FROM TRAJECTORY MEMORY TO ACTION SELECTION: TRAJECTORY-CONTEXT ACTION CREDIT FOR LANGUAGE AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Language agents increasingly reuse past interaction trajectories as examples, memories, abstractions, and decision signals. However, retrieving a relevant trajectory does not directly tell an agent which action it should prefer at the current Decision State. A trajectory contains temporally extended experience, while action selection requires choosing among the Candidate Actions available now. We formalize this missing link as Trajectory-Context Action Credit, a candidate-level signal that makes the contribution of trajectory memory to action selection explicit. We introduce TrajCredit, a training-free instantiation that converts Retrieved Slices into candidate-level credit using Local State-Action Support, Procedural-Stage Consistency, and Outcome-Calibrated Evidence, and combines the resulting credit with an existing Base Selector. Across controlled candidate-action ranking, retrieval and memory controls, credit-to-progress alignment, and bounded environment intervention, TrajCredit consistently improves action ranking, provides information beyond raw retrieval support, aligns with independently measured short-horizon value, and improves executed behavior. These results support Trajectory-Context Action Credit as a useful interface between reusable agent experience and candidate-action selection.

1 INTRODUCTION

Large language model agents increasingly operate in interactive environments where solving a task requires repeated observation, reasoning, action, and adaptation to feedback. Past interaction experience has therefore become an important resource for future decisions, reused through retrieved trajectories, reflections, abstractions, skills, value signals, and search (Yao et al., 2023b; Shinn et al., 2023; Zhao et al., 2024; Wang et al., 2024; Ichter et al., 2023; Zhou et al., 2024a). These directions show that agent behavior depends not only on the base language model, but also on how previous interaction evidence is represented and brought into the current decision.

Experience reuse, however, creates an attribution problem between memory and action selection. A trajectory spans multiple steps, while action selection requires an immediate choice among Candidate Actions. Retrieving a relevant trajectory does not determine what that trajectory implies for the actions available now. A retrieved trajectory can describe a different state or procedural stage, and an action that was useful earlier may become redundant or invalid after the environment changes. Successful trajectories can also contain locally unhelpful actions, while failed trajectories can contain locally useful ones. Retrieval relevance is therefore not equivalent to action support.

This motivates a decision-time interface that converts retrieved trajectory evidence into support for the current Candidate Actions. We study decision settings where a finite Candidate Set is supplied by the environment or an upstream proposal mechanism. If trajectory reuse changes the resulting choice, it must ultimately change the relative support assigned to these actions, even when that support remains implicit in prompting, reasoning, or search. The same retrieved experience can support one candidate more strongly than another, but trajectory-level context alone does not expose this difference. We therefore isolate the candidate-level signal through which trajectory experience contributes to candidate-action selection.

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.

We formalize this signal as **Trajectory-Context Action Credit**. Given a Decision State, a Candidate Set, and an Action-Centered Trajectory Memory, it assigns a scalar credit to each Candidate Action based on evidence from past trajectories. Constructing this credit requires answering three attribution questions. First, which current action does a Retrieved Slice support? Second, is that support compatible with the current procedural stage? Third, is that support associated with subsequent progress in the reference trajectory? These questions separate candidate-level attribution from Evidence Retrieval and make the contribution of trajectory memory to action selection explicit and directly evaluable.

These considerations motivate **TrajCredit**, a training-free instantiation of Trajectory-Context Action Credit. TrajCredit retrieves contextually relevant slices and converts them into candidate-specific support through Local State-Action Support, Procedural-Stage Consistency, and Outcome-Calibrated Evidence. Candidate-Level Credit Aggregation combines these views into a credit score for each candidate, which is then fused with an existing Base Selector.

We evaluate three claims implied by this interface: whether TrajCredit improves Candidate Action ranking across Base Selectors, whether its gains extend beyond raw retrieval support and depend on aligned trajectory evidence, and whether its credit aligns with independently measured Short-Horizon Environment Value and improves behavior under Bounded Environment Intervention.

Our contributions are:

- We identify and formalize Trajectory-Context Action Credit as an explicit candidate-level interface between trajectory memory and candidate-action selection.
- We introduce TrajCredit, a training-free and decomposable instantiation that converts Retrieved Slices into candidate-level credit through Local State-Action Support, Procedural-Stage Consistency, and Outcome-Calibrated Evidence.
- We establish the utility and meaning of this credit signal through controlled ranking, retrieval and memory controls, credit-to-progress alignment, and Bounded Environment Intervention, showing gains in action ranking and executed behavior, together with credit that extends beyond raw retrieval support and aligns with short-horizon value.

2 Related Work

2.1 Trajectories as Reusable Agent Experience

Interactive agents naturally produce trajectories of observations, reasoning, and actions. ReAct established this interaction pattern (Yao et al., 2023b). Subsequent work either reuses past trajectories directly as retrieved exemplars or task-relevant experience (Zheng et al., 2024; Liu et al., 2025; Sarukkai et al., 2025), or transforms interaction experience into reflections and structured experience (Shinn et al., 2023; Zhao et al., 2024; Majumder et al., 2023), context-conditioned guidance, actionable abstractions, reusable skills, and transferable experience (Fu et al., 2024; Sarch et al., 2024; Wang et al., 2024; 2025; Gao et al., 2025; Hu et al., 2025). Recent analysis further shows that similarity-based experience reuse can propagate errors when retrieved memory is poorly aligned with the current decision (Xiong et al., 2026).

These approaches improve how experience is stored, retrieved, or presented to the policy, while its effect on the current Candidate Set often remains implicit. Recent work further separates retrieval from post-retrieval reuse by converting long-horizon trajectories into target-conditioned procedural support (Li et al., 2026). Our focus is one step closer to action selection: how retrieved evidence becomes explicit relative support over the current Candidate Actions.

2.2 From Experience Reuse to Action Credit

Action-level scoring is also well established. SayCan combines language-model action likelihood with learned affordance or value estimates (Ichter et al., 2023). Recent approaches learn critics, discriminators, or state-action value functions (Bai et al., 2025; Lin et al., 2025; Dong et al., 2025; Qian et al., 2025); Retrospex learns an offline critic from past trajectories for candidate rescoring (Xiang et al., 2024). Other work infers value from contextual examples (Ma et al., 2025) or episodic

