LEVERAGING INTERACTIONS AND SEQUENTIAL DEPENDENCIES FOR EFFECTIVE CREDIT ASSIGNMENT IN SEARCH AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Search agents answer questions by iteratively issuing queries, retrieving evidence, and reasoning over the results. A key research question is how to assign informative scores to intermediate search steps without relying on an additional critic model. Our case studies highlight two factors that such scores should capture: interaction effects, where multiple pieces of evidence jointly support an answer, and sequential dependencies, where earlier search results guide later queries. We propose *Sequential Shapley*, a fine-grained credit assignment method for multi-turn search agents that accounts for both effects. It evaluates each subset of search steps through a continuation-induced closure within the observed trajectory. The resulting coalition value models sequential dependencies through expected future contributions, while Shapley attribution captures evidence interactions by aggregating marginal contributions across coalitions. Our theoretical analysis relates Sequential Shapley to standard Shapley values and supports an approximate backward-redistribution interpretation. We integrate these step-level rewards with GRPO to obtain **SPGRPO**. On seven search-augmented QA benchmarks, SPGRPO improves average success rates over the strongest baseline by 2.8 and 1.4 percentage points for Qwen2.5-3B and 7B, respectively. Ablations further support the benefit of jointly modeling evidence interactions and search dependencies in this setting.

1 INTRODUCTION

Large Language Models (LLMs) have rapidly evolved from static text generators into decision-making agents that interact with external environments through multi-step reasoning and tool use Jin et al. (2025); Zheng et al. (2025a;b); Yao et al. (2022); Shridhar et al. (2020). In particular, search agents tackle search-augmented question answering by interleaving reasoning with queries to external search tools Jin et al. (2025); Zheng et al. (2025a;b). In this setting, reasoning becomes a sequential process of acquiring and integrating evidence, where the results of earlier searches shape subsequent queries and ultimately support the final answer.

Reinforcement learning (RL) has become a key paradigm for enhancing the reasoning capabilities of LLMs Shao et al. (2024); Zheng et al. (2025c); Hu et al. (2025); Ahmadian et al. (2024); Schulman et al. (2017). In particular, outcome-reward RL methods Shao et al. (2024); Zheng et al. (2025c); Hu et al. (2025); Ahmadian et al. (2024) have been widely adopted due to their simplicity, scalability, and critic-free design. However, when directly applied to search agents, such trajectory-level supervision fails to identify which intermediate search steps are useful, redundant, or harmful. As a result, they may reinforce redundant actions in successful trajectories while overlooking useful exploratory steps in failed ones. This limitation motivates fine-grained credit assignment for intermediate tool-use actions.

A search step is not merely a local operation with isolated utility; it is an *epistemic action* that acquires evidence and reshapes the agent's knowledge state. This evidence-acquisition process creates two coupled credit-assignment challenges, as illustrated in Figure 1: (1) **interaction effects**, and (2) **sequential dependencies** among steps. First, step contributions are often interactional. In comparative or multi-hop reasoning, a single piece of evidence may be insufficient in isolation, while several evidence pieces jointly unlock the final inference. For example, neither Alice's nor Bob's

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.

birthday alone suffices to answer the comparison question, but together they determine the answer. The second search allows the model to determine the answer, producing a larger increase in the likelihood of the correct answer. Rewards based on information gain under the current prefix may favor the second search, even though both pieces of evidence are necessary. Second, search steps are sequentially dependent: earlier observations guide the formulation of subsequent queries by revealing relevant entities, constraints, or hypotheses. For example, identifying France as the 2018 FIFA World Cup winner guides a search for its capital, which in turn identifies Paris as the target of the final population query. An early search may thus provide little direct evidence for the final answer yet deserve credit for enabling useful subsequent searches. Additional case studies are provided in Appendix D. However, existing fine-grained reward allocation methods often rely on local or indirect estimates of step contributions. IGPO Wang et al. (2025) and AEPO Dong et al. (2025) measure information gain or uncertainty reduction under the prefix context, while GiGPO Feng et al. (2025) estimates credit by regrouping trajectories with aligned intermediate states. Although these methods provide finer-grained supervision than outcome-level rewards, they do not explicitly account for higher-order interactions among multiple search steps. Shapley-based attribution Shapley (1953) offers a principled way to capture evidence interactions by aggregating marginal contributions across subsets of search steps. However, the standard Shapley formulation treats coalitions as unordered sets of players and fails to explicitly model the sequential dependencies among search steps.

To address this challenge, we propose *Sequential Shapley*, a novel credit assignment framework that extends the classical Shapley value to multi-step search agents. The key idea is to redesign the value function by reinterpreting each subset of actions not as an unordered coalition, but as inducing a *future continuation closure* under trajectory dynamics. Specifically, the value of a subset is defined as the expected return over valid continuations composed of later actions from the same observed rollout, where a step receives credit not only for its direct contribution, but also for useful future contributions it makes reachable. Under this formulation, the Shapley value naturally retains its ability to capture higher-order interaction effects among steps, while the modified value function explicitly encodes sequential dependencies through continuation-induced closures. The resulting step-level rewards are combined with GRPO Shao et al. (2024), thereby yielding our **SPGRPO** (**S**equential Sha**p**ley-enhanced **GRPO**) algorithm, a critic-free policy optimization algorithm for LLM search agents. Theoretical analysis shows that Sequential Shapley admits an approximate interpretation as an *explainable backward redistribution* of the standard Shapley value: Credit from useful downstream steps is propagated backward to earlier steps that increase their reachability. Extensive comparisons with seven baselines and comprehensive ablation studies demonstrate the effectiveness of our method and validate the importance and effectiveness of jointly modeling step interactions and sequential dependencies. We further propose an efficient Prefix-Reuse algorithm that reduces the computational complexity of Sequential Shapley to match that of standard Shapley. In practice, this leads to only a $+35.0\%$ increase in overall training time compared to GRPO while requiring no external critic model, and thus preserving efficiency and simplicity. Our contributions can be summarized as follows:

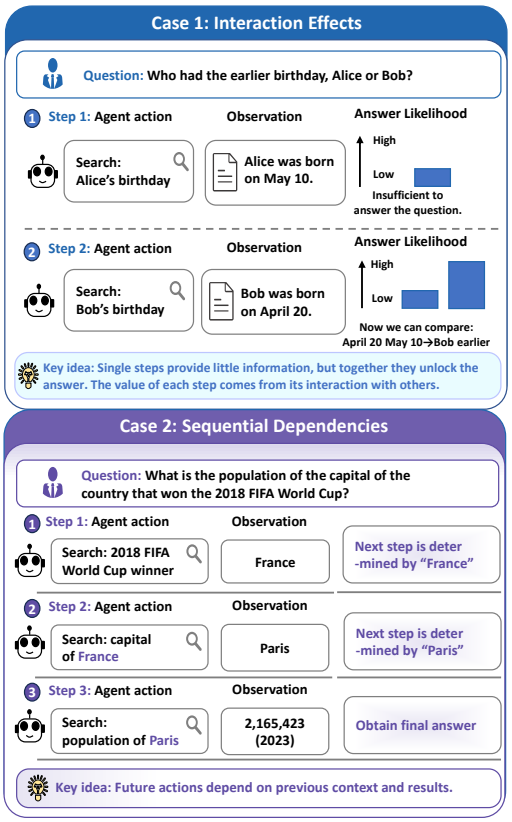

Figure 1: Two case studies illustrating the importance of modeling interaction effects and sequential dependencies in multi-step search agents.

- We propose Sequential Shapley, a novel, scalable and computationally tractable fine-grained credit assignment framework for multi-step LLM agents. By redesigning the value function via

continuation-induced closures, our method extends the classical Shapley value to sequential decision-making scenarios while preserving its interaction modeling capability.

- We provide theoretical analysis showing that Sequential Shapley admits an explainable backward redistribution of the standard Shapley value, which propagates contributions from future steps to earlier ones, offering a principled way to model sequential dependencies in Shapley-like credit assignment.
- We conduct extensive experiments on 7 search-augmented QA benchmarks. Our method broadly outperforms the strongest baselines, achieving average improvements of +2.8 (3B scale) and +1.4 (7B scale), and ablation studies further validate the effectiveness of jointly modeling interaction effects and sequential dependencies.

2 PRELIMINARIES

2.1 SEARCH-AGENT ROLLOUT SETUP

Let $\mathcal{D} = \{(q, \text{ans})\}$ denote a dataset of question–answer pairs, and let $\mathcal{E}$ represent an external search engine. The goal of the search agent is to answer a question q by generating a rollout $o = (\tau_1, \ldots, \tau_L, \hat{\text{ans}})$ through iterative interaction with the search engine $\mathcal{E}$, where L is the total number of search steps. The agents behavior is governed by a policy π_θ, parameterized by a large language model, which defines a distribution over search trajectories conditioned on the question: $\pi_\theta(o \mid q) = \prod_{l=1}^{L} \pi_\theta(\tau_l \mid \tau_{<l}, q)$. Each search step τ_l consists of three components:

$$\tau_l = (\texttt{<think>}_l, \texttt{<search>}_l, \texttt{<information>}_l).$$

Here, $\texttt{<think>}_l$ denotes the intermediate reasoning, and $\texttt{<search>}_l$ denotes the search query issued to $\mathcal{E}$. Both components are generated by the policy π_θ, while $\texttt{<information>}_l$ denotes the retrieved evidence returned by $\mathcal{E}$ in response to that query. After L search steps, the agent produces a rationale followed by the predicted answer $\hat{\text{ans}}$. Then a scalar reward $R(o)$ is assigned based on the correctness of the final answer. In this setting, supervision is typically sparse and only provided at the trajectory level, which can hinder effective policy optimization for multi-step search agents.

