, Proposition A.1 and Equation 49 give the stated optimum at every supervised prefix. □

Teacher reasoning generates the candidates but is marginalized in their action law; student reasoning specifies where the same marginal action contrast is taught, not which contrast is taught. This neither replaces $P_\pm$ by teacher continuation laws conditioned on z_s nor labels the student's reasoning as correct. Equation 44 is a relative-preference target, not generally the teacher action law P_+. With variable but prefix-independent action lengths, Equation 49 likewise yields the same isolated action law at every supervised prefix. A fixed action-only checker also leaves the per-prefix risk identical, although a filtered infimum may only be approached in a limit.

Importantly, our auxiliary preference objective favors consistency across student reasoning contexts but does not, by itself, preserve useful reasoning-dependent action variation. This does not imply that the combined HAD optimum is prefix-invariant, as it also includes response distillation and may constrain action conditionals through shared parameters.

Corollary A.3 (Reusing positive teacher reasoning changes the action target)**.** *Let $(z^+, a^+) \sim J_+$ and $(z^-, a^-) \sim J_-$ be independent teacher responses. If the positive reasoning z^+ is reused as the shared scoring prefix, then, for every z with $J_{+,Z}(z) > 0$, the conditional candidate law is $\Pr(a^+ = a, a^- = b \mid z^+ = z) = J_+(a \mid z)P_-(b)$. When $J_+(\cdot \mid z)$ and P_- have full support and $V \equiv 1$, the optimal unrestricted scores at that prefix are*

$$g_z^*(a) = \log \frac{J_+(a \mid z)}{P_-(a)} + \kappa_z = s(a) + \log \frac{J_+(z \mid a)}{J_{+,Z}(z)} + \kappa_z. \tag{45}$$

Proof. Conditioning the positive draw on its reasoning changes its action law to $J_+(\cdot \mid z)$, while independence leaves the negative action law equal to P_-. Proposition A.1 gives the first score identity, and Bayes' rule gives the second. □

This target adds compatibility with the selected positive reasoning trace to the marginal action contrast. An independently sampled student prefix avoids this extra conditioning; choosing the student rather than another independent prefix law additionally determines the weighting analyzed next. Full-response preferences are not simply Equation 41 under a different prefix law: they also change the scored span and couple each candidate to its own teacher reasoning.

Which prefixes receive supervision? We now hold the candidate laws $P_\pm$ and checker fixed, changing only the distribution of independently sampled scoring prefixes. Definition A.4 supplies the common per-prefix excess risk used to compare these distributions.

Proposition A.3 (Prefix coverage and excess-risk transfer)**.** *Let τ be another fixed prefix distribution used with the same action comparison, and define*

$$C_{\omega|\tau} = \max_{z:\omega(z)>0} \frac{\omega(z)}{\tau(z)}, \tag{46}$$

with the ratio interpreted as $+\infty$ when $\tau(z) = 0$. For finite $C_{\omega|\tau}$ and every admissible student Q,

$$\mathcal{E}_{V,\omega}(Q) \le C_{\omega|\tau}\, \mathcal{E}_{V,\tau}(Q). \tag{47}$$

If $\mathcal{R}_V(f_q)$ is nonconstant and positive action conditionals can vary independently, this is the smallest uniform factor; when $C_{\omega|\tau} = +\infty$, no finite factor suffices.

Proof. Every $e_V(Q, z)$ is nonnegative by definition of $r^\star_V$. For finite $C_{\omega|\tau}$, multiplying $\omega(z) \le C_{\omega|\tau}\tau(z)$ by these excesses and summing proves the bound. For sharpness, choose a prefix z_0 attaining the largest ratio and an action law with strictly positive excess there, which exists because the risk is nonconstant. At every other prefix, independently choose action laws whose risks approach $r^\star_V$. The ratio of averaged excesses then approaches $\omega(z_0)/\tau(z_0)$. If $\tau(z_0) = 0 < \omega(z_0)$, the τ-averaged excess can approach zero while the ω-averaged excess remains bounded away from zero, ruling out every finite factor. □

