SPECULATIVE ROLLBACK CORRECTION: COUNTERFACTUAL CREDIT ASSIGNMENT FOR GUI AGENT IMITATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Interactive imitation learning supervises agents on learner-induced states, yet conventional methods typically identify errors via disagreement with a reference policy. In multi-solution GUI tasks, this proxy frequently conflates genuinely harmful actions with valid alternative trajectories. To resolve this ambiguity, we define errors through task recoverability: an action is harmful to the extent that it reduces the maximum probability of eventual success achievable by any continuation policy. Because recoverability optimizes over policy space, it reflects an intrinsic property of the task and environment rather than the choices of a specific teacher. We show that the resulting recoverability process admits an exact Doob decomposition, whose predictable compensator yields per-step harm and provides a principled mechanism for localizing feasibility-reducing actions. We instantiate this theoretical framework as Speculative Rollback Correction (SRC). In SRC, the student agent executes a short speculative trajectory segment, after which the teacher localizes the earliest action causing a drop in recoverability. A reset-and-replay mechanism then restores the corresponding state for targeted correction, followed by a hard verifier that filters candidate continuations before updating a standard next-action policy. Across WebArena-Infinity, WebArena-Lite, and OSWorld, augmenting Expert SFT with SRC data improves teacher-free deployment success rates by 9.7, 3.6, and 12.9 percentage points, respectively. On the two web benchmarks, SRC outperforms OEC-style training by 14.6 and 9.7 points. It also outperforms LEAP-style training by 3.2 and 4.8 points while using 56–58% fewer teacher queries. Controlled experiments further confirm the efficacy of corrective labels and teacher-guided localization. At test time, the deployed policy operates autonomously without requiring teacher queries, verifiers, or environment resets, establishing recoverability-guided correction as a practical data-collection principle for resettable GUI environments.

1 INTRODUCTION

In imitation learning, policy evaluation occurs on execution states induced by the agent's own past decisions—an exposure to covariate shift that motivated DAgger (Ross et al., 2011). This challenge is severely amplified in long-horizon GUI agents, where accumulating context causes a single early mistake to corrupt all subsequent observations (Daumé III et al., 2009; Chang et al., 2015; Lamb et al., 2016; Laskey et al., 2017). Interactive imitation mitigates this by querying the teacher on learner-induced states: LEAP provides privileged feedback during on-policy rollouts (Choudhury & Sodhi, 2025), while On-Policy Expert Corrections intervenes mid-rollout to resume expert control (Lauffer et al., 2025). Although both approaches establish a sound premise, they leave unresolved the foundational challenge this paper addresses: localizing the precise point of failure within a learner's rollout and constructing the corresponding corrective supervision.

The limit of deviation-based error. Existing interactive imitation methods rely on a central premise: **an error is defined as a deviation from a reference policy**. This premise trivializes localization—identifying an error reduces to finding where the student strays. DAgger evaluates every step against this baseline, post-hoc correction applies it retrospectively, and random takeover

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.

bypasses localization entirely. However, this formulation breaks in GUI environments due to their inherently *multi-solution* nature: retrieving a message can be accomplished by searching or scrolling; filtering a list can be executed via UI controls, keyboard shortcuts, or reordered subgoals (Figure 1). Consequently, penalizing a student action simply because it diverges from a single demonstrated path incorrectly punishes alternative valid trajectories and collapses the policy's expressive capacity. Conversely, uncritically accepting all deviations introduces loops, redundant exploration, and inefficient detours. Resolving this tension requires fundamentally redefining error rather than relaxing existing heuristics.

Defining error by recoverability. We ground the definition of error in an action's marginal cost to task completion rather than its resemblance to a reference trajectory. Let h denote the interaction history, a a candidate action, π a continuation policy, and $P(\cdot \mid h, a)$ the environment's next-observation distribution. For $o' \sim P(\cdot \mid h, a)$, let (h, a, o') denote the resulting extended history. We define the recoverability $V^*(h)$ and action harm $\Delta(h, a)$ as

$$V^*(h) = \sup_{\pi} \Pr\big(\text{success} \mid h, \pi\big), \qquad \Delta(h, a) = V^*(h) - \mathbb{E}_{o' \sim P(\cdot|h,a)}\big[V^*\big((h, a, o')\big)\big].$$

Because V^* denotes the supremum over all possible policies, it reflects an intrinsic property of the task and environment dynamics rather than any individual teacher. A teacher serves merely as a lower-bound estimator of V^*. Consequently, this formulation ensures objective error evaluation: unlike teacher-dependent value functions where the same student action might be misclassified depending on teacher strength, Δ provides a consistent, objective measure.

Theoretical guarantees. The harm field Δ systematically governs both error localization and the acceptance of multi-solution trajectories. Crucially, the total loss of task feasibility decomposes *exactly* into step-wise harms. By the uniqueness of the Doob decomposition, Δ represents the sole per-step credit attribution consistent with the final task outcome (Lemma 1), elevating credit assignment from a heuristic approximation to a well-posed estimation problem. Where $\Delta > 0$, the action incurs irreversible harm and must be localized and repaired; where multiple candidate actions yield $\Delta \approx 0$ *simultaneously*, divergence from a single reference path does not constitute an error. Our data collector thus retains such valid alternative trajectories whenever validated by a hard verifier, without making unrealistic assumptions about recovering the entire solution space.

Operationalizing Δ via resettability. In text-based reasoning tasks, estimating the value of a prefix by rolling forward is straightforward because prefixes are token sequences that can be duplicated freely. For interactive agents, however, a prefix corresponds to a physical or system *environment state*. Sampling a counterfactual continuation thus requires restoring the environment to that exact state—an operational requirement that previously led researchers to rely on short-window inspection instead of process-level value estimation. Environment resettability eliminates this obstacle: during collection, the framework rolls back to a candidate error state, replays alternative actions, and leverages a hard verifier to determine whether the task is recovered. Resettability is therefore the core enabling condition of our theoretical formulation, required strictly during data collection; the deployed policy operates without teacher guidance or environment resets.

Speculative Rollback Correction. We operationalize this formulation into a unified three-step pipeline (Figure 2) derived directly from Δ. The student executes an unmonitored speculative action segment; the teacher inspects the segment as a whole to localize the earliest action incurring harm ($\Delta > 0$); the environment then rolls back to that state for teacher correction before resuming rollout, with the resulting trajectory validated by a hard verifier. The collector subsequently trains a standard next-action policy on localized corrections and verified success paths. This speculative paradigm mirrors speculative decoding (Leviathan et al., 2023; Chen et al., 2023): a low-cost draft is generated autonomously, validated in a single pass, accepted upon verification, or rolled back upon failure. In practice, the review horizon is fixed at $K = 3$ steps, balancing teacher query budget against localization difficulty and discarded student progress.

Contributions.

1. **Theoretical Reframing.** We recast interactive imitation for GUI agents as a credit assignment problem in multi-solution environments. We formalize the flaws of the traditional

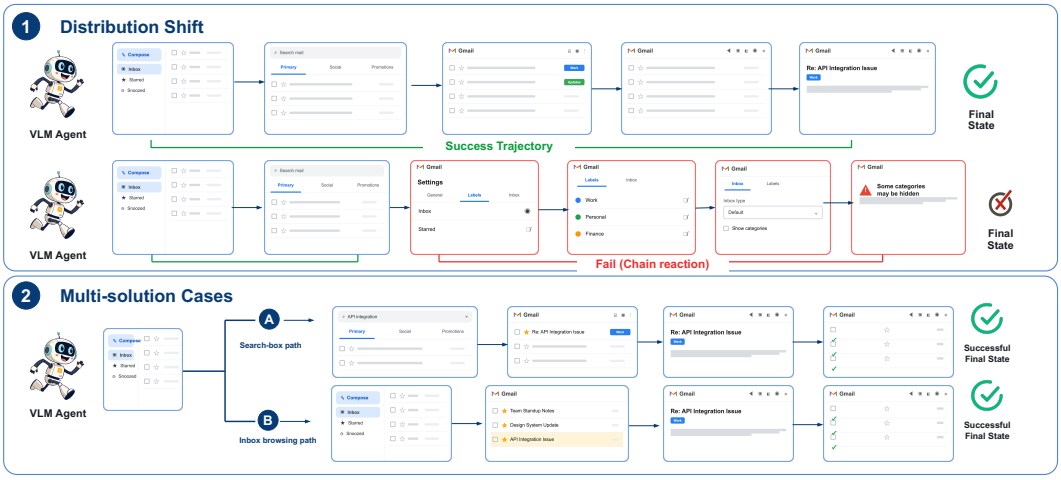

Figure 1: Two facts that a correction mechanism must respect. Top: in long-horizon GUI tasks one early mistake regenerates every later observation, so a correction applied after the fact is assigned to the wrong place. Bottom: the same instruction admits several verifier-passing routes — search-box retrieval and inbox browsing here — so a deviation from a demonstrated path is not by itself an error. Defining error by recoverability rather than by resemblance to a reference resolves both at once.

deviation-based error assumption and replace it with a recoverability-based formulation under which credit decomposes uniquely and exactly.