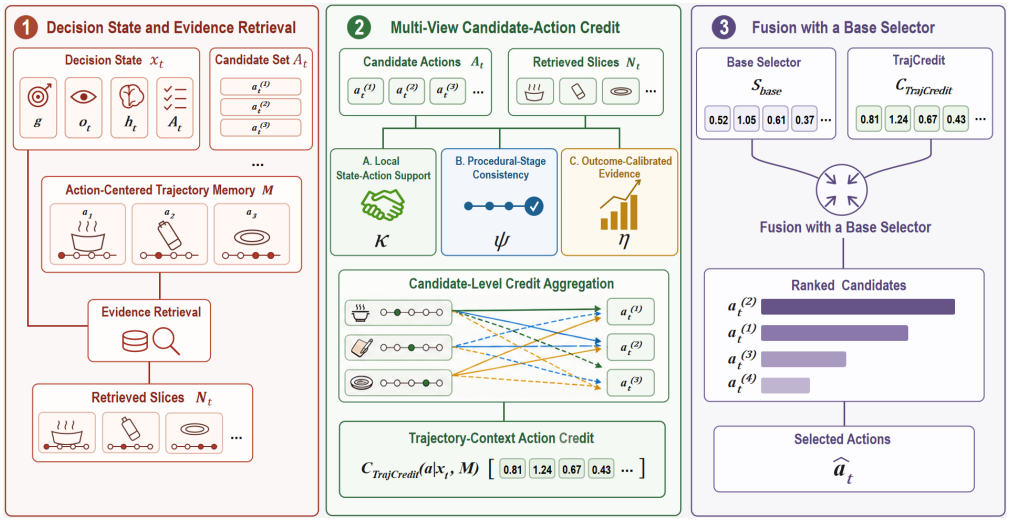

Figure 1: Overview of TrajCredit. Relevant trajectory evidence is retrieved from Action-Centered Trajectory Memory and attributed to Candidate Actions through Local State-Action Support, Procedural-Stage Consistency, and Outcome-Calibrated Evidence. The resulting Trajectory-Context Action Credit is fused with a Base Selector to rank and select the current action.

returns (Pritzel et al., 2017), while trajectory attribution has been studied for explaining trained reinforcement-learning policies (Deshmukh et al., 2023). Unlike these learned or value-estimation approaches, TrajCredit directly attributes retrieved trajectory evidence to the current Candidate Set without training an auxiliary critic.

Action evaluation also appears within broader planning and learning systems. Tree of Thoughts, RAP, RAFA, LATS, ToolChain, and DEPS evaluate candidate trajectories during reasoning, planning, or search (Yao et al., 2023a; Hao et al., 2023; Liu et al., 2024; Zhou et al., 2024a; Zhuang et al., 2024; Wang et al., 2023), while ETO and ArCHer use trajectory feedback for policy improvement (Song et al., 2024; Zhou et al., 2024b). These methods couple action evaluation with search, rollout, or policy adaptation. In contrast, we isolate how retrieved trajectory evidence becomes explicit relative support over the current Candidate Set.

3 Problem Setup: Trajectory-Context Action Credit

Experience affects control only when it changes the choice among candidate actions. We study this decision time interface directly. At step t, an agent receives a task description g, an observation o_t, and a recent interaction history h_t. The action interface provides a finite candidate set

$$\mathcal{A}_t = \{a_t^{(1)}, \ldots, a_t^{(K)}\}. \tag{1}$$

The Candidate Set contains the actions presented to the selector at the current decision point. We define the decision state as

$$x_t = (g, o_t, h_t, \mathcal{A}_t). \tag{2}$$

A base selector assigns a real-valued score to each candidate:

$$S_{\text{base}}(a \mid x_t), \qquad a \in A_t. \tag{3}$$

It may be an LM likelihood, prompted ranker, score-fusion rule, or memory-augmented selector. We keep it fixed and ask what additional action-level signal can be supplied by stored trajectories.

Let M denote a memory of past interaction trajectories. We define Trajectory-Context Action Credit as

$$C(a \mid x_t, M), \qquad a \in A_t. \tag{4}$$

Unlike trajectory-level context or a state-value estimate, C assigns evidence to each current candidate separately. By fixing x_t, A_t, and S_{base}, we evaluate trajectory memory directly through its effect on candidate ranking.

4 METHOD

TrajCredit is a training-free instantiation of the Trajectory-Context Action Credit interface. We use $C(a \mid x_t, M)$ for the general interface and $C_{\text{TrajCredit}}(a \mid x_t, M)$ for TrajCredit. It constructs candidate-level credit through Local State-Action Support, Procedural-Stage Consistency, and Outcome-Calibrated Evidence. All coefficients are fixed before evaluation. Figure 1 summarizes the pipeline.

Trajectory memory must preserve the information needed to attribute past experience to current actions. TrajCredit therefore represents the Action-Centered Trajectory Memory as action-centered slices. Each memory item is written as

$$m_i = (g_i, o_i, h_i, a_i, p_i, \Delta_i^{\text{imm}}, \Delta_i^{\text{fut}}). \tag{5}$$

Here, g_i is the task description, o_i is the pre-action observation, h_i is the recent history before the action, and a_i is the action taken. The scalar p_i is a stage proxy, implemented with the pre-action score or progress estimate when available. The terms Δ_i^{imm} and Δ_i^{fut} denote immediate and future progress measured from the reference trajectory.

Each component serves a distinct role in subsequent attribution. The tuple (g_i, o_i, h_i) determines whether the slice is contextually relevant, a_i identifies which current action the slice can support, p_i captures procedural-stage compatibility, and the progress terms indicate whether the surrounding trajectory segment subsequently moved the task forward.

4.1 EVIDENCE RETRIEVAL

Evidence Retrieval determines which past slices are relevant to the current context, but it does not determine which current action they support. For the current Decision State, TrajCredit constructs a context query

$$q_t = \text{concat}(g, o_t, h_t). \tag{6}$$

For each memory item, it constructs

$$q_i = \text{concat}(g_i, o_i, h_i). \tag{7}$$

The retrieved neighborhood is

$$\mathcal{N}_t = \text{Top}_N\big(\{m_i \mid m_i \in \mathcal{M}\}, r(q_t, q_i)\big). \tag{8}$$

Here, $r(q_t, q_i)$ is a context similarity score and N is the retrieval budget. In our implementation, r is TF-IDF cosine similarity over task, observation, and history text. The Candidate Set $\mathcal{A}_t$ does not enter the retrieval query. It is introduced only after retrieval, when the Retrieved Slices are attributed to the current candidate actions. The Trajectory-Context Action Credit interface does not depend on this specific retriever.

For each retrieved item, we define a nonnegative retrieval weight

$$\tilde{w}_i = \max(0, r(q_t, q_i)). \tag{9}$$

We normalize this weight within the retrieved neighborhood:

$$w_i = \frac{\tilde{w}_i}{\sum_{m_j \in \mathcal{N}_t} \tilde{w}_j + \epsilon_w}. \tag{10}$$

Here, $\epsilon_w > 0$ avoids division by zero. The normalized weight w_i measures how strongly retrieved item m_i contributes as contextual evidence.

4.2 MULTI-VIEW CANDIDATE-ACTION CREDIT

The Retrieved Slices remain trajectory-level evidence until they are attributed to the actions in the current Candidate Set. TrajCredit performs this attribution through three complementary evidence views corresponding to the three questions above.

The first view is Local State-Action Support. It asks whether the action taken in a slice retrieved under similar context supports the current candidate action under consideration. For $a \in \mathcal{A}_t$ and $m_i \in \mathcal{N}_t$, we define

$$\kappa(a, a_i) = \max\left(\mathbf{1}[a = a_i], s_{\text{act}}(a, a_i)\right). \tag{11}$$

Here, $s_{\mathrm{act}}(a, a_i) \in [0, 1]$ is an action string similarity score. We implement s_{act} with character n-gram cosine similarity, which captures lexical overlap in structured command strings, including operation words, object names, and receptacles. Larger $\kappa(a, a_i)$ implies stronger candidate-level support from the retrieved action.