The bound applies to parameterized students too; only its sharpness uses independently variable conditionals. It addresses unequal coverage even when both prefix laws have full support. For example, with two prefixes, $\tau = (1-\varepsilon, \varepsilon)$, $\omega = (1/2, 1/2)$, $0 < \varepsilon < 1/2$, and excesses $(0, \delta)$ with $\delta > 0$, the two averaged excesses are $\varepsilon\delta$ and $\delta/2$. Such excesses are realizable in the unfiltered full-support setting, where the isolated minimum is attained. With full prefix coverage and unrestricted conditionals, alternative prefix weights share the same pointwise target and risk infimum. Their practical difference concerns residual errors due to finite data, limited model capacity, or imperfect optimization, rather than a better pointwise population target. Student-prefix sampling avoids this transfer step by using the collected prefix law ω itself; it does not guarantee small excess risk or improved task return.

Uncovered prefixes in teacher-forced losses. The decomposition in Observation A.4 also makes the zero-coverage case explicit for the actual response losses. If both teacher reasoning marginals vanish at z_s, varying $Q_A(\cdot \mid z_s)$ at fixed Q_Z and other action conditionals leaves both $\mathcal{D}$ and $\mathcal{R}_{\text{full}}$ unchanged: no positive-weight term scores a response there. Filtering positive actions cannot create this missing coverage. If $\omega(z_s) > 0$ and the per-prefix action risk is nonconstant, such a variation can nevertheless change $\mathcal{R}_{V,\omega}$ through Equation 41. This concerns directly scored conditionals, not the absence of generalization through shared neural parameters.

What the positive-only control retains. The loss in Equation 10 retains student-prefix supervision but removes the negative action contrast. At the fixed history, write $V(a) = V(a \mid \tilde{x}^\theta)$ for the binary checker and let q be a normalized action law on the finite teacher action set. Its isolated risk is $-\sum_a V(a)P_+(a) \log q(a)/L(a)$, whose optimum is

$$q^*_{\text{pos}}(a) = \frac{V(a)P_+(a)/L(a)}{\sum_b V(b)P_+(b)/L(b)} \tag{48}$$

when the denominator is positive; if it vanishes, the risk is identically zero. The weighted-imitation argument in Section A.8 proves the formula, allowing zero probabilities on zero-weight actions. Thus this control retains direct prefix supervision without learning the P_+/P_- contrast. It does not by itself isolate the effect of student rather than teacher reasoning conditioning.

A.8 From Scores to Probabilities

This section supplies the probability realizations used in the preceding comparisons. The score characterizations in Proposition A.1 and Equation 32 apply to mean token log-likelihoods, not directly to sequence probabilities. More generally, suppose a risk on a finite, nonempty alphabet $\mathcal{S}$ has exactly the optimal scores $g + \kappa$. For positive lengths $\ell(\xi)$ and scores $\beta \log p(\xi)/\ell(\xi)$, its unique optimum over positive probability laws normalized on $\mathcal{S}$ is

$$p^*(\xi) = \exp\left[\frac{\ell(\xi)}{\beta}\big(g(\xi) + \kappa\big)\right], \qquad \sum_{\xi \in \mathcal{S}} p^*(\xi) = 1. \tag{49}$$

The normalization sum is continuous, strictly increasing in κ, and ranges from zero to infinity, so exactly one constant realizes the optimal score differences. For action preferences, take $\mathcal{S} = \mathcal{A}$,

$g = s$, and $\ell = L$; for full-response preferences, take $\mathcal{S} = \mathcal{Y}$, $g = \log(J_+/J_-)$, and $\ell = D$. When ℓ is constant, $p^* \propto \exp(\ell g/\beta)$, preserving the score ordering; with variable lengths, probability and score orderings can differ. Without normalization on the specified alphabet, score calibration alone does not identify probability mass outside it. These are isolated preference-loss optima, not optima of the combined HAD objective. For comparison, minimizing a weighted imitation loss $-\sum_\xi w(\xi) \log p(\xi)$ gives $p^*(\xi) = w(\xi)/\sum_{\xi'} w(\xi')$ whenever $w \geq 0$ has positive total mass, allowing boundary probabilities. This follows by expressing the loss as that total mass times cross-entropy from the normalized weights.