2.2 SHAPLEY VALUE

The Shapley value Shapley (1953) is a classical method from cooperative game theory for fairly attributing the total value of a set of players to individual participants. Given the player set $N = \{1, 2, \cdots, n\}$, a player coalition $S \subseteq N$ and a value function $V : 2^N \to \mathbb{R}$, the Shapley value of player $i \in N$ can be written as

$$\phi_i = \sum_{S \subseteq N} \Phi(i, S) V(S), \text{where } \Phi(i, S) = \begin{cases} \frac{(|S|-1)!(|N|-|S|)!}{|N|!}, & i \in S, \\ -\frac{|S|!(|N|-|S|-1)!}{|N|!}, & i \notin S. \end{cases}$$

Here, $\Phi(i, S)$ gives the signed weight of coalition value $V(S)$ in the Shapley value of player i. While still facing challenges, the Shapley value provides a promising way to decompose the trajectory-level reward into contributions from individual steps fairly.

3 SEQUENTIAL SHAPLEY VALUE ENHANCED POLICY OPTIMIZATION

3.1 PROBLEM DEFINITION: WITHIN-TRAJECTORY CREDIT ASSIGNMENT

For a question q, let o be a rollout generated by policy π_θ through interaction with environment $\mathcal{E}$. Let n_o denote the number of realized search actions in o, and let $N_o = \{1, \ldots, n_o\}$ be their position-indexed player set. We write $A_{N_o} = (a_i)_{i \in N_o} = (a_1, \ldots, a_{n_o})$ for the corresponding ordered action sequence, with e_i denoting the environmental response to a_i. We use an *action-centric* definition: a_i is the exact search-tool call generated by the policy. Unlike a *state-centric* alternative, we do not collapse different calls that retrieve identical or semantically similar evidence. This is consistent with the standard MDP abstraction, which distinguishes an action from the environment transition

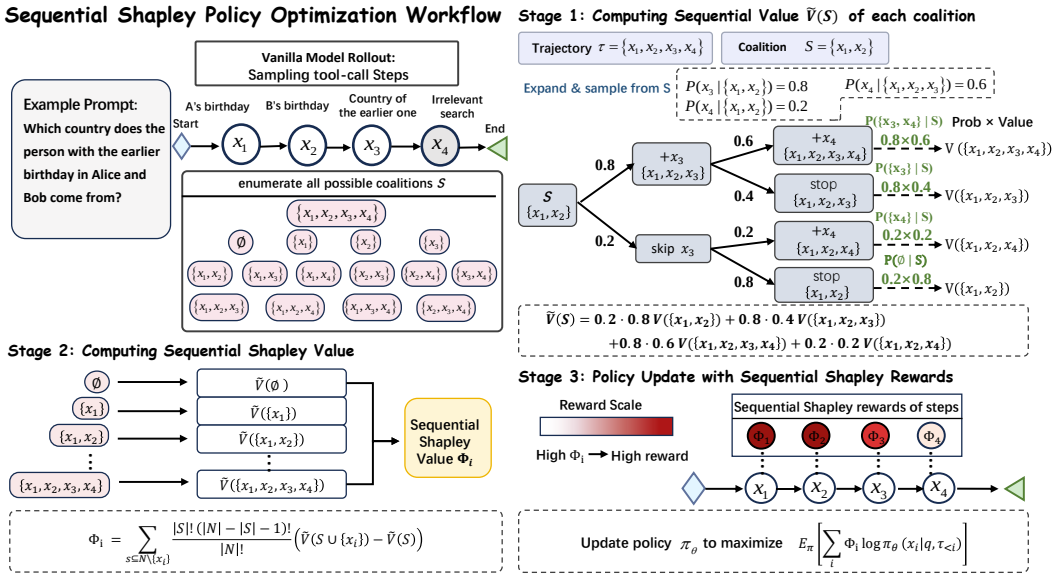

Figure 2: Overview of the proposed Sequential Shapley-enhanced Policy Optimization framework. Given a multi-step tool-use trajectory, we first enumerate candidate coalitions S and estimate their sequential values by averaging over trajectory-supported future closures, i.e., $\widetilde{V}(S) = \mathbb{E}_{C \sim P(C|S)}[V(S \cup C)]$, where $P(C \mid S)$ assigns probability to inclusion patterns of later steps from the same observed rollout (Stage 1). The resulting values are then used to compute Sequential Shapley rewards, which assign fine-grained credit to individual tool-use steps (Stage 2). Finally, these step-level rewards guide policy optimization by reinforcing steps with higher Sequential Shapley values (Stage 3).

and resulting observation it induces Sutton and Barto (2018). Thus, different search queries remain distinct actions even when their responses e_i contain similar evidence.

Each rollout induces a trajectory-conditioned cooperative game $\mathcal{G}_o = (N_o, V_o)$, where $V_o : 2^{N_o} \to \mathbb{R}$ is the value function. Player i corresponds to action a_i at position i. Final-answer generation is excluded from N_o and instead receives a separate correctness-based signal. This per-rollout construction matches our objective of distinguishing useful from unhelpful search steps within each realized trajectory. Different futures induced by the policy are represented through GRPO's outer rollout sampling, while credit assignment remains within each sampled trajectory.

3.2 Joint Modeling of Interactions and Sequential Dependencies

Definition 1 (Sequential Shapley Value)**.** *Let $N = \{1, 2, \cdots, n\}$ be the player set, and let $A_N = (a_i)_{i \in N} = (a_1, a_2, \cdots, a_n)$ be the ordered action sequence associated with these players. For each coalition $S \subseteq N$, let $A_S = (a_i)_{i \in S}$, with indices in ascending order, denote the corresponding action subsequence. When such a subsequence is used as model context, its corresponding environmental responses are included. Let $V : 2^N \to \mathbb{R}$ be a base value function, where $V(S)$ is the value of coalition S evaluated using A_S and its corresponding environmental responses. For any coalition $S \subseteq N$, let $m(S) = \max S$ denote its latest player index, with $m(\emptyset) = 0$, and define the set of valid future continuations of S as*

$$\mathcal{C}(S) = \{C : C \subseteq \{m(S) + 1, \ldots, n\}\}.$$

Let $P(C \mid S)$ be a continuation distribution over $\mathcal{C}(S)$, representing the probability that action subsequence A_S induces the future action subsequence A_C. We require

$$P(C \mid S) \in [0, 1], \qquad \sum_{C \in \mathcal{C}(S)} P(C \mid S) = 1.$$

The sequentially closed value of S is defined as

$$\widetilde{V}(S) = \mathbb{E}_{C\sim P(\cdot|S)}\left[V(S\cup C)\right] = \sum_{C\in\mathcal{C}(S)} P(C \mid S)V(S\cup C).$$

Then, the Sequential Shapley Value of player $i \in N$ is defined as

$$\phi_i^{\text{seq}} = \sum_{S\subseteq N} \Phi(i,S)\widetilde{V}(S).$$

We first introduce a general formulation of the Sequential Shapley Value, deferring the specific choices of $V(S)$ and $P(C \mid S)$ to the next section. In principle, this formulation applies to a broad class of sequential decision-making problems with ordered players; here, we focus on search agents.

Theorem 1 (Shapley Axioms)**.** *Sequential Shapley Value satisfies all standard Shapley axioms on the induced value function $\tilde{V}$, including: Efficiency: $\sum_{i\in N}\phi_i^{\text{seq}} = \tilde{V}(N) - \tilde{V}(\emptyset)$; Symmetry: if $\tilde{V}(S\cup\{i\}) = \tilde{V}(S\cup\{j\})$ for all $S \subseteq N\setminus\{i,j\}$, then $\phi_i^{\text{seq}} = \phi_j^{\text{seq}}$; Dummy: if $\tilde{V}(S\cup\{i\})-\tilde{V}(S) = 0$ for all $S \subseteq N \setminus \{i\}$, then $\phi_i^{\text{seq}} = 0$; Additivity: $\phi_i^{\text{seq}}(\tilde{V}_1+\tilde{V}_2) = \phi_i^{\text{seq}}(\tilde{V}_1) + \phi_i^{\text{seq}}(\tilde{V}_2)$.* [1]

Furthermore, the following theorems establish a formal connection between Sequential Shapley value and the standard Shapley value.