The second view is Procedural-Stage Consistency. Local State-Action Support can be misleading when the retrieved slice comes from a different task stage. Let p_t denote the stage proxy for the current Decision State. The concrete proxy is fixed within each benchmark and uses the same convention as p_i. TrajCredit defines

$$\psi_i(t) = \exp\left(-\frac{|p_t - p_i|}{\sigma_p}\right). \tag{12}$$

Here, $\sigma_p > 0$ controls tolerance to stage mismatch. Larger $\psi_i(t)$ implies that the retrieved slice is more compatible with the current procedural stage. Recent history has already influenced which slices are retrieved through q_t and q_i, while $\psi_i(t)$ specifically measures stage compatibility. When no reliable stage proxy is available, we set $\psi_i(t) = 1$, and stage modulation is then inactive and P reduces to L.

The third view is Outcome-Calibrated Evidence. Observing an action in memory does not by itself establish that the action was useful. TrajCredit therefore uses the subsequent progress of the surrounding reference segment to calibrate its evidence:

$$\eta_i = \mathrm{norm}\left(\max(0, \Delta_i^{\mathrm{fut}}) + \max(0, \Delta_i^{\mathrm{imm}})\right). \tag{13}$$

Here, $\mathrm{norm}(\cdot)$ maps progress values to $[0, 1]$. The normalization statistics are computed from the trajectory-memory split and then fixed for evaluation. Future progress is included because useful procedural steps may not produce immediate score gain. Larger η_i implies stronger evidence from segments that subsequently moved the task forward. The progress terms are stored only for memory trajectories and are not computed from held-out Decision State continuations during scoring.

4.3 Candidate-Level Credit Aggregation

The three evidence views are converted into candidate-level scores by aggregation over the Retrieved Slices. Each score retains Local State-Action Support through $\kappa(a, a_i)$. Procedural-Stage Consistency and Outcome-Calibrated Evidence further modulate that support according to stage compatibility and subsequent progress. The local support score is

$$L(a \mid x_t, \mathcal{M}) = \sum_{m_i \in \mathcal{N}_t} w_i \kappa(a, a_i). \tag{14}$$

The procedural-stage score is

$$P(a \mid x_t, \mathcal{M}) = \sum_{m_i \in \mathcal{N}_t} w_i \kappa(a, a_i) \psi_i(t). \tag{15}$$

The outcome-calibrated score is

$$O(a \mid x_t, \mathcal{M}) = \sum_{m_i \in \mathcal{N}_t} w_i \kappa(a, a_i) \eta_i. \tag{16}$$

Accordingly, L, P, and O represent local support, stage-compatible support, and outcome-calibrated support, respectively. The final TrajCredit is

$$C_{\mathrm{TrajCredit}}(a \mid x_t, \mathcal{M}) = \theta_L L(a \mid x_t, \mathcal{M}) + \theta_P P(a \mid x_t, \mathcal{M}) + \theta_O O(a \mid x_t, \mathcal{M}). \tag{17}$$

For the full model, the view weights satisfy

$$\theta_L + \theta_P + \theta_O = 1. \tag{18}$$

Here, $\theta_L, \theta_P, \theta_O \geq 0$. The view weights are fixed before evaluation. Removing one candidate-level score sets its corresponding weight to zero while leaving the remaining credit interface unchanged.

4.4 Fusion with a Base Selector

Trajectory-Context Action Credit is used as a plug-in signal for action selection. To combine it with a Base Selector whose scores may have a different scale, we normalize both signals over the current Candidate Set. For any candidate scoring function $u(a)$, define

$$z_{\mathcal{A}_t}(u, a) = \frac{u(a) - \mu_{\mathcal{A}_t}(u)}{\sigma_{\mathcal{A}_t}(u) + \epsilon_z}. \tag{19}$$

The candidate set mean is

$$\mu_{\mathcal{A}_t}(u) = \frac{1}{K} \sum_{a \in \mathcal{A}_t} u(a). \tag{20}$$

The candidate set standard deviation is

$$\sigma_{\mathcal{A}_t}(u) = \left(\frac{1}{K} \sum_{a \in \mathcal{A}_t} \left(u(a) - \mu_{\mathcal{A}_t}(u) \right)^2 \right)^{1/2}. \tag{21}$$

Here, $\epsilon_z > 0$ avoids division by zero.

The final score is

$$S_{\text{final}}(a \mid x_t) = z_{\mathcal{A}_t}\left(S_{\text{base}}, a\right) + \lambda z_{\mathcal{A}_t}\left(C_{\text{TrajCredit}}, a\right). \tag{22}$$

Here, $\lambda \geq 0$ controls the influence of Trajectory-Context Action Credit. The selected action is

$$\hat{a}_t = \arg\max_{a \in \mathcal{A}_t} S_{\text{final}}(a \mid x_t). \tag{23}$$

Because fusion operates only on candidate scores, the same Trajectory-Context Action Credit interface can be composed with different Base Selectors without changing their underlying decision mechanisms.

5 Experiments

We evaluate Trajectory-Context Action Credit through three questions. First, does it improve the ranking of Candidate Actions across different Base Selectors? Second, do the gains depend on structured and aligned trajectory evidence rather than raw retrieval support alone? Third, does the resulting credit align with Short-Horizon Environment Value and improve behavior when used for action selection? We address these questions through controlled candidate-action ranking, attribution controls, credit-to-progress alignment, and Bounded Environment Intervention.

5.1 Experimental Setup

We evaluate TrajCredit on ScienceWorld (Wang et al., 2022) and ALFWorld (Shridhar et al., 2021). ScienceWorld emphasizes long-horizon procedural tasks, while ALFWorld contains household manipulation tasks with strong local object-state dependencies. Each controlled evaluation item is a Decision State $x_t = (g, o_t, h_t, \mathcal{A}_t)$, and all methods rank the same Candidate Set $\mathcal{A}_t$. Top-1 accuracy measures whether the highest-ranked candidate matches the reference action in the held-out trajectory.

We use Qwen2.5-3B and Qwen2.5-7B as the primary backbones and Llama-3.1-8B for cross-family evaluation. We construct protocol-aligned action-ranking variants of ReAct (Yao et al., 2023b), SayCan (Ichter et al., 2023), Reflexion (Shinn et al., 2023), ExpeL (Zhao et al., 2024), and CLIN (Majumder et al., 2023) within the same candidate-action interface. These selectors cover LM-based ranking, score fusion, and retrieved-experience conditioning. TrajCredit is added through the same fusion interface without changing the underlying Base Selector. Each paired comparison uses identical Decision States, Candidate Sets, model budgets, and trajectory pools.

All TrajCredit coefficients, including one fusion coefficient λ per selector family, are selected on an independent development split and frozen before test evaluation. Trajectory memory, development states, and test states are disjoint at the episode level. We report Top-1 accuracy as mean $\pm$ standard deviation over task–seed units and use paired hierarchical bootstrap resampling over tasks and Decision States for confidence intervals. Additional ranking summaries, fusion-weight sensitivity, memory-quality robustness, and extended value and intervention analyses are provided in the appendix.