B Experiment settings

B.1 Models and tasks

Table 8 summarizes the environments and models. All models are public checkpoints served with thinking disabled, and the Gemma pair uses the same prompts, harness, and training recipe as the Qwen3 pair on ALFWorld. Every environment asks the model for exactly two lines, `Thought: <brief reasoning>` followed by `Action: <one command>`. A reply without a parsable action consumes a turn. Training rollouts use temperature 1.0, and evaluations use temperature 0.4.

Table 8: Environments, models, and evaluation settings.

	ALFWorld	WebShop	ScienceWorld
Teacher	Qwen3-8B	Qwen3-30B-A3B	Qwen3-30B-A3B
Student	Qwen3-1.7B	Qwen3-0.6B	Qwen3-0.6B
Teacher (other family)	Gemma-4-12B	–	–
Student (other family)	Gemma-4-E2B	–	–
Test set	140 seen, 134 unseen	500 goals	149 tasks (30 types)
Training pool	3,553 tasks	11,587 goals	3,592 tasks
Step limit	30	15	15
Max reply tokens	2,048	512	128
Context length	24,576	24,576	16,384
History in the prompt	full	last 2 steps	full
Metrics	SR	score, SR	score, SR

ALFWorld. Each user turn contains the task, the observation, and the list of admissible commands, and the full history is kept as alternating turns. Harness lines are inserted only in the latest turn, between the observation and the admissible commands. The task line is the first human annotation of each trial rather than the templated goal, as in the original release. This annotation differs from the templated goal in 10 of the 274 evaluation tasks, all methods see the same text, and these tasks are not used for the qualitative examples.

WebShop. We use the 1,000-product version of the environment. The score is the mean final reward multiplied by 100, and the success rate is the percentage of episodes with reward 1.0.

ScienceWorld. We use version 1.2.3 with the teleport simplification and 149 test variations over 30 task types. The score is the mean final score, which can be negative, and the success rate is the percentage of episodes with a score of at least 100.

B.2 Harness

Design. All three environments share the same harness code and templates, and only the part that reads the environment state differs. The harness contains no hints about the task, such as subgoals or solution steps, and reports only what the environment and the agent's own history already reveal. It only adds text to the observation and never selects or blocks an action. Some environments already display part of this information in their own observations, such as the list of admissible commands in ALFWorld and WebShop and the interaction history in the prompt. Since the agent sees this information regardless of the harness, we keep it in every condition, including evaluation without the harness and with a harness category hidden, and the remaining components are added only when the harness is used.

Lines. Table 9 lists the lines the harness can add at each step. Each line has three paraphrases, and one of them is chosen at random for each firing with a seed that depends on the episode and step. The same templates are used in all environments.

Table 9: Harness lines. One of the three paraphrases is shown for each line.

Line	Function	Template (one paraphrase)	When it fires
Situation	State tracking	`You are at {location}. You are holding {holdings}. {m} of {n} places visited; {k} unvisited ({names}).`	Every step
Visit	State tracking	`This visit to {location}: arrived turn {a}, {c} commands executed since.` and, for a revisit, `Previous visit: arrived turn {a}, {c} commands executed, left turn {l}.`	Every step with a known location
Budget	State tracking	`Step {t} of {cap}.` and the warning `Only {left} steps remain. Commit to finishing the task now.`	Counter at every step, warning when fewer than θ steps remain
Failure	Behavior monitoring	`'{action}' did not work here (this was attempt {N}). It will keep failing at this state - choose a different action from the displayed list.`	The same action was refused or had no effect at the same state at least the minimum number of times
Stall	Behavior monitoring	`Nothing new has been observed for {k} steps. You are not making progress.`	No new state was reached in the last k steps

Thresholds. The thresholds are part of the harness design, which is shared by all methods, including the baselines, for both training and evaluation (Table 10). They are fixed once for each environment before any distillation, using a validation split held out from the training tasks, and no evaluation task is used. The stall window k is the smallest value, and the budget threshold θ the largest value, for which the corresponding line fires on at most 5% of the steps within successful validation episodes of an untrained model, so that the harness rarely interrupts an agent that is making progress. The minimum number of failed attempts is chosen with the same criterion. WebShop does not announce refused actions, so its failure line fires from the first attempt without effect. This calibration belongs to the harness and not to HAD, and the training objective of HAD uses no task rewards or success labels.