Theorem 2 (Sequential Shapley Decomposition)**.** *Let the value function be decomposed as $V(S) = V(\emptyset)+\sum_{k\in S}\phi_{\text{standard}}(k)+R(S)$, where $R(S)$ captures the non-linear residual deviation between $V(S)$ and the additive surrogate induced by the standard Shapley values. Then we have* [1]

$$\phi_{\text{seq}} = (I+B)\phi_{\text{standard}} + \Delta,$$

where $I \in \mathbb{R}^{n\times n}$ is the identity matrix, $B \in \mathbb{R}^{n\times n}$ is induced by the continuation distribution, where $B_{i,j} = \sum_S \Phi(i,S)\sum_C P(C \mid S)\,\mathbf{1}_{j\in C}$, and the residual term Δ is given by

$$\Delta_i = \sum_S \Phi(i,S)\sum_C P(C \mid S)\,R(S\cup C).$$

Corollary 1 (Residual Estimation)**.** *Assume that $\exists M \geq 0$ such that $|R(S)| \leq M$ for all $S \subseteq N$, then the residual is bounded by* [1] *$|\sum_i \Delta_i| \leq 2M$ and $\sum_i |\Delta_i| \leq \pi\sqrt{n}\,M$.*

Theorem 2 and Corollary 1 show that Sequential Shapley can be approximately interpreted as a linear redistribution of the standard Shapley values. Under a mild monotonicity assumption that adding an earlier step does not reduce the probability of inducing later valid continuations, this redistribution is directional: future-step credits are propagated backward to earlier steps. This property matches the motivation of Sequential Shapley, since earlier tool calls may enable later useful actions and should therefore receive part of their future contributions. This perspective is also analogous to generalized advantage estimation (GAE) Schulman et al. (2015) in traditional RL, where future advantage-related signals are propagated backward to the current step with exponentially decayed weights $(\gamma\lambda)^{k-t}$. Instead of relying on a hand-designed temporal decay factor, our backward redistribution coefficient has a concrete probabilistic interpretation. We provide a supplementary explanation in Appendix E.5.

3.3 Instantiation in Search-Agent Scenarios

In practice, for a coalition $S \subseteq N$, the base value function $V(S)$ is implemented as the information gain Wang et al. (2025), which measures the model's probability of producing the correct final answer ans given the selected action subsequence A_S and its corresponding tool outputs. Specifically, we preserve only these tool outputs while masking all other tool uses through prompt injection. The value $V(S)$ is then defined as follows:

$$V(S) = \exp\left(\frac{1}{L}\sum_{j=1}^{L}\log p_\theta(\text{ans}_j \mid q, A_S, \text{ans}_{<j})\right), \tag{1}$$

[1] Proofs can be found in Appendix E.

where $p_\theta(\text{ans}_j \mid q, A_S, \text{ans}_{<j})$ is the probability of generating token ans_j conditioned on the question q, the preserved action subsequence A_S, and previously generated answer tokens $\text{ans}_{<j}$.

For the continuation distribution $P(C \mid S)$, we instantiate it as a sequential Bernoulli selection process over future actions. Let $C = \{c_1, \ldots, c_r\} \in \mathcal{C}(S)$, where $m(S) < c_1 < \cdots < c_r \leq n$, be a coalition of future players, and let $A_C = (a_{c_1}, \ldots, a_{c_r})$ be its corresponding ordered action subsequence. For each $k > m(S)$, define $C_{<k} = \{j \in C : j < k\}$. We define $P(C \mid S)$ as the probability of sequentially selecting the actions indexed by C and skipping the other future actions:

$$P(C \mid S) = \prod_{k=m(S)+1}^{n} \left[\pi_\theta(a_k \mid q, A_S, A_{C_{<k}})\right]^{\mathbf{1}[k \in C]} \left[1 - \pi_\theta(a_k \mid q, A_S, A_{C_{<k}})\right]^{\mathbf{1}[k \notin C]} \quad (2)$$

Under the action-centric definition, the token probabilities have a direct interpretation. By autoregressive factorization, the conditional probability of reproducing the observed action a_k is

$$\pi_\theta^{\text{raw}}(a_k \mid q, A_S, A_{C_{<k}}) = \left(\prod_{\ell=1}^{|a_k|} p_\theta(a_{k,\ell} \mid q, A_S, A_{C_{<k}}, a_{k,<\ell})\right) p_\theta(\text{eos} \mid q, A_S, A_{C_{<k}}, a_k).$$

Thus, the raw token product represents the policy probability of reaching the exact observed future action from the preserved action–state context. However, this product decreases systematically with the number of tokens and therefore biases continuation probabilities against longer actions. To reduce this length dependence, we geometrically average all factors in the raw action probability and define the effective Bernoulli probability as

$$\pi_\theta(a_k \mid q, A_S, A_{C_{<k}}) := p\left[\pi_\theta^{\text{raw}}(a_k \mid q, A_S, A_{C_{<k}})\right]^{1/(|a_k|+1)},$$

where $p \in (0, 1]$ is a global correction factor shared across datasets. Appendix B reports a re-rollout calibration check for this surrogate. Lemma 1 shows that $P(C \mid S)$ is well-defined, and Figure 2 provides an intuitive illustration.

Lemma 1 (Validity of Continuation Distribution)**.** *For any coalition $S \subseteq N$, $P(C \mid S)$ satisfies* [1]

$$P(C \mid S) \in [0, 1], \qquad \sum_{C \in \mathcal{C}(S)} P(C \mid S) = 1.$$

After obtaining $\tilde{V}(S)$, we compute the Sequential Shapley rewards according to Definition 1. These rewards are used as step-level signals and combined with the correctness reward to construct the final training objective. The model is then optimized with GRPO, resulting in our SPGRPO (**S**equential Sha**p**ley-enhanced **GRPO**) algorithm. Implementation details are provided in Appendix C.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate the performance of our method on the search-augmented QA benchmarks, including single-hop QA datasets (NQ Kwiatkowski et al. (2019), TriviaQA Joshi et al. (2017), and PopQA Mallen et al. (2023)) and multi-hop QA datasets (HotpotQA Yang et al. (2018), 2Wiki Ho et al. (2020), MuSiQue Trivedi et al. (2022), and Bamboogle Press et al. (2023)). Among these, NQ and HotpotQA are treated as in-domain (ID) datasets for training and evaluation, while the remaining datasets are considered out-of-domain (OOD) for evaluation only.

Baselines. We compare our methods with baselines from two lines of work: (1) outcome-level RL methods, including R1-Instruct, Search-R1 Jin et al. (2025), ZeroSearch Sun et al. (2025), and Tree-GRPO Ji et al. (2026), where learning is driven by final-answer rewards. (2) step-level RL methods, including IGPO Wang et al. (2025) and GiGPO Feng et al. (2025), which introduce intermediate supervision signals for each tool-use step. For ablation study, we consider a sequence of progressively simplified variants: starting from our Sequential Shapley, we first remove sequential dependency modeling to obtain the standard Shapley value Shapley (1953); we then discard interaction effects among steps and reduce it to IGPO Wang et al. (2025); finally, we remove step-level

	Single-Hop QA			Multi-Hop QA				
Method	NQ†	TriviaQA*	PopQA*	HotpotQA†	2Wiki*	MuSiQue*	Bamboogle*	**Avg.**
Qwen2.5-3B-Instruct								
R1-Instruct	27.0	53.7	19.9	23.7	29.2	7.2	29.3	27.1
Search-R1	34.1	54.5	37.8	32.4	31.9	10.3	26.4	32.5
ZeroSearch	41.4	57.4	44.8	27.4	30.0	9.8	11.1	31.7
Tree-GRPO	44.5	60.4	43.7	33.6	32.8	11.9	38.4	37.9
IGPO	40.2	58.1	43.2	35.7	34.9	12.3	64.5	41.3
GiGPO	42.0	59.5	42.4	36.9	37.0	12.6	64.1	42.1
SPGRPO	**44.6**±0.6	**61.4**±0.3	**46.6**±0.3	**39.1**±0.1	**40.2**±0.1	**14.5**±0.4	**68.0**±0.1	**44.9**±0.1
Qwen2.5-7B-Instruct								
R1-Instruct	21.0	44.9	17.1	20.8	27.5	6.0	19.2	22.4
Search-R1	39.3	61.0	39.7	37.0	40.1	14.6	36.8	38.5
ZeroSearch	43.6	61.8	**51.5**	34.6	35.2	18.4	27.8	39.1
Tree-GRPO	45.6	62.5	45.2	39.5	35.6	17.1	41.6	41.0
IGPO	44.0	60.1	45.4	42.9	42.5	16.8	40.0	41.6
GiGPO	46.4	**64.7**	46.1	41.6	43.6	18.9	68.9	47.2
SPGRPO	**47.2**±0.7	64.6±0.4	47.3±1.0	**44.6**±0.1	**43.9**±0.7	**21.0**±0.6	**71.7**±0.8	**48.6**±0.3

Table 1: Performance comparison on single-hop and multi-hop QA benchmarks. † denotes in-domain datasets, and * denotes out-of-domain datasets. All values denote success rates (%).

contribution and arrive at GRPO Shao et al. (2024). We also include SSVPO Li et al. (2026) as a comparison, which is designed for LLM reasoning and adapted here to our setting. In contrast to our method, SSVPO smooths out the influence of step ordering on the value function by averaging over reordered reasoning chains. This provides an alternative view of the Shapley value under sequential settings.