2. **Algorithmic Framework.** We introduce Speculative Rollback Correction, a practical data collection pipeline that instantiates this theory through teacher-guided localization, rollback correction, verifier filtering, and next-action policy supervision under a fixed review schedule ($K = 3$).
3. **Comprehensive Evidence.** Through extensive experiments, we demonstrate that Speculative Rollback Correction consistently outperforms standard SFT, OEC-, and LEAP-style baselines under explicit query budgets. We further validate the efficacy of localized rollback against random baselines, quantify reset-and-replay fidelity, and show that the synthesized data successfully trains an autonomous, teacher-free, and reset-free deployment policy.

2 Related Work

Interactive imitation and credit assignment. Behavior cloning is simple to scale but suffers from covariate shift when a learned policy visits states absent from expert demonstrations. DAgger addresses this by querying an expert on learner-induced states (Ross et al., 2011), while AggreVaTe replaces action matching with a reference policy's cost-to-go (Ross & Bagnell, 2014); learning-to-search reductions similarly study how roll-ins and reference costs assign credit in sequential prediction (Daumé III et al., 2009; Chang et al., 2015). Recent agent methods bring this principle to language and tool use: LEAP corrects on-policy errors using privileged feedback (Choudhury & Sodhi, 2025), and OEC switches from student to expert partway through a rollout (Lauffer et al., 2025). These methods establish the importance of supervising states the learner actually visits. SRC changes the criterion applied at those states: it targets task-defined recoverability rather than agreement or cost-to-go under a particular reference policy, and uses reset-and-replay to obtain counterfactual evidence about which action harmed that recoverability.

Counterfactual prefix evaluation and intervention. Reasoning methods estimate the value of a partial solution by sampling continuations from a token prefix, ranging from process reward models with step supervision (Lightman et al., 2024) to automatically labeled rollout-based values (Wang et al., 2024; Luo et al., 2024). Interactive agents also revisit earlier states in tree search (Koh et al., 2024; Zhou et al., 2024; Putta et al., 2024) and in Go-Explore, which restores archived cells before

exploring onward (Ecoffet et al., 2021). The mechanism overlaps with ours, but the purpose differs: search restores a promising state to improve the current solution, whereas SRC restores a suspected error state to determine what training supervision it should receive. Our evaluated collector uses a fixed review horizon and measures its success–query trade-off; adaptive review timing remains outside the present method.

Multimodal behavior cloning. A separate line of work addresses a complementary question: how a policy represents multiple actions supported by demonstrations. Implicit Behavioral Cloning models set-valued action mappings with energy-based policies (Florence et al., 2022); Behavior Transformer uses action discretization and residual prediction to fit multiple modes (Shafiullah et al., 2022); and Diffusion Policy models multimodal action sequences with a conditional diffusion process (Chi et al., 2023). SRC instead changes the supervision criterion during interactive collection: on a learner-induced state, it asks whether a non-reference action reduced task recoverability and where the harm began. Thus, multimodal policy models represent alternatives already supported by data, whereas SRC decides which learner deviations warrant correction; the two are compatible.

3 Problem Setup and Recoverability

We consider a set of finite-horizon GUI-agent tasks $\mathcal{X}$. For a task $x \in \mathcal{X}$, the agent receives observation $o_t \in \mathcal{O}_x$, selects an action $a_t \in \mathcal{A}$, and the environment samples the next observation from $P_x(\cdot \mid h_t, a_t)$, where $h_t = (o_1, a_1, \ldots, a_{t-1}, o_t)$ is the interaction history. A horizon-H trajectory is $\tau = (o_1, a_1, \ldots, o_H, a_H, o_{H+1})$, and a hard task verifier returns $\mathcal{V}(x, \tau) \in \{0, 1\}$. We assume a finite action set for the analysis and let Π denote all history-dependent, possibly randomized policies. The learner is a student policy $\pi_\theta(a_t \mid h_t, x)$; a stronger teacher policy $T \in \Pi$ is available only during data collection, and the deployed student acts alone.

3.1 Recoverability as the error criterion

As previewed in the Introduction, the recoverability of a history is the highest success probability achievable by any continuation policy. Its one-step action value and the resulting action harm are

$$V^*(h) = \sup_{\pi \in \Pi} \mathbb{E}_\pi[\mathcal{V}(x, \tau) \mid h], \qquad Q^*(h, a) = \mathbb{E}_{o' \sim P_x(\cdot \mid h, a)}\left[V^*\big((h, a, o')\big)\right]. \tag{1}$$

$$\boxed{\Delta(h, a) = V^*(h) - Q^*(h, a).} \tag{2}$$

At a terminal history h_{H+1}, we set $V^*(h_{H+1}) = \mathcal{V}(x, \tau)$. A particular teacher does not define this value: its continuation value $\widehat{V}_T(h) := \mathbb{E}_{\tau \sim T(\cdot \mid h)}[\mathcal{V}(x, \tau)]$ satisfies $\widehat{V}_T(h) \le V^*(h)$.

Finite-horizon dynamic programming applied to Equation (1) gives $V^*(h) = \max_a Q^*(h, a)$. Therefore,

$$\Delta(h, a) \ge 0, \qquad \Delta(h, a) = 0 \iff a \in \mathcal{A}^*(h) := \arg\max_{a'} Q^*(h, a'). \tag{3}$$

The optimal-action set in Equation (3) may contain several actions. Disagreement with one reference action is therefore insufficient to establish harm: a non-reference action is an error only when it reduces recoverability.

3.2 Exact credit decomposition

Lemma 1 (Exact additive credit decomposition)**.** *Fix an initial history h_1 and any policy $\pi \in \Pi$. Let $h_1, a_1, \ldots, h_H, a_H, h_{H+1}$ be the induced process, with $\mathcal{G}_j = \sigma(h_j, a_j)$ for $j \le H$ and $\mathcal{G}_{H+1} = \sigma(h_{H+1})$. Set $Y_j = V^*(h_j)$ for $j \le H + 1$ and $\Delta_j = \Delta(h_j, a_j)$ for $j \le H$. Then $M_j = Y_j + \sum_{i<j} \Delta_i$ is a martingale, while Y is a supermartingale with predictable compensator $\sum_{i<j} \Delta_i$. In particular,*

$$\mathbb{E}_\pi[\mathcal{V}(x, \tau)] = V^*(h_1) - \mathbb{E}_\pi\left[\sum_{j=1}^{H} \Delta(h_j, a_j)\right]. \tag{4}$$