Table 10: Harness thresholds for each environment. They are part of the harness shared by all methods and are fixed before distillation on a validation split held out from the training tasks.

	ALFWorld	WebShop	ScienceWorld
Stall window k	5	4	8
Budget threshold θ	11	5	11
Minimum failed attempts	2	1	2

Example pages. Figure 5 shows one harness-equipped page from each environment, with long parts shortened.

ALFWorld (history omitted)

```
Task: Put two remotes on an ottoman.
Observation: Nothing happens.

You are at sofa 1. You are holding remotecontrol 1. 12 of 14 places
visited; 2 unvisited (garbagecan 1, ottoman 1).
This visit to sofa 1: arrived turn 18, 2 commands executed since.
Here, 'take remotecontrol 2 from sofa 1' does nothing (this was
attempt 2). This state will not accept it - select another action
from the displayed list.
Step 21 of 30.
Only 9 steps remain. Commit to finishing the task now.

Admissible commands:
examine remotecontrol 1 / examine sofa 1 / go to armchair 1 / ...
```

WebShop

```
Observation: [item page: The Spicy Beef Backpacking Bundle ...
Price: $14.49 ... [button] Buy Now]

You are at item B0978Q1KK9 / page. You are holding nothing. 3 of 14
places visited; 11 unvisited (home, b0978q1kk9, ..., +6 more).
Not yet chosen here: flavor name.
This visit to item B0978Q1KK9 / page: arrived turn 6, 1 commands
executed since.
Previous visit: arrived turn 3, 1 commands executed, left turn 4.
Nothing new has been observed for 4 steps. You are not making
progress.
Step 7 of 15.

Admissible commands: click[back to search] / click[< prev] / ...
```

ScienceWorld

```
Action: move baby wolf to yellow box in living room
Observation: No known action matches that input.

Location: outside. In hand: orange. 2 of 4 places visited;
2 unvisited (greenhouse, kitchen).
This visit to outside: arrived turn 0, 5 commands executed since.
Here, 'move baby wolf to yellow box in living room' does nothing
(this was attempt 2). This state will not accept it - select
another action from the displayed list.
Step 5 of 15.
```

Figure 5: Example pages that the zero-shot teacher sees with the harness. Long parts of the pages are shortened.

B.3 HARNESS UTILIZATION

Harness utilization counts only the components whose report implies a rule that holds for every task. The visit component and the visited places in the situation component are not included, since the right response to them depends on the task, and neither is the budget counter, which only states the current step. Each of the five remaining components is scored separately, and harness utilization is their unweighted mean (Equation 50), so that components applying to every step do not outweigh those that fire rarely.

For each component $c \in \mathcal{C}$ with a task-agnostic rule $r_c(s) \in \{0, 1\}$, harness utilization averages the fraction of the steps S_c where the component applies at which the rule holds,

$$\mathrm{HU} = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \frac{1}{|S_c|} \sum_{s \in S_c} r_c(s). \tag{50}$$

The rules require the next action to be available, to use only held objects, to avoid purely informational actions after a budget warning, to differ from an action reported as failed, and to reach a new state within k steps after a stall report.

A step s is counted for a component if the page shown before action $s{+}1$ carries that component and its rule can be evaluated. Seen and unseen ALFWorld tasks are pooled. An action outside the ALFWorld command grammar is counted as inconsistent for available actions and is not counted for the other components, so that each error is counted once.