Implementation Details. We follow the experimental settings of prior search-agent studies Jin et al. (2025); Feng et al. (2025). Specifically, we use Qwen2.5-3B/7B-Instruct Qwen et al. (2025) as the base models and E5 Wang et al. (2022) as the retriever. We use success rate, measured by exact match, for both training and evaluation. Detailed hyperparameter settings are provided in Appendix C. For methods with experimental settings consistent with ours (e.g. GiGPO), we directly report the results from the original papers. For methods with differing settings (e.g. IGPO, Tree-GRPO) or requiring adaptation (e.g. standard Shapley, SSVPO), we implement them under the framework Feng et al. (2025) we use and report the reproduced results.

4.2 Overall Performance

Table 1 shows the overall performance comparison between SPGRPO and baseline methods. Accordingly, conclusions can be drawn as follows.

Step-level reward methods generally outperform outcome-level reward approaches. As shown in Table 1, the average success rate of outcome-level methods is 32.3 on 3B-scale models and 35.3 on 7B-scale models, while step-level methods improve these values to 42.8 and 45.8, respectively. This suggests that providing fine-grained intermediate supervision is beneficial for tool-call tasks.

SPGRPO achieves the best overall performance across both 3B and 7B models. Our method outperforms all baselines on both model scales, reaching the highest average performance of 44.9 (3B) and 48.6 (7B). Compared to strong step-reward baselines such as GiGPO, SPGRPO yields consistent improvements across most datasets, demonstrating its effectiveness.

SPGRPO brings larger improvements on multi-hop reasoning tasks than single-hop QA. The advantage of SPGRPO is especially pronounced on multi-hop datasets. On 3B-scale models, our method improves over the second-best baseline by an average of +0.97 on single-hop datasets and +2.7 on multi-hop datasets. A similar trend is observed on 7B-scale models. This demonstrates that our method is particularly effective for complex reasoning scenarios, as modeling interactions across steps brings greater benefits in multi-step tool-use settings.

Method	Inter.	Seq.	Single-Hop QA			Multi-Hop QA				Avg.
			NQ†	TriviaQA*	PopQA*	HotpotQA†	2Wiki*	MuSiQue*	Bamboogle*	
GRPO	×	×	39.2	57.8	42.4	34.4	34.5	10.7	65.3	40.6
IGPO	×	×	40.2	58.1	43.2	35.7	34.9	12.3	64.5	41.3
SSVPO	✓	×	43.2	60.3	45.1	37.6	37.7	13.2	67.4	43.5
Shapley	✓	×	43.8	60.8	45.9	37.7	38.5	13.8	65.7	43.7
SPGRPO	✓	✓	**44.6**	**61.4**	**46.6**	**39.1**	**40.2**	**14.5**	**68.0**	**44.9**

Table 2: Ablation study on Qwen2.5-3B-Instruct. Performance decreases progressively as sequential dependency modeling and interaction-aware credit assignment are removed.

4.3 Ablation Study

We conduct a step-by-step ablation study starting from GRPO and progressively introducing step-level credit assignment (IGPO), interaction modeling (standard Shapley value), and sequential dependency (SPGRPO), to analyze the contribution of each component. Table 2 shows that:

Step-level credit assignment provides an initial performance improvement. Comparing GRPO and IGPO, introducing step-level rewards leads to consistent performance gains across both single-hop and multi-hop datasets. This shows that step-level credit assignment is essential for improving LLM tool-use capabilities by providing more informative supervision for intermediate decisions.

Interaction modeling brings the largest performance gain among all components. Building on IGPO, incorporating interaction-aware credit assignment yields a substantial improvement, increasing the average success rate from 41.3 to 43.7, a gain of 2.4 percentage points. This suggests that local credit assignment alone is insufficient, and capturing higher-order interactions between steps plays a more important role in effective reasoning.

Sequential dependency modeling further improves performance by capturing structured dependencies. Finally, SPGRPO achieves the best performance by additionally modeling sequential dependencies on top of interaction-aware credit assignment, yielding improvements of 0.7 percentage points on single-hop datasets and 1.5 percentage points on multi-hop datasets over the standard Shapley value. We also observe that SSVPO, which aims to eliminate the influence of ordering on the value function, achieves a slightly lower average success rate than the standard Shapley value (43.5 vs. 43.7). This suggests that smoothing out sequential effects may not be beneficial under tool-use scenarios. Appendix D provides case studies illustrating the roles of interactions and sequential dependencies in credit assignment. These examples show how SPGRPO rewards steps that provide complementary evidence or enable useful subsequent searches.

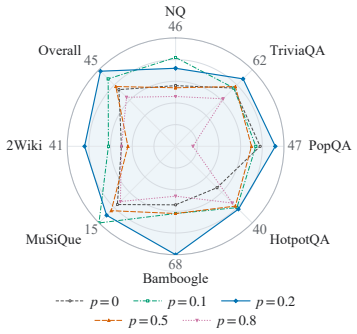

Figure 3: Effect of scaling parameter p across benchmarks (success rate, %). Concentric circles represent increments of 1%.

4.4 Insights into Sequential Dependency Modeling

We study the scaling hyperparameter p introduced in Section 3.3, which scales the likelihood-based probabilities of including future actions, with $p \in \{0, 0.1, 0.2, 0.5, 0.8\}$. The boundary case $p = 0$ disables future continuations and recovers standard Shapley credit assignment. As shown in Figure 3, both $p = 0.1$ and $p = 0.2$ improve overall performance over $p = 0$, with $p = 0.2$ achieving the highest overall accuracy. At $p = 0.5$, overall performance is slightly higher than that of the standard Shapley baseline. These results suggest relative robustness to the choice of p across the tested moderate settings, with gains at multiple values and competitive overall performance retained at $p = 0.5$. A single p is shared across benchmarks, so these gains do not require dataset-specific tuning. At $p = 0.8$, however, overall performance drops below the baseline, suggesting that overly strong weighting of future continuations can be detrimental. Effective later steps also deserve appropriate

credit, so accounting for sequential dependencies should not introduce an excessive positional bias toward earlier steps.

4.5 Computational Budget Analysis

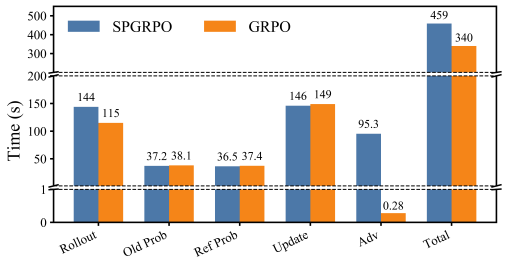

Figure 4: Per-step training time breakdown of SPGRPO vs. standard GRPO.

We analyze the computational overhead of our method when training Qwen-2.5-3B-Instruct on 4×H20 GPUs. Since our approach follows the same core pipeline as GRPO, the additional cost primarily stems from the proposed Sequential Shapley reward and advantage computation. As shown in Figure 4, the average per-iteration training time of SPGRPO is 459s, compared to 340s for GRPO, corresponding to an overhead of 119s (+35%). The main difference arises from the reward and advantage computation stage, increasing from 0.28s to 95.3s. This component accounts for 20.8% of the total training time in SPGRPO, and 79.9% of the overall overhead.

This overhead is reasonable, as Sequential Shapley evaluates interaction-aware contributions across step subsets. With the Prefix-Reuse Algorithm introduced in Appendix A.1, the additional cost remains limited. Moreover, SPGRPO does not introduce any additional critic model, thereby preserving the lightweight design of GRPO while enabling more fine-grained training signals.

5 Related Works

Recent advances in reinforcement learning (RL) have established it as a key technique for enhancing LLM-based agents, with methods such as PPO Schulman et al. (2017), GRPO Shao et al. (2024), RLOO Kool et al. (2019), and GSPO Zheng et al. (2025c) demonstrating strong performance. Building on this, a line of work has applied RL to search-based agents, enabling end-to-end optimization of retrieval-augmented reasoning. Approaches such as DeepRetrieval Jiang et al. (2025) and subsequent extensions such as Search-R1 Jin et al. (2025), DeepResearcher Zheng et al. (2025b), R1-searcher Song et al. (2025a) and R1-searcher++ Song et al. (2025b) have expanded this paradigm to multi-turn reasoning and tool use, further improving interaction with external environments.

To address the limitations of sparse outcome-level rewards, recent methods have introduced step-level supervision. For example, ReasonRAG Zhang et al. (2025) uses Monte Carlo Tree Search to approximate step values, StepSearch Zheng et al. (2025a) supervises intermediate steps with retrieved evidence, IGPO Wang et al. (2025) assigns credit based on information gain, and GiGPO Feng et al. (2025) estimates relative advantages via anchor-based grouping. While these methods provide more informative signals than outcome-only rewards, they either rely on external supervision or fail to jointly capture interactions among steps and the sequential dependencies in multi-step reasoning, leaving room for more principled and generalizable credit assignment mechanisms.

6 Conclusion

In this paper, we propose *Sequential Shapley*, an interaction- and dependency-aware credit assignment framework for multi-step LLM agents. Each realized rollout defines a Shapley game, and by extending the classical Shapley value with continuation-induced closures, our method captures both step interactions and sequential dependencies among tool-use steps. We further provide theoretical analysis showing that Sequential Shapley can be approximately interpreted as a backward redistribution of standard Shapley values. Extensive experiments demonstrate consistent improvements over strong baselines, while ablation studies verify the importance of modeling both interaction effects and sequential dependencies. Compared to the original GRPO algorithm, our method introduces only a 35% increase in computational cost, thus still preserving the lightweight nature of GRPO. These results establish Sequential Shapley as a strong and practical baseline for fine-grained credit assignment in multi-turn LLM agents.