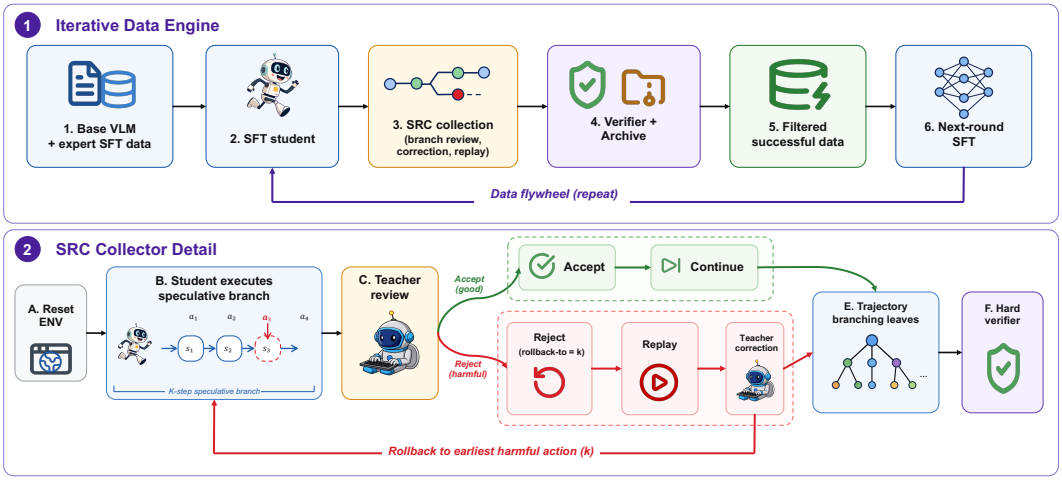

Figure 2: Speculative Rollback Correction data engine and collector loop. Top: iterative data generation and SFT—a student collects SRC trajectories, verifier and quality filtering retain successful data, and the next-round student trains on the filtered mixture. Bottom: one SRC collection step—the teacher reviews a speculative segment, estimates the earliest harmful action, and proposes a corrective action only after rollback restores the corresponding observation.

The proof is a one-step conditioning argument and appears in Appendix A. Equation (4) identifies the expected gap between initial recoverability and terminal success with accumulated action harm; the Doob decomposition makes this predictable attribution unique.

From criterion to implementation. The formalism identifies the target of corrective supervision but does not provide an estimator for V^*, Q^*, or Δ. Section 4 describes how SRC turns this criterion into a practical collection procedure.

4 Speculative Rollback Correction

Figure 2 outlines Speculative Rollback Correction (SRC). To operationalize the unobserved harm objective Δ in Equation (2), SRC leverages the teacher to infer harm onset and location, while using environmental verifiers to filter the synthesized dataset. The engine functions in three integrated phases: (1) **Review and localize:** the teacher evaluates a speculative student execution segment to pinpoint the earliest action index degrading task recoverability; (2) **Restore and correct:** reset-and-replay restores the environment to the localized state, where a teacher-provided corrective action is injected before resuming rollout; and (3) **Filter and train:** a hard verifier filters for successful trajectory continuations, yielding localized corrections and verified success paths to supervise a standard next-action policy.

4.1 Speculative Review and Rollback Correction

In SRC, the student policy operates via speculative execution over unmonitored action segments. Given history h_t and a review horizon $K \in \mathbb{N}$, the student autonomously generates a K-step trajectory segment:

$$B_t = \big((o_t, h_t, a_t), \ldots, (o_{t+K-1}, h_{t+K-1}, a_{t+K-1})\big).$$

The teacher then evaluates B_t as a unified block, issuing an ACCEPT decision or identifying the earliest harmful step index $\hat{j}$. An ACCEPT signal indicates that the segment maintains task recoverability, even if the student follows an alternative valid route. Rejection is strictly reserved for substantive failures that degrade task feasibility, such as invalid state transitions, execution loops, or goal drift.

Upon rejection at index $\hat{j}$, SRC retains the valid prefix prior to $\hat{j}$, resets the environment, replays execution up to $h_{\hat{j}}$, and queries the teacher for a single corrective action before resuming student rollout. This decoupled design minimizes teacher overhead: the first query localizes the error, while the second provides the targeted repair label post-rollback.

Optimal Rollback Location. The choice of prefix retention and rollback location directly emerges from the properties of the harm field Δ.

Proposition 1 (Minimal Sufficient Rollback)**.** *Assume deterministic transitions and consider a reviewed window with action indices $\{\sigma, \dots, m\}$. Let $\delta \geq 0$ be the harm threshold, let $r \in \{\sigma, \dots, m+1\}$ denote a candidate rollback index, and let $j^\star$ be the earliest index in the window with $\Delta(h_j, a_j) > \delta$. Then:*

$$\begin{aligned} 0 \leq V^*(h_r) - V^*(h_{j^\star}) &\leq \delta\,(j^\star - r), \quad && r \leq j^\star, \\ V^*(h_{j^\star}) - V^*(h_r) &\geq \Delta(h_{j^\star}, a_{j^\star}) > \delta, && r > j^\star. \end{aligned}$$

Consequently, let κ denote the marginal value of a discarded student step. If $\kappa > \delta$ and the mean harm on $[j^\star, r)$ exceeds κ for every $r > j^\star$, then $r = j^\star$ uniquely maximizes $V^*(h_r) - \kappa\,(m+1-r)$.

Proposition 1 establishes that rolling back earlier than $j^\star$ yields diminishing returns bounded by δ per discarded step, thereby calibrating the harm tolerance δ directly to the marginal value of student progress. The proposition defines the theoretical optimum $j^\star$; our practical implementation substitutes $j^\star$ with the teacher estimate $\hat{j}$, whose empirical effectiveness is benchmarked against random rollback baselines in Section 5.3. Proofs are provided in Appendix A.

4.2 Trajectory Curation and Policy Training

Review Horizon Schedule. In implementation, the speculative engine evaluates student progress over a fixed horizon of K steps. The choice of K governs an explicit operational trade-off: smaller horizons increase teacher query frequency, whereas larger horizons exacerbate progress loss upon rejection and complicate downstream error localization. We set $K = 3$ as the standard schedule to balance query efficiency against localization accuracy across benchmarks, with comprehensive horizon ablations ($K \in \{1, 3, 5, 7\}$) detailed in Section 5.4.

Verifier-Constrained Archiving. Beyond teacher-corrected mainlines, speculative execution can yield alternative student continuations at near-optimal decision points. Rather than discarding these valid paths, SRC retains them as distinct logical leaves whenever the fork budget permits. For task x, let $\mathcal{C}_x(\pi_\theta, T)$ be the candidate trajectories generated by the mixed student–teacher collector and let $\mathcal{A}_x$ be the retained archive. We summarize the rule-based filters by $q(\tau) = (|\tau|, \mathrm{repeat}(\tau), \mathrm{interventions}(\tau))$, where $|\tau|$ is the number of action steps, $\mathrm{repeat}(\tau)$ is the maximum number of consecutive identical actions, and $\mathrm{interventions}(\tau)$ counts teacher corrections. We set $q_{\max} = (60, 4, 6)$, with the inequality below interpreted componentwise. The framework admits candidates only if they pass both these filters and the hard task verifier (Appendix B):

$$\mathcal{A}_x \subseteq \{\tau \in \mathcal{C}_x(\pi_\theta, T) : \mathcal{V}(x, \tau) = 1,\ q(\tau) \leq q_{\max}\}, \tag{5}$$

After applying Equation (5), we index each surviving trajectory by the coarse archive descriptor $b(\tau)$, comprising buckets for path length, the most frequent action type, and the teacher-intervention count. This is a Quality-Diversity-inspired curation scheme (Mouret & Clune, 2015; Pugh et al., 2016): candidates compete within coarse execution buckets while the archive filters loops and inefficient detours. The buckets are bookkeeping devices rather than semantic strategy labels. Branching functions strictly as a data-collection mechanism and remains fully decoupled from policy inference.

Training Objective. The final training set integrates localized rollback corrections $\mathcal{D}_{\mathrm{corr}}$ and next-action labels extracted from verified success trajectories $\mathcal{D}_{\mathrm{succ}}$. Denoting the extracted action dataset as $\mathcal{D}^{\mathrm{act}}_{\mathrm{succ}} = \{(x, h, a) : (x, \tau) \in \mathcal{D}_{\mathrm{succ}}, (h, a) \in \tau\}$ and the unified mixture as $\mathcal{D}_{\mathrm{sft}} = \mathcal{D}_{\mathrm{corr}} \cup \mathcal{D}^{\mathrm{act}}_{\mathrm{succ}}$, we train the deployment policy with the standard next-action Supervised Fine-Tuning (SFT) objective in Equation (6):

$$\mathcal{L}(\theta) = \mathbb{E}_{(x,h,a)\sim\mathcal{D}_{\mathrm{sft}}}\big[-\log \pi_\theta(a \mid h, x)\big]. \tag{6}$$

5 EXPERIMENTS

We evaluate Speculative Rollback Correction through four operational objectives derived from our recoverability-based formulation: **(Q1) End-to-End Performance** compares recoverability-aligned data with standard interactive-imitation baselines under teacher-free deployment. **(Q2) Supervision Quality** measures the training value of localized corrections relative to an equally sized set of accepted student actions. **(Q3) Localization Precision** evaluates error localization via Δ against random rollback and naive post-hoc correction. **(Q4) System Fidelity & Trade-offs** quantifies reset-and-replay accuracy and the effect of the review horizon K on query efficiency. Overall, our experimental design systematically validates the operational predictions derived from our harm formulation, isolating the contributions of precise localization, rollback corrections, and verifier filtering across complex GUI domains.