Rules. Available actions applies to every step, and the action is consistent if it is in the admissible commands shown at that step. The situation component applies to every step, and the action is inconsistent if it places, heats, cools, or cleans an object the agent is not holding, slices with a tool the agent is not holding, or takes an object while already holding one. The budget component applies to steps with a budget warning, and the action is inconsistent if it is `look`, `inventory`, `examine`, or `help`. Failure applies to steps with a failure report, and the action is inconsistent if it equals the reported action. Stall applies to steps with a stall report, and the step is consistent if a state not seen earlier in the episode is reached within the next $k = 5$ steps or the episode ends with success. States are identified by the same key the harness uses, namely the location, the held objects, and the admissible commands.

Table 11: Harness utilization and the rate of each component on ALFWorld with the harness.

	Available actions	Situation	Budget	Failure	Stall	HU
Teacher	90.1	97.2	87.8	79.4	16.9	74.3
Student	73.7	96.2	79.1	61.2	18.1	65.7
OPD	87.8	97.0	88.4	52.9	17.6	68.8
SAGE-OPD	88.0	97.9	83.7	52.3	11.0	66.6
Guided-OPD	82.5	94.1	85.7	28.6	5.3	59.2
SOPD	84.8	96.5	92.3	67.3	24.7	73.1
HAD	**93.3**	**99.0**	87.7	**91.1**	**33.8**	**81.0**

B.4 Implementation details

Common implementation. All baselines and HAD run in one synchronous on-policy loop. In each round, the current student rolls out 16 tasks at temperature 1.0 with the harness on the page, and the teacher reads the same harness-equipped page. Collected steps are kept for two policy versions and each is used once. All methods use full fine-tuning with the settings in Table 12. For methods based on OPD, the KL divergence is computed over the teacher's top 64 tokens at each position instead of the full vocabulary. The original papers use a learning rate of 10^{-6}, which did not train the students in our setting, so we use 10^{-5} for all methods.

Teacher sampling, truncation, and validity check. Following prior on-policy distillation work for multi-turn agents (Li et al., 2026; Sun et al., 2026), student rollouts are sampled at temperature 1.0 and all evaluations use temperature 0.4. The teacher samples its responses with and without the harness independently at the same temperature 1.0, so that the two teacher actions are draws from the action laws P_+ and P_- analyzed in Appendix A. Since both teacher responses are sampled, the rates in Appendix D.4 at which the two teacher actions differ include differences due to sampling as well as those due to the harness. For the methods based on OPD, the KL divergence is computed over the teacher's top 64 tokens at each position, with both distributions renormalized over these tokens. The validity checker of HAD applies the same checks in all three environments, rejecting a positive action that is not among the admissible commands, that uses an object the agent does not hold, or that repeats an action reported as having had no effect at the same state.

Baseline implementations. For OPD, the student is trained with the token-level reverse KL divergence to the teacher on its own sampled responses, the on-policy case of GKD (Agarwal et al., 2024). For SAGE-OPD, following the original paper (Zhou et al., 2026), each student turn is weighted by the teacher's judgment of whether it needs no, weak, or strong correction, multiplied by the teacher's confidence over the turn, and the weights are normalized in each update. For Guided-OPD, following the original paper (Li et al., 2026), the teacher takes over each turn with a probability that decays

Instructions shared by all prompts

```
<system>You are an agent acting in a text-based ALFWorld household environment. Your goal is to complete
the household task by choosing one environment action at a time.

On every turn, respond with exactly two lines and nothing else:

Thought: <brief reasoning about the next step>
Action: <one command>

Choose the Action from the admissible commands provided for the current state. Copy the command exactly,
including object and receptacle names and their numeric identifiers. Issue exactly one action.

If the environment returns "Nothing happens.", the previous action had no effect.

Use the current observation and the interaction history to decide what to do next. Do not add any text
before the Thought line or after the Action line.
```

Figure 6: Instructions elided from the per-case figures. They are identical across all ALFWorld cases.

from 1 to 0 with a cosine schedule over the first 80% of updates, and student turns are trained with the OPD loss. Teacher turns are trained with cross-entropy on the teacher's decoded tokens instead of the forward KL divergence of the original paper. Finally, for SOPD, following the original paper (Sun et al., 2026), the teacher writes one full step at every state the student visits, and the student is trained with cross-entropy on these steps, averaged over the tokens of each step and then over steps.