AI USE STATEMENT

In this work, we used generative AI tools for assisting with translation, revising the problem definition, and strengthening the mathematical claim in Corollary 1 and refining its proof. We have not used generative AI tools for generating synthetic datasets or cleaning and reformatting datasets. Additionally, we used generative AI tools for improving the plotting and visual presentation of Figure 3 and editing the manuscript to improve language and readability. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Xuhui Zheng, Kang An, Ziliang Wang, Yuhang Wang, and Yichao Wu. Stepsearch: Igniting llms search ability via step-wise proximal policy optimization. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 21816–21841, 2025a.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 414–431, 2025b.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. *arXiv preprint arXiv:2010.03768*, 2020.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025c.

Jian Hu, Jason Klein Liu, Haotian Xu, and Wei Shen. Reinforce++: Stabilizing critic-free policy optimization with global advantage normalization. *arXiv preprint arXiv:2501.03262*, 2025.

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *arXiv preprint arXiv:2402.14740*, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Guoqing Wang, Sunhao Dai, Guangze Ye, Zeyu Gan, Wei Yao, Yong Deng, Xiaofeng Wu, and Zhenzhe Ying. Information gain-based policy optimization: A simple and effective approach for multi-turn llm agents. *arXiv preprint arXiv:2510.14967*, 2025.

Guanting Dong, Licheng Bao, Zhongyuan Wang, Kangzhi Zhao, Xiaoxi Li, Jiajie Jin, Jinghan Yang, Hangyu Mao, Fuzheng Zhang, Kun Gai, et al. Agentic entropy-balanced policy optimization. *arXiv preprint arXiv:2510.14545*, 2025.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. Group-in-group policy optimization for llm agent training. *arXiv preprint arXiv:2505.10978*, 2025.

Lloyd S. Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2:307–317, 1953.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, Cambridge, MA, 2nd edition, 2018.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1601–1611, 2017.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st annual meeting of the association for computational linguistics*, pages 9802–9822, 2023.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, 2018.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, 2020.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711, 2023.

Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Yan Zhang, Fei Huang, and Jingren Zhou. Zerosearch: Incentivize the search capability of llms without searching. *arXiv preprint arXiv:2505.04588*, 2025.

Yuxiang Ji, Ziyu Ma, Yong Wang, Guanhua Chen, Xiangxiang Chu, and Liaoni Wu. Tree search for llm agent reinforcement learning. *arXiv preprint arXiv:2509.21240*, 2026.

Yugu Li, Zehong Cao, Jianglin Qiao, and Siyi Hu. Ssvpo: Effective step-level credit assignment for rl training of language models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 reinforce samples, get a baseline for free! *ICLR 2019 Deep Reinforcement Learning meets Structured Prediction Workshop*, 2019.

Pengcheng Jiang, Jiacheng Lin, Lang Cao, Runchu Tian, SeongKu Kang, Zifeng Wang, Jimeng Sun, and Jiawei Han. Deepretrieval: Hacking real search engines and retrievers with large language models via reinforcement learning. *arXiv preprint arXiv:2503.00223*, 2025.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025a.

Huatong Song, Jinhao Jiang, Wenqing Tian, Zhipeng Chen, Yuhuan Wu, Jiahao Zhao, Yingqian Min, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher++: Incentivizing the dynamic knowledge acquisition of llms via reinforcement learning. *arXiv preprint arXiv:2505.17005*, 2025b.

Wenlin Zhang, Xiangyang Li, Kuicai Dong, Yichao Wang, Pengyue Jia, Xiaopeng Li, Yingyi Zhang, Derong Xu, Zhaocheng Du, Huifeng Guo, et al. Process vs. outcome reward: Which is better for agentic rag reinforcement learning. *arXiv preprint arXiv:2505.14069*, 2025.

A Computational Efficiency Improvements

A.1 Prefix-Reuse algorithm

An intuitive implementation is to sample continuation subsets C according to $P(C \mid S)$ in order to evaluate $\tilde{V}(S)$, which introduces an extra cost on top of the original exponential Shapley complexity. To address this bottleneck, we propose the Prefix-Reuse Algorithm, which evaluates each subset only once and reuses its probability estimates across all compatible prefix subsets. The key idea is to construct a serialized trajectory sequence that embeds search steps, retrieved information, and the final answer into a unified template. Specifically, for each index coalition $T = \{t_1, \ldots, t_{|T|}\} \subseteq N$, with $t_1 < \cdots < t_{|T|}$, let $A_T = (a_{t_1}, \ldots, a_{t_{|T|}})$ denote the corresponding ordered action subsequence. With ans denoting the reference answer, we construct trajectory sequences by mirroring the rollout format used during training: `[SYSTEM PROMPT] [QUESTION] question <search>`a_{t_1}`</search> <information>`e_{t_1}`</information>` $\cdots$ `<answer>ans</answer>` .

We first enumerate all index coalitions $T \subseteq N$, evaluate the serialized template under teacher forcing, and extract the token probabilities for every search segment a_{t_j} and for ans. Their geometric means give the action-likelihood score $\bar{\pi}_\theta(a_{t_j} \mid q, A_{T_{<t_j}})$ and the coalition value $V(T)$, respectively, where $T_{<k} = \{j \in T : j < k\}$ and $A_{T_{<k}}$ denotes the corresponding action subsequence together with its environmental responses. We cache $V(T)$ and the retention parameters $\rho_\theta(a_{t_j} \mid q, A_{T_{<t_j}}) = p\,\bar{\pi}_\theta(a_{t_j} \mid q, A_{T_{<t_j}})$ for coefficient computation. Let $m(T) = \max T$, with $m(\emptyset) = 0$. The prefix-reuse quantities are

$$U(T) = \prod_{i=1}^{m(T)} \rho_\theta(a_i \mid q, A_{T_{<i}})^{\mathbb{I}[i \in T]} \left(1 - \rho_\theta(a_i \mid q, A_{T_{<i}})\right)^{\mathbb{I}[i \notin T]}$$

$$\mathrm{coef}(T) = U(T)\,V(T) \prod_{i=m(T)+1}^{n} \left(1 - \rho_\theta(a_i \mid q, A_T)\right), \qquad \tilde{V}(S) = \frac{1}{U(S)} \sum_{T:\, S \preceq T} \mathrm{coef}(T),$$

where $S \preceq T$ means that $S \subseteq T$ and every index in $T \setminus S$ is larger than $m(S)$, and the retention parameters for skipped intermediate actions in $U(T)$ are obtained from cached forward passes of subsets where the same indexed action is retained under the corresponding prefix. Details and proof of correctness can be found in Appendix F.

Lemma 2 (Computational Complexity Analysis)**.** *Considering only teacher-forced forward passes, the Prefix-Reuse Algorithm matches standard Shapley complexity.*

A.2 Sampled Prefix Reuse for Longer-Horizon Scalability

Although the prefix-reuse algorithm removes the additional rollout required to estimate each continuation, its exact version still evaluates all 2^n coalitions of n search-step players. This is practical for the default four-step setting (three search actions followed by the final answer), which requires only $2^3 = 8$ coalition evaluations, but becomes increasingly expensive for longer trajectories. For example, increasing the maximum trajectory length from four to six steps increases the number of player coalitions from 8 to $2^5 = 32$. We therefore use exact enumeration for trajectories containing at most three search actions and switch to the following sampled estimator for longer trajectories.

One-shot coalition sampling and prefix reuse. We first draw a fixed budget of coalitions $\Omega_B = \{T_1, \ldots, T_B\}$ from a known proposal $q(T)$ with full support, using KernelSHAP weights to favor informative coalition sizes. For every sampled T, we perform one teacher-forced evaluation of A_T with the serialized template in Appendix A.1. This same evaluation provides the coalition value $V(T)$ and the retention parameters required by prefix reuse. Consequently, for each coalition T, we define the probability of stopping after its last retained action as

$$P_{\text{stop}}(T) := \prod_{k=m(T)+1}^{n} \big(1 - \rho_\theta(a_k \mid q, A_T)\big),$$

namely, the probability of skipping every remaining action after $m(T)$. Therefore, for each compatible prefix $S \preceq T$, prefix reuse also provides the observed continuation quantity

$$P(T \setminus S \mid S) = P_{\text{stop}}(T)\frac{U(T)}{U(S)}.$$

Thus, the value observations and the continuation-probability observations are collected from the same sampled coalitions; the method does not introduce a second sampling stage.

Shapley-additive completion of continuation probabilities. Because only a subset of coalitions is evaluated, some Bernoulli factors needed to construct $P(C \mid S)$ are not present in the prefix-reuse cache. We use the observed continuation information to fit an auxiliary KernelSHAP additive surrogate in logit space. In particular, for a conditional Bernoulli factor $x_{S,k}(R) = \rho_\theta(a_k \mid q, A_S, A_R)$, where $R \subseteq \{m(S)+1, \ldots, k-1\}$ indexes the continuation prefix before action a_k, we use the approximation

$$\operatorname{logit} \widehat{x}_{S,k}(R) = \psi_{S,k,0} + \sum_{j \in R} \psi_{S,k,j}.$$

The auxiliary contributions $\psi_{S,k,j}$ are estimated by KernelSHAP-weighted regression from the factors exposed by the sampled coalitions. Observed factors are retained, while the additive surrogate is used only for missing prefix contexts. Applying the sigmoid function gives a valid Bernoulli parameter, and substituting these parameters into the product in Equation 2 yields $\widehat{P}(C \mid S)$. Since the completed distribution preserves the sequential Bernoulli form, it is normalized by construction and requires no additional normalization over continuations.