Table 1: Controlled candidate-action ranking across benchmarks, Base Selectors, and model backbones. Top-1 accuracy is reported as mean ± standard deviation over task–seed units. Bold indicates the better result within each base/TrajCredit pair.

Method	SW 3B	SW 7B	SW Llama-8B	ALF 3B	ALF 7B
ReAct (Yao et al., 2023b)	0.1738 ± 0.0661	0.2470 ± 0.0478	0.2301 ± 0.0591	0.4206 ± 0.0308	0.4659 ± 0.0379
ReAct + TrajCredit	$\mathbf{0.4794 \pm 0.0617}$	$\mathbf{0.5001 \pm 0.0703}$	$\mathbf{0.5016 \pm 0.0654}$	$\mathbf{0.7186 \pm 0.0270}$	$\mathbf{0.7257 \pm 0.0109}$
SayCan (Ichter et al., 2023)	0.4275 ± 0.1977	0.4489 ± 0.2024	0.4499 ± 0.2105	0.6317 ± 0.0156	0.6553 ± 0.0229
SayCan + TrajCredit	$\mathbf{0.4715 \pm 0.0717}$	$\mathbf{0.4866 \pm 0.0779}$	$\mathbf{0.4903 \pm 0.0765}$	$\mathbf{0.6868 \pm 0.0373}$	$\mathbf{0.6912 \pm 0.0270}$
Reflexion (Shinn et al., 2023)	0.4060 ± 0.0925	0.5808 ± 0.1589	0.4562 ± 0.1181	0.6088 ± 0.0164	0.7521 ± 0.0458
Reflexion + TrajCredit	$\mathbf{0.5341 \pm 0.0712}$	$\mathbf{0.5981 \pm 0.0626}$	$\mathbf{0.5422 \pm 0.0572}$	$\mathbf{0.7689 \pm 0.0309}$	$\mathbf{0.8272 \pm 0.0371}$
ExpeL (Zhao et al., 2024)	0.4295 ± 0.1174	0.4686 ± 0.1302	0.4050 ± 0.1015	0.6109 ± 0.0371	0.7585 ± 0.0329
ExpeL + TrajCredit	$\mathbf{0.5462 \pm 0.0850}$	$\mathbf{0.5588 \pm 0.0593}$	$\mathbf{0.5318 \pm 0.0619}$	$\mathbf{0.7498 \pm 0.0324}$	$\mathbf{0.8503 \pm 0.0256}$
CLIN (Majumder et al., 2023)	0.3887 ± 0.1072	0.4756 ± 0.1323	0.4428 ± 0.1290	0.5842 ± 0.0373	0.7460 ± 0.0359
CLIN + TrajCredit	$\mathbf{0.5325 \pm 0.0714}$	$\mathbf{0.5473 \pm 0.0640}$	$\mathbf{0.5419 \pm 0.0603}$	$\mathbf{0.7835 \pm 0.0332}$	$\mathbf{0.8517 \pm 0.0271}$

Table 2: Retrieval-confusing control on ScienceWorld with Qwen2.5-7B. Entries are Top-1 accuracy. Paired differences compare each TrajCredit variant with its corresponding Base Selector and report 95% hierarchical bootstrap confidence intervals.

Candidate set	ReAct	ReAct + TrajCredit	Paired Δ	SayCan	SayCan + TrajCredit	Paired Δ
Standard	0.2470	**0.5001**	$\mathbf{+0.2531}$ $[\mathbf{0.226}, \mathbf{0.280}]$	0.4489	**0.4866**	$\mathbf{+0.0377}$ $[\mathbf{0.019}, \mathbf{0.057}]$
Retrieval-confusing	0.2144	**0.4278**	$\mathbf{+0.2134}$ $[\mathbf{0.176}, \mathbf{0.251}]$	0.3644	**0.4248**	$\mathbf{+0.0604}$ $[\mathbf{0.034}, \mathbf{0.087}]$

5.2 Controlled Candidate-Action Ranking

We first test whether explicit Trajectory-Context Action Credit improves action ranking when the Decision State, Candidate Set, and Base Selector are held fixed. Each TrajCredit row in Table 1 is paired with the immediately preceding base row, so the comparison isolates the contribution of candidate-level trajectory credit.

TrajCredit improves Top-1 ranking in all 25 method–benchmark–backbone comparisons (Table 1; Figure 2a). The gains extend from LM-based and score-fusion selectors to Reflexion, ExpeL, and CLIN, whose base variants already receive retrieved experience. Explicit candidate-level credit therefore provides information complementary to both the Base Selector and trajectory context supplied through retrieval.

5.3 Attributing the Ranking Gains

The ranking gains in Table 1 do not identify why the credit is useful. We therefore ask whether the gains can be explained by raw retrieval support, and which parts of the proposed credit construction account for the remaining benefit. Table 2 addresses the first question, while Table 3 evaluates the candidate-level scores and memory alignment.

Retrieval-confusing candidates. We construct retrieval-confusing Candidate Sets by replacing ordinary distractors with incorrect actions that receive high raw retrieval support, while keeping the Candidate Set size, Decision State, and reference action unchanged. This directly tests whether TrajCredit provides information beyond retrieved-action similarity.

TrajCredit retains positive paired gains for both ReAct and SayCan under retrieval-confusing candidates, with confidence intervals excluding zero. Thus, the gain persists even when incorrect candidates are deliberately given strong raw retrieval support, ruling out the simple explanation that TrajCredit merely follows retrieved-action similarity.

Evidence-view attribution. We next intervene on Candidate-Level Credit Aggregation while keeping Evidence Retrieval, Candidate Sets, the Base Selector, and the fusion protocol fixed. Each score ablation removes one term from the final credit aggregation. The random-memory control instead permutes memory across Decision States within each benchmark while preserving memory size, trajectory length, and marginal action frequency.

The ablations show that candidate-level scores contribute differently across environments. On ScienceWorld, removing procedural-stage consistency causes the largest drop, while removing local

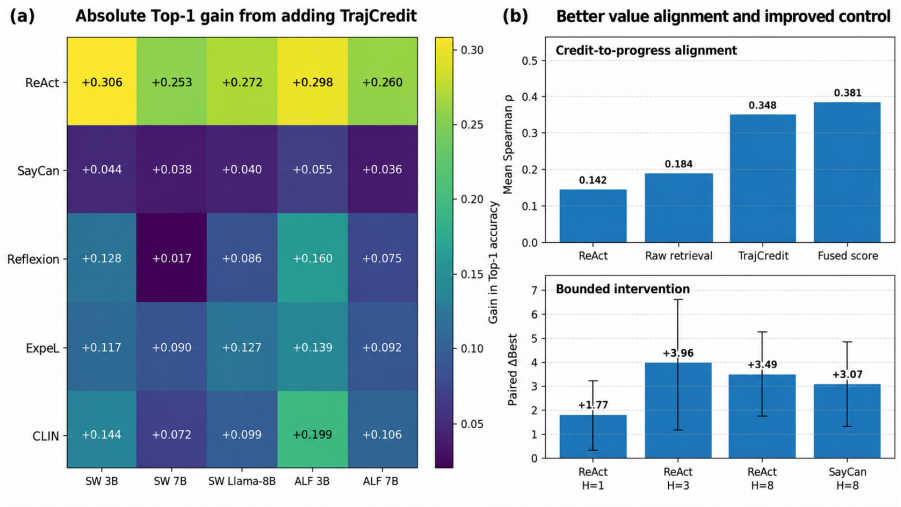

Figure 2: Summary of the main empirical results. (a) Absolute Top-1 gains from adding TrajCredit across Base Selectors, benchmarks, and model backbones. (b) Credit-to-progress alignment with forced-rollout value (top) and paired improvement in ΔBest under bounded environment intervention (bottom).