Table 12: Training hyperparameters, shared by all methods unless noted.

	ALFWorld	WebShop	ScienceWorld
Optimizer	AdamW, weight decay 0.01, gradient clipping 1.0		
Learning rate	10^{-5}	10^{-5}	10^{-5}
Schedule	cosine to 0, 15 warmup updates		
Effective batch size	60	60	60
Updates	250	150	150
Tasks per round	16	16	16
Staleness	2	2	2
Rollout temperature	1.0	1.0	1.0
Evaluation temperature	0.4	0.4	0.4
HAD β	0.5	0.5	0.5
HAD ρ	0.5	0.5	0.5

C Prompts and Examples

We show the prompts and replies of each method for one success case and one failure case on ALFWorld. Each prompt is shown as recorded, with the role markers `<system>`, `<user>`, and `<assistant>`, and the server adds the chat template tokens of each model. The instructions are the same in all prompts, so each figure replaces them with `[... instructions elided ...]`, and Figure 6 shows them in full. Each figure keeps only the last three turns and shows the admissible commands only in the last turn. The harness lines appear in the last user turn, between the observation and the admissible commands. The task line is the first human annotation of the trial, while the templated goal appears once in the first observation of each episode.

D Further Experiments

D.1 Analysis: turns

Table 13 reports the mean number of turns over successful ALFWorld episodes, evaluated with and without the harness. With the harness, HAD takes a similar number of turns to the other methods while solving the most tasks. Adding the harness increases the number of turns for most models, including the teacher and the untrained student.

Full prompt (HAD, turn 22)

```
<system>[... instructions elided ...]

[... earlier turns elided ...]
<user>Task: Put two remotes on an ottoman.
Observation: On the diningtable 1, you see a book 1, a
bowl 3, a bowl 2, a bowl 1, a newspaper 1, a plate 1, a
tissuebox 1, and a vase 2.

[... admissible commands elided ...]
<assistant>Thought: There are no remotes on the dining
table. I should check the remaining unvisited locations
like the garbagecan or ottoman.
Action: go to ottoman 1
<user>Task: Put two remotes on an ottoman.
Observation: On the ottoman 1, you see a creditcard 1.

[... admissible commands elided ...]
<assistant>Thought: I need to place the first remote on
the ottoman. I should put it there now.
Action: put remotecontrol 1 in/on ottoman 1
<user>Task: Put two remotes on an ottoman.
Observation: You put the remotecontrol 1 in/on the
ottoman 1.

[... admissible commands elided ...]
<assistant>Thought: I need to find the second remote
control. I should check the remaining unvisited locations
 like the garbagecan or sofa.
Action: go to garbagecan 1
<user>Task: Put two remotes on an ottoman.
Observation: On the garbagecan 1, you see nothing.

You are at garbagecan 1. You are holding nothing. 13 of
14 places visited; 1 unvisited (sofa 1).
This visit to garbagecan 1: arrived turn 20, 0 commands
executed since.
Step 21 of 30.
You have 9 steps remaining. Commit to completing the task
 now.

Admissible commands:
examine garbagecan 1
go to armchair 1
go to coffeetable 1
go to diningtable 1
go to drawer 1
go to drawer 2
go to drawer 3
go to drawer 4
go to drawer 5
go to drawer 6
go to drawer 7
go to ottoman 1
go to sidetable 1
go to sofa 1
help
inventory
look
```

Templated goal

```
find two remotecontrol and put them in
ottoman.
```

HAD, turns 22 and 23 (succeeds at turn 25)

```
Thought: I need to check the sofa next as it
 is the last unvisited location. I should go
 to the sofa to find the second remote.
Action: go to sofa 1
-> You arrive at sofa 1. On the sofa 1, you
see a creditcard 3, a keychain 2, a laptop
3, a laptop 2, a newspaper 2, and a
remotecontrol 2.
Thought: I found the second remote control
on the sofa. I should take it and place it
on the ottoman.
Action: take remotecontrol 2 from sofa 1
-> You pick up the remotecontrol 2 from the
sofa 1.
```