Sequential Shapley estimation from the same samples. After the continuation probabilities have been completed, the Sequential Shapley value can be rearranged as a weighted sum over coalition values:

$$\phi_i^{\text{seq}} = \sum_{T \subseteq N} W_i(T) V(T), \qquad W_i(T) = \sum_{S : S \preceq T} \Phi(i, S) P(T \setminus S \mid S).$$

We therefore need $V(T)$ only for the coalitions sampled at the beginning. For sampling with replacement, the resulting importance-sampling estimator is

$$\widehat{\phi}_i^{\text{seq}} = \frac{1}{B}\sum_{b=1}^{B} \frac{\widehat{W}_i(T_b)}{q(T_b)} V(T_b), \qquad \widehat{W}_i(T) = \sum_{S : S \preceq T} \Phi(i, S) \widehat{P}(T \setminus S \mid S).$$

This reuses both $V(T_b)$ and the probability information obtained from the same B evaluations, and does not require imputing the unobserved coalition values. The model-evaluation cost is therefore

reduced from 2^n to B; we use $B = 8$ for six-step trajectories, matching the exact-enumeration budget in the default four-step setting.

If every $P(C \mid S)$ used in $W_i(T)$ is known exactly, the estimator above is an unbiased importance-sampling estimator of the exact Sequential Shapley value. In our efficient implementation, however, missing continuation probabilities are obtained through the Shapley-additive surrogate. This approximation changes the sampling weights and therefore does not preserve an unbiasedness guarantee. Unbiasedness can be recovered by allowing additional continuation resampling until all required $P(C \mid S)$ terms are evaluated exactly. We instead retain the fixed budget and evaluate the practical adequacy of the additive approximation empirically below.

Method	Rollout	Old Logprob	Ref Logprob	Update	Advantage	Total	Advantage / Total
GRPO (4 step)	115	38.1	37.4	149	0.28	340	–
SPGRPO (4 step)	144	37.2	36.5	146	95.3	459	20.8%
GRPO (6 step)	203	61.4	60.1	241	0.29	567	–
SPGRPO (6 step)	208	62.5	61.2	248	233	813	28.7%
SPGRPO (6 step, 8 samples)	209	63.7	62.1	252	127	714	17.8%

Table 3: Runtime breakdown under four- and six-step maximum trajectory lengths. The sampled estimator reduces six-step advantage-computation time from 233 to 127 and lowers its share of total training time from 28.7% to 17.8%.

Method	TriviaQA	PopQA	NQ	HotpotQA	2Wiki	MuSiQue	Bamboogle	Avg. (Multi-hop)	Avg. (All)
SPGRPO (4 step)	58.5	44.9	40.2	34.9	35.2	12.1	62.5	36.2	41.2
SPGRPO (6 step)	59.3	44.5	40.4	36.6	35.8	12.3	65.3	37.5	42.1
SPGRPO (6 step, 8 samples)	58.6	44.7	40.2	36.3	35.7	12.7	64.9	37.4	41.9

Table 4: Performance after 100 training steps. Eight-coalition sampling nearly matches exact six-step enumeration while substantially reducing advantage computation time.

Tables 3 and 4 demonstrate the resulting trade-off. Exact six-step SPGRPO improves the average score from 41.2 to 42.1, but its advantage computation takes 233 seconds. With only eight sampled coalitions, the average score remains 41.9 (42.1 for exact enumeration), while advantage computation drops to 127 seconds and total time decreases from 813 to 714 seconds. The sampling extension therefore preserves nearly all of the longer-horizon gain while making Sequential Shapley credit assignment substantially more scalable as the number of tool-use steps grows.

B Calibration Check for the Continuation Surrogate

The continuation probability in Equation 2 follows an action-centric definition: its underlying raw autoregressive likelihood is the probability of reproducing an exact observed action under the context induced by the retained action–state pairs. The geometric mean corrects the systematic dependence of this raw sequence probability on action length, while the shared factor p calibrates the resulting numerical scale. We assess this calibration against empirical re-rollout frequencies for observed continuations.

Re-rollout protocol. We sample 100 HotpotQA trajectories and enumerate all valid (S, C) pairs in each trajectory. For every evaluated context induced by S, we perform 32 independent re-rollouts and compare the estimated $P(C \mid S)$ with two empirical continuation frequencies. Under *strict-query matching*, every search action in C must be reproduced exactly and in the same order. Under *functional-evidence matching*, the re-rolled query strings may differ, but their retrieved evidence should overlap with that of the corresponding observed actions. The former directly matches our action-centric definition, whereas the latter measures a more permissive, state-centric notion of continuation.

Table 5 provides an absolute-error check of the calibrated surrogate. Across the two matching criteria, MAE ranges from 0.157 to 0.185 and RMSE ranges from 0.180 to 0.222. These errors provide a scale-level sanity check for using the calibrated likelihood as an action-centric weighting signal for continuations from the observed rollout.

Matching criterion	Global MAE	Within-trajectory MAE	Global RMSE	Within-trajectory RMSE
Strict query	0.185	0.184	0.222	0.209
Functional evidence	0.162	0.157	0.198	0.180

Table 5: Calibration error between $P(C \mid S)$ and empirical continuation frequencies on the same 100 trajectories, using 32 re-rollouts per evaluated context. Lower is better.

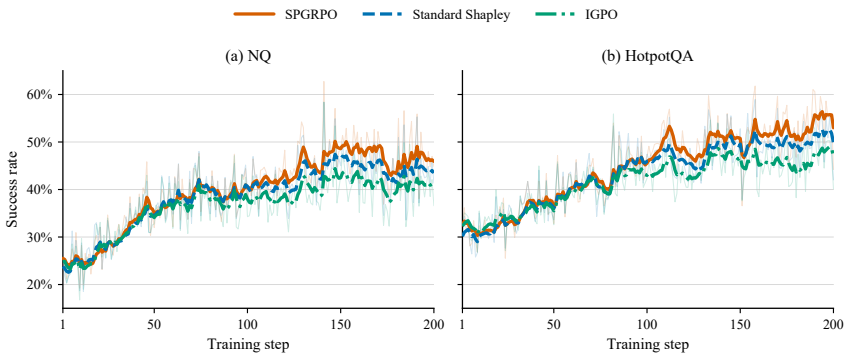

Figure 5: Training success rates of Qwen2.5-3B-Instruct on the two training datasets.

C Implementation Details

In practice, we treat `</search>` as the EOS token for each search action when computing its action likelihood. We incorporate Sequential Shapley into policy optimization by treating it as a step-level reward signal. For intermediate search steps, we directly use the corresponding Sequential Shapley value as the reward, while for the final answer step, we use a correctness-based signal defined as $\mathrm{Acc} - \tilde{V}(N)$, where $\mathrm{Acc} \in \{0, 1\}$ denotes whether the final answer is correct and $\tilde{V}(N)$ is the estimated value of the full trajectory. By the efficiency property of the Shapley value, the total Shapley reward is: $\mathrm{Acc} - \tilde{V}(\emptyset)$. Here, $\tilde{V}(\emptyset)$ serves as an initial baseline that reflects the model's expected performance. As a result, correctly solving harder questions with lower initial baseline yields larger surplus rewards, encouraging stronger policy updates for trajectories where tool use provides substantial improvement.

During optimization, we follow the standard GRPO framework to compute advantages and policy gradients. Notably, we separate search steps and the answer step into different groups for advantage estimation, allowing each type of step to be normalized and compared within its own context. The resulting advantages are then used in the standard GRPO loss to update the policy.

We train the 3B models on 4 NVIDIA H20 GPUs and the 7B models on 8 NVIDIA H20 GPUs. For the SPGRPO results in Table 1, we conduct three independent runs and report the mean and standard deviation across runs. Main hyperparameter settings are provided in Table 6.

Hyperparameter	Value	Hyperparameter	Value
Max steps	4	Actor learning rate	1×10^{-6}
Rollout group size	5	Rollout temperature	1.0
Training batch size	256	Mini-batch size	512
Max prompt length	4096	Max response length	512
KL-divergence loss coefficient	0.001	Invalid action penalty coefficient	0.01

Table 6: Training hyperparameters.

Figure 5 shows that the three methods perform similarly at the beginning of training, while their performance gradually separates after approximately 50–100 steps. This delayed separation suggests that the advantage of SPGRPO develops progressively during optimization.

Selection of the scaling hyperparameter p. Since no validation split is available in our training data setup, we select p using training-set performance on NQ and HotpotQA. We train each of the four configurations $p \in \{0.1, 0.2, 0.5, 0.8\}$ for 100 training steps and average the training success rates over steps 80–100 separately for each dataset. We then take the unweighted mean of the two dataset-level averages as the selection criterion. As shown in Figure 6, the resulting mean success rates are 41.3%, 42.4%, 41.55%, and 41.0%, respectively. We therefore select $p = 0.2$, which achieves the highest mean, and use this single value across benchmarks.