Table 3: Candidate-level score and memory-alignment ablations. All variants use the same ReAct Base Selector and report Top-1 accuracy as mean ± standard deviation. Each applicable score ablation removes one term from Candidate-Level Credit Aggregation. Stage ablation is N/A on ALFWorld because $\psi_i(t) = 1$ and $P = L$.

Benchmark	Model	TrajCredit	w/o local	w/o stage	w/o outcome	Random memory
ScienceWorld	Qwen 3B	$\mathbf{0.4794 \pm 0.0617}$	0.4035 ± 0.1504	0.3662 ± 0.1342	0.4498 ± 0.1690	0.2196 ± 0.0705
ScienceWorld	Qwen 7B	$\mathbf{0.5001 \pm 0.0703}$	0.4218 ± 0.1567	0.3824 ± 0.1398	0.4687 ± 0.1810	0.2462 ± 0.0696
ALFWorld	Qwen 3B	$\mathbf{0.7186 \pm 0.0270}$	0.5704 ± 0.0355	N/A	0.6802 ± 0.0338	0.3768 ± 0.0252
ALFWorld	Qwen 7B	$\mathbf{0.7257 \pm 0.0109}$	0.5731 ± 0.0409	N/A	0.6844 ± 0.0451	0.5073 ± 0.0368

support has the largest effect on ALFWorld. Outcome-calibrated evidence improves all settings, and randomized memory causes a substantially larger degradation, showing that useful credit depends on Decision State–trajectory alignment rather than memory size or marginal action frequency alone.

5.4 Credit-to-Progress Alignment

Reference-action accuracy establishes ranking utility, but it does not show whether the induced ordering reflects the consequences of alternative actions. A value-like credit signal should also rank candidates consistently with their Short-Horizon Environment Value. We therefore compare decision-time scores with an independent value estimate obtained by executing each candidate in the environment.

We evaluate 80 held-out ScienceWorld handoff states using the same score-independent set of eight candidate actions at each state. Every candidate is force-executed once and then followed by the same frozen ReAct continuation policy for three additional steps. The continuation policy does not use TrajCredit or any score signal under evaluation. The resulting ΔBest provides an independent Short-Horizon Environment Value for each candidate.

Table 4 and Figure 2(b, top) show that TrajCredit is more strongly aligned with forced-rollout value than either the ReAct score or raw retrieval, with mean Spearman correlations of 0.348, 0.142, and 0.184, respectively. Its paired improvement over ReAct excludes zero. The fused score further improves rank alignment and high-value action coverage. Thus, the ranking gains in Table 1 are not only reference-action effects: Trajectory-Context Action Credit also reflects prospective short-horizon progress.

Table 4: Credit-to-progress alignment on 80 ScienceWorld handoff states with Qwen2.5-7B. Mean Spearman ρ measures within-state ranking agreement with forced-rollout ΔBest. Recall@3 measures whether the highest-value action appears among the top three candidates, and Top-1 value capture measures the fraction of the best attainable gain obtained by the highest-scored action. Brackets report 95% hierarchical bootstrap confidence intervals.

Score signal	Mean Spearman ρ	Paired $\Delta\rho$ vs. ReAct	Best-action Recall@3	Top-1 value capture
Random permutation	$0.001\,[-0.047, 0.049]$	$-0.141\,[-0.226, -0.056]$	$0.376\,[0.270, 0.482]$	$0.418\,[0.346, 0.490]$
ReAct score	$0.142\,[0.061, 0.224]$	—	$0.850\,[0.763, 0.919]$	$0.687\,[0.621, 0.748]$
Raw retrieval score	$0.184\,[0.091, 0.276]$	$+0.042\,[-0.031, 0.115]$	$0.900\,[0.821, 0.950]$	$0.672\,[0.608, 0.733]$
TrajCredit	$\mathbf{0.348}\,[\mathbf{0.259}, \mathbf{0.435}]$	$\mathbf{+0.206}\,[\mathbf{0.094}, \mathbf{0.315}]$	$0.938\,[0.868, 0.974]$	$0.724\,[0.662, 0.782]$
Fused score	$\mathbf{0.381}\,[\mathbf{0.298}, \mathbf{0.463}]$	$\mathbf{+0.239}\,[\mathbf{0.128}, \mathbf{0.347}]$	$\mathbf{0.963}\,[\mathbf{0.902}, \mathbf{0.988}]$	$\mathbf{0.758}\,[\mathbf{0.699}, \mathbf{0.811}]$

Table 5: Bounded ScienceWorld intervention. ΔBest is the maximum environment-score improvement within the intervention window, Progress rate is the fraction of handoffs with positive gain, and Invalid reduction is $\mathrm{Invalid}_{\mathrm{base}} - \mathrm{Invalid}_{+\mathrm{TrajCredit}}$. ReAct uses 47, 44, and 55 paired handoffs at horizons 1, 3, and 8; SayCan uses 55 paired handoffs at horizon 8. Brackets report paired 95% hierarchical bootstrap confidence intervals.

Selector	H	ΔBest		Progress rate		Invalid rate	Invalid reduction
		Base / +TrajCredit	Paired Δ	Base / +TrajCredit	Paired Δ	Base / +TrajCredit	Paired Δ
ReAct	1	$6.660/8.426$	$\mathbf{+1.766}\,[\mathbf{0.31}, \mathbf{3.22}]$	$0.532/0.638$	$+0.106\,[-0.025, 0.237]$	$0.000/0.000$	0.000
ReAct	3	$7.909/11.864$	$\mathbf{+3.955}\,[\mathbf{1.15}, \mathbf{6.74}]$	$0.523/0.545$	$+0.022\,[-0.125, 0.168]$	$0.045/0.038$	$+0.007\,[-0.034, 0.049]$
ReAct	8	$8.691/12.182$	$\mathbf{+3.491}\,[\mathbf{1.73}, \mathbf{5.25}]$	$0.364/0.636$	$\mathbf{+0.272}\,[\mathbf{0.118}, \mathbf{0.426}]$	$0.182/0.132$	$\mathbf{+0.050}\,[\mathbf{0.006}, \mathbf{0.095}]$
SayCan	8	$10.236/13.307$	$\mathbf{+3.071}\,[\mathbf{1.29}, \mathbf{4.85}]$	$0.455/0.618$	$\mathbf{+0.163}\,[\mathbf{0.021}, \mathbf{0.305}]$	$0.164/0.109$	$\mathbf{+0.055}\,[\mathbf{0.005}, \mathbf{0.106}]$

5.5 Bounded Environment Intervention

Value alignment matters for control only if it survives action execution. We therefore allow either the Base Selector or its TrajCredit variant to control the environment from the same handoff state for a bounded number of steps. ReAct is evaluated at horizons 1, 3, and 8, with SayCan providing an additional eight-step selector check. Each paired comparison uses identical handoff states and environment dynamics.