5.1 SETUP

Benchmarks and Protocols. Main evaluations follow the established protocols of WebArena-Infinity (Zhou, 2026), WebArena-Lite (Liu et al., 2024), and OSWorld (Xie et al., 2024). WebArena-Infinity utilizes its real-task split (1,620 tasks), and WebArena-Lite employs its complete test set (165 tasks). Our OSWorld benchmark consists of 186 tasks across Chrome (46), LibreOffice Calc (47), Impress (47), Writer (23), and OS-level workflows (23).

Controlled supervision experiments use disjoint held-out splits to prevent dataset contamination. The WebArena-Infinity supervision split contains 346 tasks (16 easy, 44 medium, 286 hard), whereas the OSWorld supervision comparison utilizes 125 held-out task templates distinct from the 200 training templates. The rollback-localization study uses the 186-task OSWorld subset. No policy is ever trained on its corresponding evaluation tasks or templates.

Models and Metrics. The default student policy is Qwen3.5-9B, and the local teacher is Kimi K2.5. All entries in Table 1 represent our clean reruns under a uniform 30-step interaction limit. The teacher inspects screenshots and action histories, supplemented by hard-verifier execution feedback for termination checks. Primary evaluation metrics are Success Rate (SR, %) and OSWorld task-progress scores. All reported deployment results are fully autonomous: trained student policies execute at test time without teacher access, verifiers, or environment resets. Statistical significance is assessed via paired McNemar tests on WebArena-Infinity and 20,000 paired bootstrap draws over templates on OSWorld. Appendix B details collection configurations, prompts, and quality filters.

5.2 END-TO-END AUTONOMOUS DEPLOYMENT PERFORMANCE (Q1)

Table 1 presents primary end-to-end deployment results for autonomous policies evaluated without test-time teacher guidance or environment resets. Figure 3 summarizes the mechanism and diagnostic evidence detailed in subsequent sections.

Table 1: Teacher-free deployed-policy performance (Success Rate, %). All values are standardized reruns under a uniform 30-step horizon; no method uses a teacher, verifier, or reset at evaluation time. Parentheses indicate the absolute gain of Expert SFT + SRC over Expert SFT. Bold denotes the best trained student result in each column.

Method	WebArena-Infinity ($N = 1{,}620$)	WebArena-Lite ($N = 165$)	OSWorld subset ($N = 186$)
Base model	16.4	15.2	23.7
Expert SFT	25.3	20.6	27.4
OEC-style random switch	20.4	14.5	21.5
LEAP-style post-hoc correction	31.8	19.4	24.7
Expert SFT + SRC	**35.0** (+9.7)	**24.2** (+3.6)	**40.3** (+12.9)
Teacher Upper Bounds (Zero-Shot Execution)			
Kimi K2.5	39.7	29.1	39.8
Qwen3.5 Plus	43.7	30.3	43.0

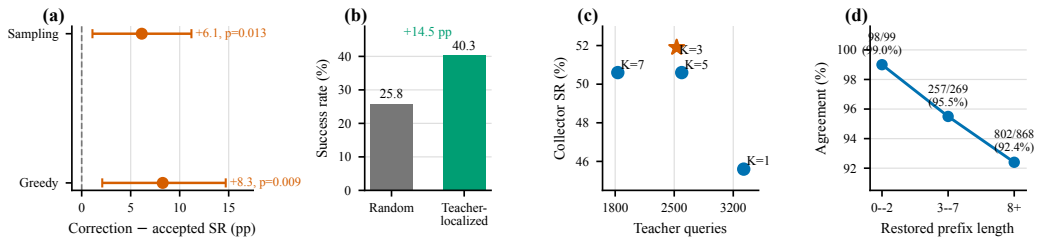

Figure 3: Diagnostic evidence for core SRC mechanisms. (a) Downstream success-rate difference between correction-trained and accepted-action-trained policies on OSWorld; horizontal bars show 95% confidence intervals, and the dashed line marks no difference. (b) Performance impact of teacher-localized versus random rollback positions on OSWorld. (c) Collector success rate versus teacher query budget across review horizons K on WebArena-Infinity. (d) Replay state-agreement fidelity as a function of restored prefix depth.

Deployment Results. Across all three benchmarks, augmenting Expert SFT with SRC data yields substantial gains: $+9.7$ percentage points on WebArena-Infinity (35.0% vs. 25.3%), $+3.6$ points on WebArena-Lite (24.2% vs. 20.6%), and $+12.9$ points on OSWorld (40.3% vs. 27.4%). Crucially, on OSWorld, the SRC-trained 9B student (40.3%) matches the zero-shot performance of its proprietary teacher Kimi K2.5 (39.8%). These empirical improvements confirm that recoverability-aligned training data successfully transfers teacher capability to an autonomous student. To dissect the specific mechanism driving these gains, we isolate supervision density and localization accuracy in the following controls.

The same teacher-free protocol enables a direct comparison with interactive imitation baselines. On WebArena-Infinity, Expert SFT + SRC reaches 35.0%, compared with 20.4% for OEC-style random switching (Lauffer et al., 2025) and 31.8% for LEAP-style post-hoc correction (Choudhury & Sodhi, 2025). On WebArena-Lite, SRC reaches 24.2%, exceeding OEC at 14.5% and LEAP at 19.4%. Thus, SRC outperforms both interactive-imitation baselines on the two web benchmarks under the teacher-free deployment protocol.

5.3 Supervision Quality and Localization Precision (Q2 & Q3)

Training-time teacher efficiency. Although every policy in Table 1 is teacher-free at deployment, OEC, LEAP, and SRC use teacher queries to construct their training data. On WebArena-Infinity, SRC uses 12.01 queries per collection task, compared with 10.09 for OEC and 27.42 for LEAP; on WebArena-Lite, the corresponding costs are 11.90, 10.15, and 28.64. Relative to OEC, SRC therefore obtains 14.6- and 9.7-point deployment gains with a similar training-time query budget. Relative to LEAP, SRC uses 56%–58% fewer teacher queries while achieving 3.2- and 4.8-point deployment gains. Appendix Table 5 separates deployment SR from training-time query cost.

Teacher-assisted collector diagnostic. During SRC data generation, the online teacher-assisted collector itself reaches 42.5% SR on WebArena-Infinity and 36.0% on WebArena-Lite. These values characterize the collection process rather than a deployed policy and are not compared directly with the teacher-free rows in Table 1.

Value of Corrective Labels vs. Accepted Actions (Figure 3(a)). To isolate the information density of localized corrections, we compare training on teacher-corrected steps ($\mathcal{D}_{\text{corr}}$) against training on accepted student steps ($\mathcal{D}_{\text{acc}}$), controlling for dataset size (1,897 records) and template coverage (200 training templates) on OSWorld. As shown in Figure 3(a), localized corrections outperform accepted actions by 6.13 SR points under sampling (95% CI $[1.1, 11.2]$, $p = 0.013$) and by 8.27 points under greedy decoding (95% CI $[2.1, 14.7]$, $p = 0.009$).

Because volume and domain coverage are strictly controlled, this performance gap confirms that localized corrections possess substantially higher signal density. Accepted actions primarily reinforce

steps the student already executes reliably. In contrast, corrections inject targeted supervision at critical state transitions where student choices would otherwise trigger unrecoverable failure.