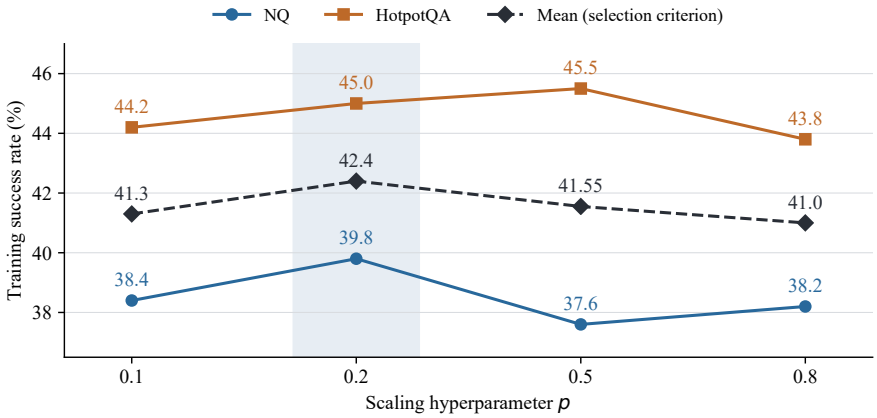

Figure 6: Selection of p using training-set success rates averaged over training steps 80–100 of 100-step runs. NQ and HotpotQA scores are shown alongside their unweighted mean. The shaded column highlights the selected value $p = 0.2$, which yields the highest mean success rate (42.4%).

D Case Studies: Interactions and Sequential Dependencies

We use case studies to compare how IGPO, standard Shapley, and Sequential Shapley assign credit to individual search steps. Section D.1 examines credit assignment when answering a question requires combining information from multiple search steps. Section D.2 examines sequentially dependent searches, where information retrieved at earlier steps guides or enables later queries.

D.1 Case Study of Interaction Modeling

Environment | Question

Who was born first, Renen Schorr or Michael Schultz?

Agent | Search 1

<think> To determine who was born first between Renen Schorr and Michael Schultz, I need to find their respective birth years. I will first look up Renen Schorr's birth year. </think>
<search>Renen Schorr birth year</search>

Environment | Information 1

<information>
[...]**Doc 3: Renen Schorr**
Renen Schorr [...] (Hebrew: [...]; born Jerusalem, Israel, **1952**) is a film director, screenwriter, film producer and Israeli film activist. In 1989, he founded Israel's first independent, national school for film and television [...].
</information>

Agent | Search 2

<think> The search result states that Renen Schorr was born in 1952. I still need Michael Schultz's birth year to compare the two. </think>
<search>Michael Schultz birth year</search>

Environment | Information 2

<information>
Doc 1: Michael Schultz (gallerist)
[...] Michael Schultz (born **4 November 1951**) is an internationally active German gallerist.
[...]</information>

Agent | Answer

<think> From the search results provided, we can confirm that Renen Schorr was born in 1952, and Michael Schultz was born on 4 November 1951. Since 1951 is earlier than 1952, Michael Schultz was born first. </think>
<answer>Michael Schultz</answer>

Step credits

Method	Step a_1	Step a_2
IGPO	0.0558	0.7378
Standard Shapley	0.3761	0.4175
Sequential Shapley	0.3694	0.3931

In this two-search trajectory, a_1 retrieves Schorr's birth year and a_2 retrieves Schultz's. Either date alone leaves the comparison incomplete. IGPO gives 0.0558 to the first acquisition and 0.7378 to the second, indicating that the second step is considered much more important than the first step. Standard Shapley recognizes that both pieces contribute to the inference and assigns 0.3761 and 0.4175. Sequential Shapley also captures this complementarity, assigning similarly balanced credits of 0.3694 and 0.3931. The smaller total credit arises from the baseline: IGPO and standard Shapley sum to $V(N) - V(\emptyset)$, whereas Sequential Shapley sums to $\tilde{V}(N) - \tilde{V}(\emptyset)$ (Theorem 1). Here, $\tilde{V}(N) = V(N)$ because no future steps remain, while $\tilde{V}(\emptyset) = \mathbb{E}_{C\sim P(\cdot|\emptyset)}[V(C)]$ already includes the expected value of future continuations, making it usually greater than $V(\emptyset)$.

D.2 Case Studies of Sequential Modeling

Enabling credit. An earlier search can supply information needed to formulate a later useful query. The following proposition isolates this sequential effect in a limiting case where the earlier step has no direct evidence contribution. It provides a theoretical reference for the cases below, whose value functions may also include direct contributions and evidence interactions.

Proposition 1 (Two-Step Enabling Credit)**.** *For $N = \{1, 2\}$, suppose $V(S) = v_0 + d\,\mathbf{1}_{2\in S}$ with $d > 0$. Under the Bernoulli closure in Eq. equation 2, let $\rho_k(T)$ denote the retention gate of step k given retained context T, and set $\alpha = \rho_1(\emptyset)$, $\beta = \rho_2(\emptyset)$, and $\gamma = \rho_2(\{1\})$. Then*

$$\phi_1^{\text{seq}} = \frac{d}{2}(1 - \alpha)(\gamma - \beta).$$

For $\alpha < 1$, the first step receives positive credit if and only if it increases the second step's conditional retention probability.

Proof. The closed values are $\tilde{V}(\emptyset) = v_0 + d[\alpha\gamma + (1 - \alpha)\beta]$, $\tilde{V}(\{1\}) = v_0 + d\gamma$, and $\tilde{V}(\{2\}) = \tilde{V}(\{1, 2\}) = v_0 + d$. Substituting these into the two-player Shapley formula gives

$$\phi_1^{\text{seq}} = \tfrac{1}{2}[\tilde{V}(\{1\}) - \tilde{V}(\emptyset) + \tilde{V}(\{1, 2\}) - \tilde{V}(\{2\})] = \tfrac{d}{2}(1 - \alpha)(\gamma - \beta).$$

□

D.2.1 CASE 1

Environment | Question

In what year was the architect of the St Andrew's Church, Blackburn born?

Agent | Search 1

<think> To answer this question, I need to determine the year of birth of the architect who designed St Andrew's Church in Blackburn. [...] I will first attempt to find this information using a search. </think>
<search>architect of St Andrew's Church, Blackburn, birth year</search>

Environment | Information 1

<information>
Doc 1: St Andrew's Church, Blackburn
St Andrew's Church is in Livesey Branch Road, Blackburn, Lancashire, England. It is a redundant Anglican church designed by the Lancaster architect **E. G. Paley**. The foundation stone was laid in December 1866, but the church was not consecrated until 1877. [...]

Doc 3: Holy Trinity Church, Blackburn
[...] The church was designed by the Lancaster architect Edmund Sharpe [...].
</information>

Agent | Search 2

<think> From the provided information, I see that the architect for St Andrew's Church, Blackburn is identified as E. G. Paley. However, it does not provide his birth year. To find the birth year, I will conduct a search for E. G. Paley. </think>
<search>What year was architect E. G. Paley born?</search>
Query translated from Chinese.

Environment | Information 2

<information>
Doc 1: Harvard Summit for Young Leaders in China
[...] 2007, Aug. 7 Chinese Education News Online [...] Harvard student leaders and student participants at US-China Summit have a lively debate with Yi Zhongtian [...].

Doc 2: Paley–Wiener integral
In mathematics, the Paley–Wiener integral is a simple stochastic integral. [...] The integral is named after its discoverers, **Raymond Paley** and Norbert Wiener. [...]

Doc 3: John Frank (epidemiologist)
[...] Interventions in community settings that prevent or delay disablement in later life: An overview of the evidence. [...]
</information>

Agent | Search 3

<think> I have found the architectural information for St Andrew's Church, Blackburn, but the birth year of the architect E. G. Paley is not mentioned in the provided information. To answer this question, I will need to perform an additional search. </think>
<search>architect E. G. Paley birth year</search>

Environment | Information 3

<information>
Doc 1: Edward Graham Paley
Edward Graham Paley, usually known as E. G. Paley (**3 September 1823** – 23 January 1895), was an English architect who practised in Lancaster, Lancashire, in the second half of the 19th century. After leaving school in 1838, he went to Lancaster to become a pupil of Edmund Sharpe, and in 1845 he joined Sharpe as a partner. [...]

Doc 3: Albert Paley
Albert Paley (born 1944) is an American modernist metal sculptor. [...]
</information>

Agent | Answer

<think> From the information provided in <information>, the details about Edward Graham Paley, specifically his birth year, are given as 1823. Since Edward Graham Paley is identified as the architect of St Andrew's Church in Blackburn, he must be the person we're looking for. </think>
<answer>1823</answer>

Step credits

Method	Step a_1	Step a_2	Step a_3
IGPO	0.5516	0.0117	0.2633
Standard Shapley	0.3823	0.0258	0.4186
Sequential Shapley	0.4066	0.0228	0.3612

In this three-search trajectory, a_1 identifies the architect as E. G. Paley, a_2 retrieves irrelevant information, and a_3 finds his birth year. Both later queries use the identity established by a_1. Standard Shapley assigns 0.3823 to a_1 and 0.4186 to a_3, while Sequential Shapley raises the first step's credit to 0.4066 and lowers the third step's credit to 0.3612, accounting for the first search's role in enabling the subsequent birth-year retrieval. The uninformative second step receives only 0.0117, 0.0258, and 0.0228 under IGPO, standard Shapley, and Sequential Shapley, respectively. This indicates that uninformative steps will not be rewarded. IGPO also gives the first step more credit than the third (0.5516 versus 0.2633). However, the next case (Section D.2.2) shows that this is a special case and does not imply that IGPO accounts for the expected future benefits enabled by earlier steps.