Table 5 and Figure 2(b, bottom) show that TrajCredit increases ΔBest at all three ReAct horizons, with paired confidence intervals excluding zero. At eight steps, Progress Rate also improves and Invalid Rate decreases. SayCan shows the same pattern at eight steps, providing a cross-selector check. These results connect the value alignment in Table 4 to improved short-horizon behavior when the credit is used for action selection.

6 Conclusion

We formalized Trajectory-Context Action Credit as a candidate-level interface between trajectory memory and action selection, and introduced TrajCredit as a training-free instantiation. Across controlled ranking, attribution controls, credit-to-progress alignment, and bounded intervention, the resulting credit provides information beyond raw retrieval support, depends on aligned trajectory evidence, and reflects short-horizon environment progress. These results support explicit action credit as a useful decision-facing interface for reusable agent experience. Future work will extend the interface to exploration, search, and longer-horizon control.

7 Limitations

TrajCredit relies on trajectory memory containing evidence relevant to the current Decision State, so its credit may become less informative when memory coverage is sparse. The present study also assumes a finite Candidate Set supplied by the action interface and evaluates downstream effects over bounded horizons. Finally, the implementation uses lightweight lexical retrieval and action matching to keep the credit signal training-free and directly auditable. Extending the interface to dynamically generated action spaces, stronger retrieval mechanisms, and longer-horizon agent control remains future work.

AI USE STATEMENT

Generative AI tools were used to assist with methodology and experimental design, implementation and debugging, analysis and interpretation of experimental results, literature search, and manuscript drafting and editing. All AI-assisted outputs, including code, experimental analyses, citations, and text, were reviewed and verified by the authors. The authors take full responsibility for the methodology, experimental results, claims, and final content of this work.

REFERENCES

Hao Bai, Yifei Zhou, Li Erran Li, Sergey Levine, and Aviral Kumar. Digi-q: Learning VLM q-value functions for training device-control agents. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=CjfQssZtAb.

Shripad Vilasrao Deshmukh, Arpan Dasgupta, Balaji Krishnamurthy, Nan Jiang, Chirag Agarwal, Georgios Theocharous, and Jayakumar Subramanian. Explaining RL decisions with trajectories. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=5Egggz1q575.

Heng Dong, Kefei Duan, and Chongjie Zhang. Enhancing decision-making of large language models via actor-critic. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/dong25c.html.

Yao Fu, Dong-Ki Kim, Jaekyeom Kim, Sungryull Sohn, Lajanugen Logeswaran, Kyunghoon Bae, and Honglak Lee. Autoguide: Automated generation and selection of context-aware guidelines for large language model agents. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/d8efbb5dd415974eb095c3f06bff1f48-Abstract-Conference.html.

Jinglong Gao, Xiao Ding, Lingxiao Zou, Bibo Cai, Bing Qin, and Ting Liu. Expetrans: Llms are experiential transfer learners. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 10577–10616. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.520. URL https://doi.org/10.18653/v1/2025.acl-long.520.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pp. 8154–8173. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.507. URL https://doi.org/10.18653/v1/2023.emnlp-main.507.

Mengkang Hu, Tianxing Chen, Qiguang Chen, Yao Mu, Wenqi Shao, and Ping Luo. Hiagent: Hierarchical working memory management for solving long-horizon agent tasks with large language model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 32779–32798. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.1575. URL https://doi.org/10.18653/v1/2025.acl-long.1575.

Brian Ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander T Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell

Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Nikhil J. Joshi, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. Do as I can, not as I say: Grounding language in robotic affordances. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 287–318. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/ichter23a.html.

Yifei Li, Heng Wang, Lingling Zhang, Muye Huang, Xinyu Zhang, Jiashuai Liu, Hang Yan, and Rongman Xu. Beyond retrieval: Query-conditioned reuse of long-horizon agent trajectories, 2026. URL https://arxiv.org/abs/2608.12847.

Zongyu Lin, Yao Tang, Xingcheng Yao, Da Yin, Ziniu Hu, Yizhou Sun, and Kai-Wei Chang. QLASS: boosting language agent inference via q-guided stepwise search. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/lin25l.html.

Yitao Liu, Chenglei Si, Karthik R. Narasimhan, and Shunyu Yao. Contextual experience replay for self-improvement of language agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 14179–14198. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.694. URL https://doi.org/10.18653/v1/2025.acl-long.694.

Zhihan Liu, Hao Hu, Shenao Zhang, Hongyi Guo, Shuqi Ke, Boyi Liu, and Zhaoran Wang. Reason for future, act for now: A principled architecture for autonomous LLM agents. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 31186–31261. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/liu24ab.html.

Yecheng Jason Ma, Joey Hejna, Chuyuan Fu, Dhruv Shah, Jacky Liang, Zhuo Xu, Sean Kirmani, Peng Xu, Danny Driess, Ted Xiao, Osbert Bastani, Dinesh Jayaraman, Wenhao Yu, Tingnan Zhang, Dorsa Sadigh, and Fei Xia. Vision language models are in-context value learners. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=friHAl5ofG.

Bodhisattwa Prasad Majumder, Bhavana Dalvi Mishra, Peter A. Jansen, Oyvind Tafjord, Niket Tandon, Li Zhang, Chris Callison-Burch, and Peter Clark. CLIN: A continually learning language agent for rapid task adaptation and generalization. *CoRR*, abs/2310.10134, 2023. doi: 10.48550/ARXIV.2310.10134. URL https://doi.org/10.48550/arXiv.2310.10134.

Alexander Pritzel, Benigno Uria, Sriram Srinivasan, Adrià Puigdomènech Badia, Oriol Vinyals, Demis Hassabis, Daan Wierstra, and Charles Blundell. Neural episodic control. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2827–2836. PMLR, 2017. URL http://proceedings.mlr.press/v70/pritzel17a.html.

Haofu Qian, Chenjia Bai, Jiatao Zhang, Fei Wu, Wei Song, and Xuelong Li. Discriminator-guided embodied planning for LLM agent. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=TjP1d8PP8l.

Gabriel Sarch, Lawrence Jang, Michael J. Tarr, William W. Cohen, Kenneth Marino, and Katerina Fragkiadaki. VLM agents generate their own memories: Distilling experience into embodied programs of thought. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/8ac50fd0a4eeeb1f077b17bb7c5353c3-Abstract-Conference.html.