Efficacy of Teacher Error Localization (Figure 3(b)). We further examine whether teacher-driven error localization ($\hat{j}$) outperforms an unlocalized baseline by comparing SRC against uniform random rollback. Holding all other collection variables constant, teacher-localized rollback achieves 40.3% SR on OSWorld compared to 25.8% for random rollback—a 14.5 percentage point margin (Figure 3(b)).

This gap directly supports the theoretical premise of Proposition 1. Random rollback frequently resets the agent too early (discarding valid progress and inflating context cost) or too late (after irreversible harm has already altered downstream state dynamics). Teacher localization accurately targets the boundary where task recoverability degrades, maximizing the utility of the injected corrective action.

5.4 Operational Trade-offs and Replay Reliability (Q4)

Review Horizon Ablation (Figure 3(c)). We evaluate the operational trade-off of the review horizon K on WebArena-Infinity across three core task domains. As illustrated in Figure 3(c), $K = 3$ achieves the optimal collector success rate (51.9% SR with $2{,}529$ queries), outperforming myopic review at $K = 1$ (45.6% SR, $3{,}324$ queries), $K = 5$ (50.6% SR, $2{,}589$ queries), and $K = 7$ (50.6% SR, $1{,}831$ queries).

These dynamics reflect two competing overheads: **(1) Review Myopia ($K = 1$):** Evaluating every single step forces high review overhead ($2{,}603$ review queries) and lacks sufficient temporal context to distinguish benign execution variation from genuine task harm, leading to premature interventions. **(2) Error Cascade ($K \geq 5$):** Longer horizons reduce review query frequency ($1{,}227$ at $K = 7$) but allow unrecoverable errors to propagate further. This elevates rejection rates (accept rate drops to 20.9%) and forces deeper rollbacks (averaging 3.43 steps), causing collection efficiency to plateau. Thus, $K = 3$ serves as the primary balanced configuration, while $K = 7$ offers a highly query-efficient alternative for budget-constrained collection (Appendix Table 8).

State Replay Fidelity (Figure 3(d)). Counterfactual collection relies on environmental reset-and-replay to faithfully restore target states. We assess state restoration fidelity by comparing application snapshots before rollback with those regenerated via reset-and-replay. Across $1{,}236$ snapshot-verified WebArena-Infinity rollbacks, environment state matching reaches 93.6% under strict comparison and 96.3% after accounting for non-functional UI artifacts ($1{,}190/1{,}236$). On OSWorld, state agreement reaches 97.4% across $6{,}888$ rollbacks.

Figure 3(d) illustrates fidelity as a function of prefix depth: agreement remains exceptionally high at short prefix lengths (99.0% for lengths 0–2) and gradually degrades for longer prefixes (92.4% for lengths ≥ 8). Manual auditing indicates that deep replay failures stem primarily from cumulative UI rendering latency and transient network drops rather than logic breakdown. Overall, these high agreement rates validate environmental reset-and-replay as a robust foundation for counterfactual trajectory collection.

6 Conclusion

We introduced a recoverability-based framework for GUI-agent error modeling and instantiated it as Speculative Rollback Correction (SRC), which localizes harmful actions, rolls back for targeted correction, and filters recovered trajectories with a hard verifier. Across WebArena-Infinity, WebArena-Lite, and OSWorld, adding SRC data to Expert SFT improves teacher-free deployment SR by 9.7, 3.6, and 12.9 percentage points, respectively; controlled comparisons further support the roles of corrective labels and teacher-guided localization. The approach nevertheless depends on resettable training environments, reliable replay, and teacher judgments that approximate recoverability; its guarantees do not directly extend to irreversible or partially observed dynamics, and our experiments do not establish exhaustive coverage of valid solution modes. These requirements are confined to data collection: the deployed policy is a standard next-action model that uses neither teacher queries nor environment resets. Overall, the results support recoverability-guided correction as a practical training-time data-collection principle for GUI agents.

Ethics Statement

All experiments use benchmark-controlled local or self-hosted environments from WebArena-Infinity, WebArena-Lite, and OSWorld; no human subjects were involved and no real user data, credentials, or personally identifying information were collected, stored, or processed. The clinical- and mail-themed applications in WebArena-Infinity contain fabricated records only, and we did not connect any evaluated agent to a live service. The method requires the ability to reset an environment to a previous state, which confines data collection to sandboxed deployments: we make no claim that the collection procedure is appropriate on live systems, where a speculative action cannot be undone and a mistaken action has real consequences. We see no dual-use concern specific to this work beyond those already attached to GUI agents in general.

Reproducibility Statement

The formal claims are stated in Section 3 and Section 4 with their assumptions made explicit, and complete proofs of Lemma 1 and Proposition 1 are given in Appendix A, with their assumptions stated explicitly. Section 5.4 measures replay agreement, an operational requirement of the implemented collector. The collection protocol — review horizon, fork and leaf budgets, intervention cap, trajectory filters, and the final supervised mixture — is specified in Section 5.1 and expanded in Appendix B, and the exact teacher prompts are reproduced in Appendix B.2. Because the collector is stochastic under a fixed configuration, we report repeats with task-level bootstrap confidence intervals rather than single-pass point estimates, and state explicitly wherever only one pass was affordable. An anonymized source-code archive containing the collector, the teacher interface, the prompt templates, and the analysis utilities is included with the supplementary material.

AI Use Statement

We utilized large language models (LLMs) in three capacities during this study. First, LLMs provided writing assistance for drafting and refining prose; all technical content, formal statements, proofs, and citations were independently derived, verified, and finalized by the authors. Second, LLMs offered code assistance for refactoring data collection scripts and analysis pipelines. All code was thoroughly reviewed, executed, and tested by the authors, and all reported numbers derive directly from these script executions. Third, LLMs served as the subject of study, as both teacher and student policies in Section 5.1 are language-model-based. Generative AI was not used to synthesize data, impute experimental results, or review literature without author verification. The authors assume full responsibility for the contents of this work.

References

Kai-Wei Chang, Akshay Krishnamurthy, Alekh Agarwal, Hal Daumé III, and John Langford. Learning to search better than your teacher. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2058–2066, 2015.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. Accelerating large language model decoding with speculative sampling. *arXiv preprint arXiv:2302.01318*, 2023. URL https://arxiv.org/abs/2302.01318.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin C. M. Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, 2023. doi: 10.15607/RSS.2023.XIX.026. URL https://roboticsproceedings.org/rss19/p026.html.

Sanjiban Choudhury and Paloma Sodhi. Better than your teacher: LLM agents that learn from privileged AI feedback. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=st7XqFgbAH.

Hal Daumé III, John Langford, and Daniel Marcu. Search-based structured prediction. *Machine Learning*, 75(3):297–325, 2009.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. First return, then explore. *Nature*, 590(7847):580–586, 2021. doi: 10.1038/s41586-020-03157-9.

Pete Florence, Corey Lynch, Andy Zeng, Oscar A. Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit behavioral cloning. In *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pp. 158–168. PMLR, 2022. URL https://proceedings.mlr.press/v164/florence22a.html.

Jing Yu Koh, Stephen McAleer, Daniel Fried, and Ruslan Salakhutdinov. Tree search for language model agents. *arXiv preprint arXiv:2407.01476*, 2024. URL https://arxiv.org/abs/2407.01476.

Alex M. Lamb, Anirudh Goyal, Ying Zhang, Saizheng Zhang, Aaron C. Courville, and Yoshua Bengio. Professor forcing: A new algorithm for training recurrent networks. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Michael Laskey, Jonathan Lee, Roy Fox, Anca Dragan, and Ken Goldberg. DART: Noise injection for robust imitation learning. In *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pp. 143–156, 2017.

Niklas Lauffer, Xiang Deng, Srivatsa R. Kundurthy, Brad Kenstler, and Jeff Da. Imitation learning for multi-turn LM agents via on-policy expert corrections. *arXiv preprint arXiv:2512.14895*, 2025. URL https://arxiv.org/abs/2512.14895.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19274–19286, 2023. URL https://arxiv.org/abs/2211.17192.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024.