OPD, turns 26 and 27 (fails at turn 30)

```
Thought: I have checked all locations and
found no remote controls. I should check the
 ottoman again to place the first remote
control.
Action: go to ottoman 1
-> You arrive at ottoman 1. On the ottoman
1, you see a creditcard 1, and a
remotecontrol 2.
Thought: I need to place the first remote
control on the ottoman. I should put
remotecontrol 1 in the ottoman.
Action: put remotecontrol 1 in/on ottoman 1
-> Nothing happens.
```

SOPD, turn 27 (fails at turn 30)

```
Thought: I have already placed one remote on
 the ottoman. I need to find the second
remote and place it there.
Action: go to sidetable 1
-> You arrive at sidetable 1. On the
sidetable 1, you see a book 2, a houseplant
1, and a watch 1.
```

Figure 7: ALFWorld success case, seen task `trial_T20190906_182952_858723`. HAD follows the harness report that the sofa is the only unvisited place and finds the second remote. The baselines had passed the sofa earlier without taking the remote, and their replies are shown at the corresponding point near the end of their episodes. OPD tries to place a remote it is not holding, and SOPD returns to a place it has already searched.

Full prompt (HAD, turn 28)

```
<system>[... instructions elided ...]

[... earlier turns elided ...]
<user>Task: Move a heated mug to the base of the coffee
maker.
Observation: On the coffeemachine 1, you see a mug 1.

[... admissible commands elided ...]
<assistant>Thought: The task is complete as the mug is
now in the coffee machine.
Action: inventory
<user>Task: Move a heated mug to the base of the coffee
maker.
Observation: You are not carrying anything.

[... admissible commands elided ...]
<assistant>Thought: I need to confirm that the mug is now
 in the coffee machine and that the task is complete.
Action: examine coffeemachine 1
<user>Task: Move a heated mug to the base of the coffee
maker.
Observation: On the coffeemachine 1, you see a mug 1.

[... admissible commands elided ...]
<assistant>Thought: The task is complete as the mug is
now in the coffee machine.
Action: go to countertop 1
<user>Task: Move a heated mug to the base of the coffee
maker.
Observation: Nothing happens.

Location: coffeemachine 1. In hand: nothing. 12 of 25
places visited; 13 unvisited (countertop 2, countertop 3,
 garbagecan 1, microwave 1, shelf 1, +8 more).
This visit to coffeemachine 1: arrived turn 20, 5
commands executed since.
Nothing new has been observed for 5 steps. You are not
making progress.
Step 27 of 30.
You have 3 steps remaining. Commit to completing the task
 now.

Admissible commands:
examine coffeemachine 1
examine countertop 1
go to cabinet 1
go to cabinet 2
go to cabinet 3
go to cabinet 4
go to cabinet 5
go to cabinet 6
go to countertop 2
go to countertop 3
go to drawer 1
go to drawer 2
go to drawer 3
go to fridge 1
go to garbagecan 1
go to microwave 1
go to shelf 1
go to shelf 2
go to shelf 3
go to sinkbasin 1
go to stoveburner 1
go to stoveburner 2
go to stoveburner 3
go to stoveburner 4
go to toaster 1
help
inventory
look
take mug 1 from coffeemachine 1
take pan 1 from countertop 1
take spoon 1 from countertop 1
```

Templated goal

```
put a hot mug in coffeemachine.
```

HAD, turn 28 (fails at turn 30)

```
Thought: I need to confirm that the mug is
now in the coffee machine and that the task
is complete.
Action: inventory
-> You are not carrying anything.
```

OPD, turn 28 (succeeds at turn 28)

```
Thought: I need to place the heated mug in
the coffee machine.
Action: put mug 2 in/on coffeemachine 1
-> You put the mug 2 in/on the coffeemachine
 1.
```

Figure 8: ALFWorld failure case, unseen task `trial_T20190907_171933_349922`. Earlier in the episode, HAD placed an unheated mug on the coffee machine and concluded that the task was done. The harness reports a lack of progress, but HAD repeats `inventory`, one of the actions it has been cycling through, and fails at the step limit. OPD succeeds, but it is in a different state at this turn, holding a heated mug, so its reply does not answer the same page.