Vishnu Sarukkai, Zhiqiang Xie, and Kayvon Fatahalian. Self-generated in-context examples improve LLM agents for sequential decision-making tasks. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/5d1f02132ef51602adf07000ca5b6138-Abstract-Conference.html.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652. Curran Associates, Inc., 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1b44b878bb782e6954cd888628510e90-Paper-Conference.pdf.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew J. Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=0IOX0YcCdTn.

Yifan Song, Da Yin, Xiang Yue, Jie Huang, Sujian Li, and Bill Yuchen Lin. Trial and error: Exploration-based trajectory optimization of LLM agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7584–7600, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.409. URL https://aclanthology.org/2024.acl-long.409/.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *Trans. Mach. Learn. Res.*, 2024. URL https://openreview.net/forum?id=ehfRiF0R3a.

Ruoyao Wang, Peter A. Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pp. 11279–11298. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.EMNLP-MAIN.775. URL https://doi.org/10.18653/v1/2022.emnlp-main.775.

Zihao Wang, Shaofei Cai, Guanzhou Chen, Anji Liu, Xiaojian Ma, and Yitao Liang. Describe, explain, plan and select: Interactive planning with llms enables open-world multi-task agents. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/6b8dfb8c0c12e6fafc6c256cb08a5ca7-Abstract-Conference.html.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/wang25bx.html.

Yufei Xiang, Yiqun Shen, Yeqin Zhang, and Cam-Tu Nguyen. Retrospex: Language agent meets offline reinforcement learning critic. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4650–4666, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.268. URL https://aclanthology.org/2024.emnlp-main.268/.

Zidi Xiong, Yuping Lin, Wenya Xie, Pengfei He, Zirui Liu, Jiliang Tang, Himabindu Lakkaraju, and Zhen Xiang. How memory management impacts LLM agents: An empirical study of experience-following behavior. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States,*

July 2-7, 2026, pp. 623–645. Association for Computational Linguistics, 2026. doi: 10.18653/V1/2026.ACL-LONG.27. URL https://doi.org/10.18653/v1/2026.acl-long.27.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023a. URL http://papers.nips.cc/paper_files/paper/2023/hash/271db9922b8d1f4dd7aaef84ed5ac703-Abstract-Conference.html.

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 *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023b. URL https://openreview.net/forum?id=WE_vluYUL-X.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: LLM agents are experiential learners. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2024, February 20-27, 2024, Vancouver, Canada*, pp. 19632–19642. AAAI Press, 2024. doi: 10.1609/AAAI.V38I17.29936. URL https://doi.org/10.1609/aaai.v38i17.29936.

Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: Trajectory-as-exemplar prompting with memory for computer control. In *International Conference on Learning Representations*, volume 2024, pp. 19036–19066, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/52f050499cf82fa8efb588e263f6f3a7-Paper-Conference.pdf.

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 *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62138–62160. PMLR / OpenReview.net, 2024a. URL https://proceedings.mlr.press/v235/zhou24r.html.

Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. Archer: Training language model agents via hierarchical multi-turn RL. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62178–62209. PMLR / OpenReview.net, 2024b. URL https://proceedings.mlr.press/v235/zhou24t.html.

Yuchen Zhuang, Xiang Chen, Tong Yu, Saayan Mitra, Victor S. Bursztyn, Ryan A. Rossi, Somdeb Sarkhel, and Chao Zhang. Toolchain*: Efficient action space navigation in large language models with a* search. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=B6pQxqUcT8.

APPENDIX

A EVALUATION DETAILS

This appendix provides additional details on the controlled evaluation protocol, information availability, and implementation settings used in the main experiments. Trajectory memory, development episodes, and test episodes are disjoint at the episode level.

A.1 BENCHMARKS, TASKS, AND EVALUATION CONSTRUCTION

We evaluate on the same ScienceWorld and ALFWorld settings introduced in Section 5.1. ScienceWorld is a text-based interactive environment built around procedural science tasks. Our evaluation covers eight tasks spanning physical state changes, measurement, and plant growth: Boil, Melt,

Freeze, Change the State of Matter, Use a Thermometer, Measure Melting Point, Grow a Plant, and Grow a Fruit. These tasks require multi-step interaction and provide procedural structure over which history and stage information can vary.

ALFWorld provides text-based household interaction grounded in embodied household environments. We use valid-unseen tasks from six task types: Pick & Place, Pick Two & Place, Examine in Light, Clean & Place, Heat & Place, and Cool & Place. The controlled ALFWorld evaluation contains 133 games and 786 extracted Decision States. Relative to ScienceWorld, these tasks place greater emphasis on local object state and manipulation dependencies. Table 6 summarizes the two controlled evaluation settings.

Table 6: Benchmark coverage for controlled candidate-action evaluation.

Benchmark	Evaluation source	Coverage	Controlled construction
ScienceWorld	held-out episodes	8 tasks	16-candidate Decision States
ALFWorld	valid-unseen	6 task types	133 games, 786 Decision States

Each controlled Decision State is extracted before the reference action is executed. Candidate Sets are constructed before scoring and shared across each paired comparison. In controlled ScienceWorld ranking, each set contains the reference action and 15 distractors sampled under the same admissible-action construction. For ALFWorld, Candidate Sets are constructed from the admissible action space provided at each Decision State, following the benchmark interaction protocol. The candidate-set size is therefore environment-dependent and follows the available actions at the corresponding state. The same candidate set is used for all paired comparisons under each Decision State. The retrieval-confusing control preserves the Decision State, reference action, and Candidate Set size while replacing ordinary distractors with incorrect actions that receive high raw retrieval support.

A.2 Action-Centered Trajectory Memory and Stage-Proxy Provenance

Trajectory memory is constructed from the episode-disjoint memory partition, with each stored action-centered slice represented as

$$m_i = (g_i, o_i, h_i, a_i, p_i, \Delta_i^{\mathrm{imm}}, \Delta_i^{\mathrm{fut}}).$$

In ScienceWorld, the stage proxy is the normalized cumulative task score available after the preceding transition,

$$p_t = \mathrm{info}_{t-1}[\texttt{score}]/100. \tag{24}$$

The memory value p_i uses the same definition at the corresponding memory state before action a_i is executed. This quantity is available before the current action is selected. We do not use hidden simulator state, oracle-plan position, or future information from the evaluation episode.

ALFWorld does not expose a comparable observable stage signal. We therefore set $\psi_i(t) = 1$, which disables stage modulation while preserving the same candidate-credit interface. When $\psi_i(t) = 1$, the procedural-stage score reduces to the local support score.

For the outcome terms, let r_{i-1} and r_i denote the environment scores immediately before and after reference action i, and let R_T denote the final score of the stored reference trajectory. For ScienceWorld, we define

$$\Delta_i^{\mathrm{imm}} = r_i - r_{i-1}, \tag{25}$$

$$\Delta_i^{\mathrm{fut}} = \max(0, R_T - r_{i-1}). \tag{26}$$

The corresponding progress signals are normalized using statistics from the trajectory-memory split before being used for candidate credit computation.