Xiao Liu, Tianjie Zhang, Yu Gu, Iat Long Iong, Yifan Xu, Xixuan Song, Shudan Zhang, Hanyu Lai, Xinyi Liu, Hanlin Zhao, Jiadai Sun, Xinyue Yang, Yu Yang, Zehan Qi, Shuntian Yao, Xueqiao Sun, Siyi Cheng, Qinkai Zheng, Hao Yu, Hanchen Zhang, Wenyi Hong, Ming Ding, Lihang Pan, Xiaotao Gu, Aohan Zeng, Zhengxiao Du, Chan Hee Song, Yu Su, Yuxiao Dong, and Jie Tang. VisualAgentBench: Towards large multimodal models as visual foundation agents. *arXiv preprint arXiv:2408.06327*, 2024. URL https://arxiv.org/abs/2408.06327.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Prabhakar, Yun Meng, Haiqiong Zhou, Jingchen Tan, et al. Improve mathematical reasoning in language models by automated process supervision. *arXiv preprint arXiv:2406.06592*, 2024. URL https://arxiv.org/abs/2406.06592.

Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*, 2015. URL https://arxiv.org/abs/1504.04909.

Justin K. Pugh, Lisa B. Soros, and Kenneth O. Stanley. Quality diversity: A new frontier for evolutionary computation. *Frontiers in Robotics and AI*, 3:40, 2016. doi: 10.3389/frobt.2016.00040. URL https://www.frontiersin.org/articles/10.3389/frobt.2016.00040/full.

Pranav Putta, Edmund Mills, Naman Garg, Sumeet Motwani, Chelsea Finn, Divyansh Garg, and Rafael Rafailov. Agent Q: Advanced reasoning and learning for autonomous AI agents. *arXiv preprint arXiv:2408.07199*, 2024. URL https://arxiv.org/abs/2408.07199.

Stephane Ross and J. Andrew Bagnell. Reinforcement and imitation learning via interactive no-regret learning. *arXiv preprint arXiv:1406.5979*, 2014. URL https://arxiv.org/abs/1406.5979.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635. PMLR, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

Nur Muhammad Shafiullah, Zichen Cui, Ariuntuya Altanzaya, and Lerrel Pinto. Behavior transformers: Cloning k modes with one stone. In *Advances in Neural Information Processing Systems*, volume 35, pp. 22955–22968, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/90d17e882adbdda42349db6f50123817-Abstract.html.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.510. URL https://aclanthology.org/2024.acl-long.510/.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.nips.cc/paper_files/paper/2024/hash/5d413e48f84dc61244b6be550f1cd8f5-Abstract-Datasets_and_Benchmarks_Track.html.

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. Language agent tree search unifies reasoning, acting, and planning in language models. In *International Conference on Machine Learning*, 2024.

Shuyan Zhou. WebArena-Infinity: Generating browser environments with verifiable tasks at scale. https://webarena.dev/webarena-infinity/, March 2026. GitHub repository: https://github.com/web-arena-x/webarena-infinity.

A PROOFS

Throughout, the horizon is finite, the action set is finite, and Π is the set of all history-dependent (possibly randomized) policies, so that $V^*(h) = \max_a Q^*(h, a)$ holds by the standard finite-horizon dynamic-programming argument. Non-negativity of harm, $\Delta(h, a) = V^*(h) - Q^*(h, a) \geq 0$ with equality exactly on $\arg\max_{a'} Q^*(h, a')$, is immediate from that identity.

A.1 PROOF OF LEMMA 1

Let $\mathcal{G}_j = \sigma(h_j, a_j)$, $Y_j = V^*(h_j)$ and $\Delta_j = \Delta(h_j, a_j)$. Conditioning on $\mathcal{G}_j$ and integrating over the next observation,

$$\mathbb{E}\big[Y_{j+1} \mid \mathcal{G}_j\big] = \mathbb{E}_{o' \sim P_x(\cdot \mid h_j, a_j)}\big[V^*\big((h_j, a_j, o')\big)\big] = Q^*(h_j, a_j) = Y_j - \Delta_j.$$

Hence for $M_j = Y_j + \sum_{i<j} \Delta_i$, $\mathbb{E}[M_{j+1} \mid \mathcal{G}_j] = Y_j - \Delta_j + \sum_{i \leq j} \Delta_i = M_j$, so M is a martingale. Since $\Delta_j \geq 0$, Y is a supermartingale. The compensator $A_j = \sum_{i<j} \Delta_i$ is $\mathcal{G}_{j-1}$-measurable, hence predictable and non-decreasing, so $Y = M - A$ is the Doob decomposition and is therefore unique. Taking expectations at the terminal index, where $Y_{H+1} = \mathcal{V}(x, \tau)$, gives $\mathbb{E}[\mathcal{V}] + \mathbb{E}[\sum_{j \leq H} \Delta_j] = \mathbb{E}[M_1] = V^*(h_1)$, which is the claim. □

When the collector substitutes teacher actions on a set I of steps, applying the same identity to the mixed process yields $\mathbb{E}[\mathcal{V}] = V^*(h_1) - \mathbb{E}[\sum_{j \notin I} \Delta_j] - \mathbb{E}[\sum_{j \in I} \varepsilon_j]$, where $\varepsilon_j = \Delta(h_j, a_j^T) \geq 0$ is the teacher's own suboptimality at that step. A weaker teacher therefore enlarges an additive, measurable term; it does not change the definition of Δ.

A.2 Proof of Proposition 1

Assume deterministic transitions, so along the realized path $V^*(h_{j+1}) = Q^*(h_j, a_j) = V^*(h_j) - \Delta_j$ exactly, and let $j^\star = \min\{j : \Delta_j > \delta\}$ within the reviewed window $[\sigma, m]$.

(i) Rolling back earlier than $j^\star$ buys little. For $\sigma \le r \le j^\star$, telescoping gives $V^*(h_r) - V^*(h_{j^\star}) = \sum_{i=r}^{j^\star-1} \Delta_i$. Every term is non-negative, and each is at most δ by minimality of $j^\star$, so the sum lies in $[0, \delta(j^\star - r)]$.

(ii) Rolling back later forfeits more than δ. For $j^\star < r \le m+1$, $V^*(h_{j^\star}) - V^*(h_r) = \sum_{i=j^\star}^{r-1} \Delta_i \ge \Delta_{j^\star} > \delta$.

(iii) Optimality of $r = j^\star$. With $U(r) = V^*(h_r) - \kappa(m+1-r)$ we have $U(j^\star) - U(r) = [V^*(h_{j^\star}) - V^*(h_r)] - \kappa(r - j^\star)$. For $r < j^\star$, part (i) gives $U(j^\star) - U(r) \ge (\kappa - \delta)(j^\star - r) > 0$ because $\kappa > \delta$. For $r > j^\star$, part (ii) gives $U(j^\star) - U(r) = \sum_{i=j^\star}^{r-1} \Delta_i - \kappa(r - j^\star) > 0$ because the mean harm on $[j^\star, r)$ exceeds κ. □

A sufficient condition for the latter requirement is $\Delta_i > \kappa$ for every $i \in [j^\star, m]$: once the student has begun to harm recoverability it continues to do so until corrected. This is the precise form of the informal claim that GUI errors do not repair themselves. It is a sufficient condition for rollback optimality rather than a universal property of GUI trajectories. Without determinism the telescoping identities hold in conditional expectation, with an additional mean-zero martingale term.

B SRC Collection Details

Unless otherwise stated, collection uses a fixed review horizon $K = 3$, at most four forks, eight leaves, and six teacher interventions. The WebArena-Infinity student uses a Qwen-style XML tool-call schema over a 1920×1080 viewport. All reported experiments use fixed rather than observation-dependent review schedules.

B.1 Teacher Review and Correction Interface

For each branch, the teacher reviewer receives the task instruction, recent pre-branch context, the pre-action observation for each student action, the executed action dictionaries, the student rationale when available, and the post-branch observation. If the branch proposes `terminate`, hard-verifier feedback is also passed to the reviewer. The reviewer returns one of the following:

- ACCEPT: the branch is locally progress-preserving, even if it differs from the teacher's preferred path.
- ROLLBACK: the first harmful step index and a concise reason.

The teacher should prefer the earliest rollback point that preserves all useful preceding progress. The index is prefix-preserving: `rollback_to`$= k$ keeps branch steps $0, \dots, k-1$ and discards $k, \dots, K-1$. The teacher should not roll back solely for stylistic differences, different but reasonable search terms, or alternate ordering of independent subtasks.