Table 13: Mean number of turns over successful ALFWorld episodes.

	With harness	Without harness
Teacher	14.10	12.99
Student	13.86	9.77
OPD	13.16	13.45
SAGE-OPD	14.71	12.85
Guided-OPD	14.14	12.23
SOPD	15.15	12.31
HAD	14.83	13.14

D.2 Generalization to other model architecture

Table 14 reports the full results with Gemma models, including the number of turns and harness utilization. With the harness, HAD achieves the highest success rate on both splits and the highest harness utilization at 77.9%, above even the 75.9% of the teacher. Every OPD baseline instead ends below the 71.4% of the untrained student in harness utilization, so adding the harness to standard distillation does not make the Gemma student use it better. Without the harness, HAD loses more than the baselines on unseen tasks, where SOPD reaches 29.9% against 23.9% for HAD, which is consistent with HAD relying on the harness that stays in place at deployment. HAD also takes more turns in successful episodes than the other methods, while solving the most tasks.

Table 14: Full results on ALFWorld with Gemma models (E2B ← 12B). SR cells report performance **with harness** / without harness. Turns (mean over successful episodes) and harness utilization (HU) are measured with the harness.

Method	Unseen SR	Seen SR	Turns	HU
Teacher	51.49 / 43.28	49.29 / 42.14	13.91 / 13.93	75.92
Student	14.93 / 8.21	18.57 / 15.71	12.26 / 12.92	71.38
OPD	26.87 / 18.66	31.43 / 29.29	**13.26** / 12.97	68.76
SAGE-OPD	24.63 / 20.90	28.57 / 27.14	14.30 / 13.79	67.35
Guided-OPD	29.10 / 18.66	32.86 / 24.29	14.84 / 11.58	64.05
SOPD	32.09 / 29.85	34.29 / 25.71	13.85 / 12.33	69.04
HAD (ours)	**37.31** / 23.88	**36.43** / 30.00	15.42 / 12.99	**77.91**

D.3 Additional ablations

We report two further ablations on ALFWorld, with all other settings fixed. Table 15 compares HAD with a full-response contrast and with the teacher as a reference model, whose definitions are given in Appendices A.2 and A.4 and whose results are discussed in Section 4.5. Table 16 varies the preference weight ρ from 0.25 to 2.0, and the success rate is highest at $\rho = 0.5$, the value used in all main experiments.

Table 15: Design choices of the preference term on ALFWorld, with the harness.

Variant	Unseen SR	Seen SR
Full-response contrast	59.7	47.1
Teacher as reference	56.0	46.4
HAD (ours)	**63.4**	**51.4**

Table 16: Preference weight ρ on ALFWorld, with the harness.

ρ	Unseen SR	Seen SR
0.25	53.7	47.9
0.5	**63.4**	**51.4**
1.0	49.3	50.7
2.0	51.5	49.3

D.4 Where the Harness Changes the Teacher's Action

We measure how often the teacher chooses a different action with and without the harness at the states visited during HAD training. As Figure 9 shows, the two teacher actions differ at 15.5% of the states on ALFWorld, 25.6% on WebShop, and 19.3% on ScienceWorld. These states are spread over the whole episode rather than concentrated at its start.

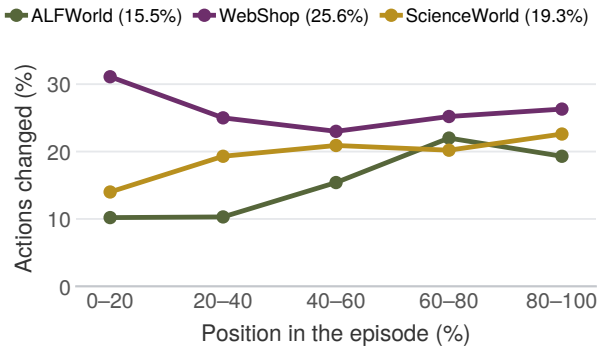


Figure 9: How often the teacher actions with and without the harness differ at each position in the episode.