A.3 Protocol-Aligned Base Selectors and Implementation

We instantiate ReAct, SayCan, Reflexion, ExpeL, and CLIN within the common candidate-ranking interface used in the main experiments. ReAct provides an LM-based candidate score. SayCan combines the LM score with action-support evidence. Reflexion, ExpeL, and CLIN condition their Base Selectors on retrieved experience. TrajCredit is composed with each selector through the same score-fusion interface without changing its underlying decision mechanism. Each paired comparison uses the same Decision State, Candidate Set, model budget, trajectory pool, and observable environment information.

Unless otherwise noted, we retain the most recent eight interaction steps, retrieve $N = 64$ memory slices, and set $\sigma_p = 0.25$ for the ScienceWorld stage-consistency kernel. Retrieval uses TF–IDF similarity over the task, observation, and recent history, while action similarity uses character n-gram cosine similarity. One global fusion coefficient λ is selected for each selector family using the development split and is frozen before test evaluation. Controlled ranking results are averaged over three seeds. Confidence intervals in the main statistical comparisons use paired hierarchical bootstrap resampling over tasks and Decision States with 10,000 replicates.

TrajCredit requires no additional model training or LM forward pass beyond the underlying Base Selector. Its additional computation consists of trajectory retrieval and candidate-credit scoring. Across 1,179 decisions from the eight ScienceWorld tasks, the scorer-only computation requires 99.945 ms per decision on average, corresponding to 6.247 ms per candidate. The median and 95th percentile of task-level mean latencies are 71.856 ms and 172.971 ms, respectively.

B Extended Controlled-Ranking Results

The main paper reports the complete controlled-ranking matrix in terms of Top-1 accuracy. Table 7 re-expresses the same paired evaluations as absolute Top-1 gains from adding TrajCredit. This view makes the consistency of the plug-in effect across Base Selectors, benchmarks, and model backbones directly visible.

Table 7: Absolute Top-1 gain from adding TrajCredit to each Base Selector in the controlled-ranking evaluations. Each entry is $\text{Top-1}_{\text{Base+TrajCredit}} - \text{Top-1}_{\text{Base}}$.

Base Selector	SW 3B	SW 7B	SW Llama-8B	ALF 3B	ALF 7B
ReAct	+0.3056	+0.2531	+0.2715	+0.2980	+0.2598
SayCan	+0.0440	+0.0377	+0.0404	+0.0551	+0.0359
Reflexion	+0.1281	+0.0173	+0.0860	+0.1601	+0.0751
ExpeL	+0.1167	+0.0902	+0.1268	+0.1389	+0.0918
CLIN	+0.1438	+0.0717	+0.0991	+0.1993	+0.1057

All 25 Base Selector and evaluation-setting pairs show a positive Top-1 gain. The effect is largest for ReAct, where trajectory credit supplements a purely LM-based ranking signal, while positive gains remain for selectors that already use score fusion or retrieved experience. This is consistent with the main result that explicit candidate-level trajectory credit is complementary to different underlying decision mechanisms.

C Robustness and Sensitivity

The main paper establishes that aligned trajectory memory and the three candidate-credit views contribute to controlled action ranking. We provide two additional analyses of the same credit interface. We first examine sensitivity to the fusion coefficient on the development split used for coefficient selection. We then evaluate Outcome-Calibrated Evidence under trajectory memories with different levels of reliability.

Table 8 shows the development-set sensitivity of the fusion weight. Performance changes smoothly over the evaluated range. The coefficient used for held-out test evaluation is selected through this development procedure and is not tuned separately for a test task or seed.

Table 8: Development-set sensitivity to the fusion coefficient λ under a representative ScienceWorld Qwen2.5-7B setting.

λ	0.5	1	2	4	8	16
Top-1	0.3436	0.4138	0.4495	0.4757	0.4865	0.5013

We next evaluate the outcome view when the trajectory pool contains less uniformly reliable experience. The high-quality condition uses the original aligned memory pool. The mixed and noisy conditions replace 25% and 50% of the stored slices with corrupted slices, respectively. The total number of memory slices is held fixed within each matched task-seed unit. All conditions use the same held-out Decision States.

Table 9: Memory-quality robustness on ScienceWorld with Qwen2.5-7B using ReAct over 24 matched task-seed units in the fixed-capacity robustness evaluation. Results are Top-1 mean $\pm$ standard deviation. Memory capacity is fixed across corruption conditions.

Memory condition	Corrupted slices	TrajCredit	w/o outcome	Δ
High-quality	0%	0.5371 ± 0.1485	0.4938 ± 0.2107	+0.0433
Mixed-quality	25%	0.4893 ± 0.1652	0.4446 ± 0.2098	+0.0447
Noisy	50%	0.4462 ± 0.1770	0.3878 ± 0.2005	+0.0584

Performance decreases for both variants as the fraction of corrupted trajectory slices increases. Full TrajCredit remains above the variant without Outcome-Calibrated Evidence in all three conditions. The mean advantage grows from 0.0433 with aligned memory to 0.0584 at 50% corruption. Because memory capacity is fixed, this comparison isolates memory reliability from the amount of stored experience.

D Extended Value and Intervention Analysis

The main paper evaluates credit-to-progress alignment on 80 held-out ScienceWorld handoff states. Each state uses the same score-independent set of eight candidates. Each candidate is force-executed once and followed by the same frozen three-step ReAct continuation, which does not use the score signal under evaluation. Table 10 re-expresses the same 80-state evaluation relative to raw retrieval support.

Table 10: Credit-to-progress alignment on the 80 ScienceWorld handoff states. The difference column uses raw retrieval as the reference. Recall@3 and value capture follow the definitions in the main paper.

Score signal	Mean ρ	$\Delta\rho$ vs. retrieval	Recall@3	Value capture
ReAct score	0.142	-0.042	0.850	0.687
Raw retrieval	0.184	—	0.900	0.672
TrajCredit	0.348	+0.164	0.938	0.724
Fused score	0.381	+0.197	0.963	0.758

TrajCredit increases mean rank alignment with forced-rollout ΔBest by 0.164 relative to raw retrieval, while the fused score increases it by 0.197. The corresponding Recall@3 and value-capture results show that the stronger rank alignment is accompanied by better coverage and selection of high-value candidate actions. This extends the main retrieval control from reference-action ranking to independently measured Short-Horizon Environment Value.

The main bounded-intervention results report aggregate changes in ΔBest, progress rate, and invalid-action rate. Table 11 reports the paired win, tie, and loss distribution for ReAct, which is evaluated at all three intervention horizons.

Table 11: Paired bounded-intervention consistency for ReAct on ScienceWorld. Wins, ties, and losses compare ΔBest for TrajCredit against the Base Selector on the same eligible handoff states.

Horizon	Cases	Wins / ties / losses	Mean Δ Best
1	47	5 / 40 / 2	+1.766
3	44	14 / 20 / 10	+3.955
8	55	24 / 23 / 8	+3.491

The paired distribution complements the aggregate intervention results in the main paper. Shorter horizons contain more ties, consistent with many short interventions producing no measurable change in ΔBest. At the eight-step horizon, positive paired differences are more frequent than negative ones.