The corrective intervention is generated by a separate teacher query after reset-and-replay restores the rollback state. The rollback reason is included in the correction prompt so the teacher avoids repeating the rejected action. Correction responses use the same XML tool-call schema as the student. A correction may contain multiple low-level tool calls, but we count it as one teacher intervention for budgeting and query accounting.

B.2 Teacher Prompts

We use the following first-shot rollback-review prompt and corrective-intervention prompt.

Rollback review prompt.

Rollback Critic / Locator Prompt

```
You are a teacher reviewing a student GUI agent's recent K-step branch on a web
application. Your first-shot job is ONLY to decide whether this branch should be rolled
back, and if so to locate the FIRST harmful step inside the branch. You are a rollback
critic / locator, not a planner, style judge, or shortest-path judge. You do NOT need to
provide the corrective action — that will be asked separately, AFTER the rollback, with
a fresh screenshot of the rolled-back screen.

INPUT FORMAT: For each of the K student steps you receive the screenshot the student
saw BEFORE acting (un-annotated) along with a text block that states the student's
thought and the action they then took. After all K step images, a final screenshot
shows the resulting screen state. You may also receive environment or verifier feedback
after the branch. Treat explicit verifier feedback as authoritative for whether a
terminal branch actually completed the task.

You may also receive a short PRE-BRANCH CONTEXT section containing recent
screenshots/actions from the trajectory before this branch. Those context steps are
not candidates for rollback in this review; they are provided so you can tell whether
prerequisites were already completed before the current branch began. Do not reject
the current branch merely because a previous prerequisite is no longer visible, if the
context shows it was already done.

For step i<K-1, the screenshot of step i+1 is the state after executing step i. The
final post-branch screenshot is the state after executing step K-1.

IMPORTANT RULES:
 - Accept the branch if it is locally progress-making or harmlessly neutral, even if it
   is not the action sequence you would have chosen.
 - Web tasks have many valid solutions. Do NOT roll back merely because the student's
   path differs from your preferred path, is longer, or is less elegant than your plan.
 - Do NOT intervene just to make the student follow a teacher-preferred strategy.
   Different valid branches are useful and should be preserved.
 - The branch does not need to prove full task success yet. It only needs to preserve
   or improve the chance of eventual success.
 - The student's thought/action text may be wrong. Judge primarily from the screenshot
   transition and the executed action dicts.
 - A branch containing action=terminate with status=success is an irreversible
   end-of-episode claim if accepted. This is the exception to the "accept when
   ambiguous" rule: accept a successful terminate only when the task is objectively
   complete. If verifier feedback is provided and says the task failed, reject the
   branch and roll back to the terminate step unless an earlier step is clearly the
   first harmful step.
 - If verifier feedback is provided, do not override it with visual intuition. Use it to
   decide whether a terminal branch is actually successful, and mention the verifier
   result briefly in the reason.
 - Do NOT judge only by the final post-branch screenshot. Review the trajectory step by
   step. If a branch needlessly changes task-relevant state away from an already-correct
   state, roll back to that first harmful step even if a later step changes it back and
   the final state looks correct.
 - If the pre-branch context or an earlier branch step already completed the task, the
   next acceptable actions are to terminate or perform harmless verification/navigation.
   Re-clicking/toggling the already correct target control is harmful because it can
   undo the task.
 - Only roll back if at least one step (a) does not advance the task, (b) clicks the
   wrong object, (c) creates a loop, (d) leaves the task, (e) substantially raises
   recovery cost / lowers success probability, or (f) prematurely terminates without
   verified task completion.
 - Treat repeated clicks on the same toggle/control, undo-redo patterns, and
   oscillations as loops unless the screenshots show the first click clearly failed to
   take effect.
 - Treat repeated scrolling, hovering, clicking, waiting, or navigation as harmful
   no-progress behavior when the screenshots show little or no useful state change, the
   target remains unavailable, or the branch is repeating an already-failed strategy. Roll
   back to the first no-progress step in that repeated segment.
 - Be strict about task semantics and UI control identity. Do not accept a branch that
   uses a visually similar but semantically different control, label, status marker, menu
   item, setting, or target object.
 - If it is genuinely ambiguous whether the branch is harmful, accept it. Rollback should
   be reserved for objective harm, not uncertainty or preference.
 - If you roll back, identily the EARLIEST harmful step (0-indexed within the branch).
   rollback_to=k means keep steps 0..k-1 and discard k..K-1.

Return ONLY a JSON object. No prose. Schema:
{
   "accept": true|false,
   "rollback_to": <int 0..K-1, only if accept=false>,
   "reason": <short string>
}
```

Figure 4: Rollback-review

Rollback-Aware Corrective Prompt

```
Please generate the next move according to the UI screenshot,
instruction and previous actions.

Instruction: You are interacting with a web application at
{{server_url}} . {{task}}

NOTE FROM REVIEWER: the previous student attempt was just
ROLLED BACK. Reason: {{rollback_reason}}
Take a corrective action that avoids repeating that mistake.

If the current screen already satisfies the task, terminate
with success instead of clicking or toggling an already-correct
control.

Do not repeat a failed pattern. If recent attempts repeatedly
clicked, scrolled, hovered, waited, or navigated without reaching
the needed state, choose a materially different strategy.

If the target item, setting, record, or control is not visible
after repeated manual navigation, consider using an in-app
search, filter, address/search field, command palette, settings
search, or other query mechanism when one is available. This
guidance is application-agnostic: use whatever query/navigation
affordance the current UI actually exposes.

When using a query strategy, output a real executable action,
not just a plan. If the field is already focused, use action=type
with the exact text to enter. If it is not focused, click the
field or control that will focus/open it now. Use action=key
with Enter only when the current UI focus makes that key press
meaningful.

Prefer one concrete progress-making action. Avoid vague clicks,
tiny scrolls that are unlikely to change the screen, and repeated
attempts at the same coordinate unless the screenshot clearly
shows the previous action failed to take effect.

Previous actions:
{{previous_actions}}
```

Figure 5: Corrective-intervention prompt

B.3 RESET-AND-REPLAY

When rollback is required, the collector restores the accepted prefix by resetting the environment, reopening the application, and replaying all actions before the rollback index. Replay skips `terminate` actions but preserves hover and mouse-move actions, because some GUI controls are only exposed after hover. Teacher corrections are inserted into the subsequent history as ordinary action records, which keeps the prompt state aligned with the environment state.

B.4 TRAJECTORY FILTERS AND DEDUPLICATION

The trajectory buffer admits only verifier-passing trajectories satisfying the rule-based constraints

$$|\tau| \leq 60, \quad \text{repeat}(\tau) \leq 4, \quad \text{interventions}(\tau) \leq 6.$$

Here $|\tau|$ counts action steps, $\text{repeat}(\tau)$ is the maximum number of consecutive identical actions, and $\text{interventions}(\tau)$ counts teacher corrections. For storage and deduplication, admitted trajectories are grouped by coarse path-length, most-frequent-action-type, and intervention-count buckets. These descriptors are bookkeeping devices, not semantic strategy labels. Each task/bucket retains at most three trajectories, preferring higher verifier-defined quality when available and otherwise shorter trajectories with fewer teacher interventions.

C ADDITIONAL EXPERIMENTAL DETAILS

C.1 WEBARENA-INFINITY DATA EXPORT

The current WebArena-Infinity data export contains 977 verifier-passing trajectories and 9,183 next-action SFT examples. Among the SFT examples, 7,882 come from accepted student-branch actions and 1,301 come from teacher interventions after rollback. The trajectory difficulty distribution is 279 easy, 244 medium, and 454 hard. We also run basic export validation before training: JSON files are readable, referenced image paths exist, each example has system, user, and assistant messages, and each assistant message follows the expected `Action:` plus `<tool_call>` format. Deduplicating by message content and image content leaves 7,726 unique examples; the remaining 15.9% are duplicate shared prefixes from multi-leaf rollouts.

Table 2: WebArena-Infinity data export summary.

Statistic	Value
Verifier-passing trajectories	977
Next-action SFT examples	9,183
Student-branch examples	7,882
Teacher-intervention examples	1,301
Easy, medium, hard trajectories	279, 244, 454
Unique examples after content deduplication	7,726
Duplicate shared-prefix examples	15.9%

C.2 CONTROLLED WEBARENA-INFINITY SUPERVISION COMPARISON

Corrections and accepted-action training sets cover the same 134 application–task pairs. They contain 468 and 467 unique records, respectively, each repeated three times during data preparation. The three-record difference after repetition is 0.21% of the correction arm. Table 3 reports both the original training format and a subsequent rerun aligning the training messages with the evaluation format. The rerun was introduced after the original comparison, so both results are retained.

Table 3: Controlled WebArena-Infinity results on the same 346 held-out tasks. These use a separate protocol from Table 1.

Training data / format	Successes	SR (%)
Base	12	3.47
Expert SFT	29	8.38
SRC data only	8	2.31
Expert + SRC data	46	13.29
Expert + corrections	22	6.36
Corrections / original format	22	6.36
Accepted actions / original format	13	3.76
Corrections / aligned format	14	4.05
Accepted actions / aligned format	4	1.16

Corrections exceed accepted actions by 2.60 percentage points in the original comparison (17:8 discordant pairs, $p = 0.1078$), and by 2.89 points in the aligned-format rerun (12:2, $p = 0.0129$). Correcting for these two versions of the same comparison gives $p = 0.0259$ for the latter. Neither correction arm shows a detectable gain over base ($p = 0.0987$ and 0.8318). Adding all SRC data to Expert SFT yields a detectable increment ($p = 0.0359$), whereas adding only corrections does not ($p = 0.2478$). The Expert-plus-corrections comparison also differs in training-update count and therefore does not isolate supervision composition alone.

C.3 Localized versus Random Rollback

We compare teacher-localized rollback with a uniformly sampled rollback position inside the reviewed branch on the 186-task OSWorld subset, using the same teacher, student, collection budget, training schedule, and $K = 3$. The two checkpoints are evaluated under the same greedy-decoding protocol and 30-step limit. Teacher-localized rollback reaches 40.3% SR, compared with 25.8% for random rollback, a difference of 14.5 percentage points. This matched comparison supports the value of teacher-guided error localization.

C.4 Replay Fidelity by Prefix Length

Table 4 breaks WebArena-Infinity replay agreement down by the length of the restored prefix. Missing snapshots are excluded from the agreement denominator.

Table 4: WebArena-Infinity replay fidelity by restored prefix length.

Prefix length	Match	Mismatch	Missing	Agreement (%)
0–2	98	1	0	99.0
3–7	257	12	7	95.5
8+	802	66	78	92.4
Overall	1,157	79	85	93.6

C.5 Training-Time Teacher Use and Collector Diagnostics

Table 5 pairs the teacher-free deployment results from Table 1 with their training-time teacher-query costs. For SRC, we additionally report the online teacher-assisted collector SR as a collection diagnostic; this quantity is not a deployed-policy result. A dash indicates that no corresponding collector diagnostic is reported.

Table 5: Teacher-free deployed-policy SR, online collector SR where available, and training-time teacher queries per task. SR values are percentages.

Benchmark	Training scheme	Deployed SR	Collector SR	Teacher q./task
WebArena-Infinity	Expert SFT + SRC	35.0	42.5	12.01
	OEC-style random switch	20.4	–	10.09
	LEAP-style post-hoc correction	31.8	–	27.42
WebArena-Lite	Expert SFT + SRC	24.2	36.0	11.90
	OEC-style random switch	14.5	–	10.15
	LEAP-style post-hoc correction	19.4	–	28.64
OSWorld subset	Expert SFT + SRC	40.3	43.27	5.29

C.6 WebArena-Infinity Reference Runs

Table 7 reports the complete closed-agent references on the WebArena-Infinity real-task split. The split has 260 easy, 260 medium, and 1,100 hard tasks.

Table 6: WebArena-Infinity difficulty breakdown for the main-result rows. The revised per-difficulty base-model breakdown is not reported; its updated aggregate SR is 16.4%.

Method	Easy SR	Medium SR	Hard SR
Base model	n.r.	n.r.	n.r.
Expert SFT	66.9	42.3	11.5
SFT collector with SRC	87.1	67.1	26.2
OEC-style random switch	57.3	43.1	6.3
Expert SFT + SRC	67.3	56.2	22.3
Kimi K2.5	69.2	57.3	28.6
Qwen3.5 Plus	77.3	56.5	32.7

Table 7: Closed-agent references on WebArena-Infinity real tasks. Average steps are computed over successful trajectories.

Model	Difficulty	Success	Avg. steps
Kimi K2.5	Easy	69.23%	5.32
Kimi K2.5	Medium	57.31%	9.07
Kimi K2.5	Hard	28.55%	14.26
Kimi K2.5	All	39.69%	10.55
Qwen3.5 Plus	Easy	77.31%	7.00
Qwen3.5 Plus	Medium	56.54%	9.91
Qwen3.5 Plus	Hard	32.73%	14.33
Qwen3.5 Plus	All	43.70%	11.33

C.7 Fixed-Horizon and Prefix-Retention Diagnostics

The fixed-horizon ablation reaches its highest collection SR at $K = 3$: 51.9%, with 2,529 teacher queries, compared with 45.6% and 3,324 queries at $K = 1$. At $K = 7$, SR is 50.6% with 1,831 queries. These results characterize sensitivity to a fixed review interval; they do not compare fixed review with an adaptive or progress-aligned schedule. Review queries are first-shot branch reviews, interventions are corrective queries after rollback, and their sum is the total teacher-query count.

Table 8: Ablation over fixed speculative review horizon on WebArena-Infinity tasks aggregated across Gmail, Linear-account-settings, and GitLab Plan-and-Track.

K	SR	Easy SR	Medium SR	Hard SR	Review q.	Interv.	Total q.	Avg. rollback	Accept rate
1	45.6	81.7	75.0	26.0	2603	721	3324	1.00	**63.7**
3	**51.9**	**91.7**	**76.7**	32.5	1744	785	2529	2.20	37.1
5	50.6	83.3	71.7	**34.5**	1708	881	2589	2.73	23.0
7	50.6	81.7	**76.7**	33.5	**1227**	**604**	**1831**	3.43	20.9

Table 9 reports how often the teacher selects index zero and therefore discards an entire full-length rejected branch. These positions are teacher judgments rather than independently measured harmful-action onsets. The decline with K alone does not demonstrate boundary misalignment: even an independent, constant per-step hazard p gives a conditional first-position probability of $p/[1-(1-p)^K]$, rather than $1/K$.

Table 9: Whole-segment discard among full-length rejected branches. The $K = 1$ case is omitted because index zero is forced by construction.

Environment	$K = 3$ (%)	$K = 7$ (%)
WebArena-Infinity	82.3	63.4
OSWorld	71.5	62.4

Table 10 expands the branch-horizon ablation by application.

Table 10: Per-app fixed-horizon ablation on WebArena-Infinity.

App	K	SR	Easy SR	Medium SR	Hard SR	Review q.	Interv.	Total q.	Accept
GitLab Plan-and-Track	1	22.1	60.0	50.0	9.0	413	40	453	85.2
GitLab Plan-and-Track	3	28.6	80.0	65.0	11.0	353	116	469	51.8
GitLab Plan-and-Track	5	28.6	70.0	45.0	17.0	537	237	774	29.8
GitLab Plan-and-Track	7	27.9	65.0	60.0	14.0	272	99	371	30.1
Gmail	1	65.0	85.0	75.0	35.0	1046	388	1434	52.4
Gmail	3	65.0	95.0	65.0	35.0	373	192	565	28.4
Gmail	5	60.0	80.0	70.0	30.0	273	145	418	20.9
Gmail	7	60.0	80.0	70.0	30.0	252	120	372	19.8
Linear Account Settings	1	63.3	100.0	100.0	45.0	1144	293	1437	66.2
Linear Account Settings	3	72.5	100.0	100.0	58.8	1018	477	1495	35.2
Linear Account Settings	5	71.7	100.0	100.0	57.5	898	499	1397	19.5
Linear Account Settings	7	72.5	100.0	100.0	58.8	703	385	1088	17.8