D.2.2 CASE 2

Environment | Question

What was the nationality of the composer who wrote *Treatise on Instrumentation*?

Agent | Search 1

<think> I don't have direct knowledge of the composer who wrote the *Treatise on Instrumentation*, so I will need to conduct a search to find this information. </think>
<search>composer of Treatise on Instrumentation nationality</search>

Environment | Information 1

<information>
Doc 1: Treatise on Instrumentation
[...] abbreviated in English as the Treatise on Instrumentation (sometimes Treatise on Orchestration) is a technical study of Western musical instruments, written by **Hector Berlioz**. It was first published in 1844 [...].
</information>

Agent | Search 2

<think> From the search results, it is clear that the treatise on instrumentation was written by Hector Berlioz. Now I need to identify Hector Berlioz's nationality. </think>
<search>Hector Berlioz nationality</search>

Environment | Information 2

<information>
Doc 1: Hector Berlioz
Louis-Hector Berlioz [...] (11 December 1803 – 8 March 1869) was a **French Romantic composer**. His output includes orchestral works such as the *Symphonie fantastique* and *Harold in Italy* [...].
</information>

Agent | Answer

<think> From the information provided, we know that Hector Berlioz wrote the Treatise on Instrumentation. The nationality of Hector Berlioz is given in the search results as French. </think>
<answer>French</answer>

Step credits

Method	Step a_1	Step a_2
IGPO	0.2452	0.5532
Standard Shapley	0.2004	0.5980
Sequential Shapley	0.3387	0.4015

In this trajectory, a_1 identifies Hector Berlioz as the composer, allowing a_2 to query his nationality. Standard Shapley assigns 0.2004 to the first step and 0.5980 to the second. Sequential Shapley again gives more credit to the enabling step, raising its score to 0.3387 while reducing the second step's score to 0.4015. IGPO assigns 0.2452 to the first step and 0.5532 to the second, reversing the ordering in Section D.2.1 despite the similar dependency between identification and subsequent retrieval. Its credit depends on the immediate increase in answer likelihood and does not explicitly account for how an earlier search affects the expected value of future continuations.

E Theoretical Analysis

E.1 Proof of Theorem 1

Proof. The key observation is that Sequential Shapley only replaces the original value function V with the induced value function $\tilde{V}$, while the Shapley aggregation rule remains unchanged. Therefore, once $\tilde{V}$ is fixed, the proof follows the standard Shapley argument. For completeness, we verify the four axioms below.

Efficiency. The efficiency axiom requires that the total credit assigned to all players equals the total induced value:

$$\sum_{i \in N} \phi_i^{\text{seq}}(\tilde{V}) = \tilde{V}(N) - \tilde{V}(\emptyset).$$

Proof. By definition,

$$\phi_i^{\text{seq}}(\tilde{V}) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n - |S| - 1)!}{n!} \left[\tilde{V}(S \cup \{i\}) - \tilde{V}(S) \right].$$

Equivalently, the Shapley value can be written as the expectation over all permutations π of N:

$$\phi_i^{\text{seq}}(\tilde{V}) = \mathbb{E}_\pi \left[\tilde{V}(P_i^\pi \cup \{i\}) - \tilde{V}(P_i^\pi) \right],$$

where P_i^π denotes the set of players preceding i in permutation π. Summing over all players gives

$$\sum_{i\in N}\phi_i^{\text{seq}}(\tilde{V}) = \mathbb{E}_\pi\left[\sum_{i\in N}\left(\tilde{V}(P_i^\pi\cup\{i\}) - \tilde{V}(P_i^\pi)\right)\right].$$

For any fixed permutation, the inner summation is telescoping, yielding

$$\sum_{i\in N}\left(\tilde{V}(P_i^\pi\cup\{i\}) - \tilde{V}(P_i^\pi)\right) = \tilde{V}(N) - \tilde{V}(\emptyset).$$

Thus,

$$\sum_{i\in N}\phi_i^{\text{seq}}(\tilde{V}) = \tilde{V}(N) - \tilde{V}(\emptyset).$$

Symmetry. The symmetry axiom requires that if two players i and j have identical marginal contributions under $\tilde{V}$, i.e.,

$$\tilde{V}(S\cup\{i\}) - \tilde{V}(S) = \tilde{V}(S\cup\{j\}) - \tilde{V}(S), \quad \forall S\subseteq N\setminus\{i,j\},$$

then they receive the same credit:

$$\phi_i^{\text{seq}}(\tilde{V}) = \phi_j^{\text{seq}}(\tilde{V}).$$

Proof. Suppose that players i and j have identical marginal contributions under $\tilde{V}$, i.e.,

$$\tilde{V}(S\cup\{i\}) - \tilde{V}(S) = \tilde{V}(S\cup\{j\}) - \tilde{V}(S), \quad \forall S\subseteq N\setminus\{i,j\}.$$

Since the Shapley weight

$$w(|S|) = \frac{|S|!(n-|S|-1)!}{n!}$$

depends only on the coalition size, the corresponding marginal contributions of i and j are assigned the same weights. Therefore,

$$\phi_i^{\text{seq}}(\tilde{V}) = \sum_S w(|S|)\big[\tilde{V}(S\cup\{i\}) - \tilde{V}(S)\big] = \sum_S w(|S|)\big[\tilde{V}(S\cup\{j\}) - \tilde{V}(S)\big] = \phi_j^{\text{seq}}(\tilde{V}).$$

Additivity. The additivity axiom requires that for any two induced value functions $\tilde{V}_1$ and $\tilde{V}_2$,

$$\phi_i^{\text{seq}}(\tilde{V}_1+\tilde{V}_2) = \phi_i^{\text{seq}}(\tilde{V}_1) + \phi_i^{\text{seq}}(\tilde{V}_2), \quad \forall i\in N.$$

Proof. By definition,

$$\phi_i^{\text{seq}}(\tilde{V}_1+\tilde{V}_2) = \sum_{S\subseteq N\setminus\{i\}} w(S)\left[(\tilde{V}_1+\tilde{V}_2)(S\cup\{i\}) - (\tilde{V}_1+\tilde{V}_2)(S)\right],$$

where

$$w(S) = \frac{|S|!(n-|S|-1)!}{n!}.$$

Expanding the marginal term gives

$$\begin{aligned}\phi_i^{\text{seq}}(\tilde{V}_1+\tilde{V}_2) &= \sum_{S\subseteq N\setminus\{i\}} w(S)\left[\tilde{V}_1(S\cup\{i\}) - \tilde{V}_1(S)\right] \\ &\quad + \sum_{S\subseteq N\setminus\{i\}} w(S)\left[\tilde{V}_2(S\cup\{i\}) - \tilde{V}_2(S)\right] \\ &= \phi_i^{\text{seq}}(\tilde{V}_1) + \phi_i^{\text{seq}}(\tilde{V}_2).\end{aligned}$$

Dummy player. The dummy player axiom requires that if player i makes no marginal contribution to any coalition under $\tilde{V}$, i.e.,

$$\tilde{V}(S\cup\{i\}) = \tilde{V}(S), \quad \forall S\subseteq N\setminus\{i\},$$

then

$$\phi_i^{\text{seq}}(\tilde{V}) = 0.$$

Proof. Substituting the dummy condition into the definition gives

$$\phi_i^{\text{seq}}(\tilde{V}) = \sum_{S\subseteq N\setminus\{i\}} w(S)\left[\tilde{V}(S\cup\{i\}) - \tilde{V}(S)\right] = \sum_{S\subseteq N\setminus\{i\}} w(S)\cdot 0 = 0.$$

Therefore, Sequential Shapley satisfies efficiency, symmetry, additivity, and dummy player on the induced value function $\tilde{V}$. □

E.2 Proof of Lemma 1

Proof. Fix a coalition $S \subseteq N$, and let

$$\mathcal{F}(S) = \{j_1, \ldots, j_K\}$$

be the ordered set of all future player indices after S, where $j_1 < \cdots < j_K$. For any continuation $C \in \mathcal{C}(S)$, denote

$$C_{<k} = C \cap \{j_1, \ldots, j_{k-1}\}.$$

Since continuation candidates are gated sequentially, the retention parameter for j_k may depend on q, the action subsequence A_S, and the previously retained continuation actions $A_{C_{<k}}$. For simplicity, write

$$p_k(C_{<k}) = \pi_\theta(a_{j_k} \mid q, A_S, A_{C_{<k}}).$$

Then the probability of a continuation C can be written as

$$P(C \mid S) = \prod_{k=1}^{K} [p_k(C_{<k})]^{\mathbf{1}[j_k \in C]} \, [1 - p_k(C_{<k})]^{\mathbf{1}[j_k \notin C]}.$$

Because $p_k(C_{<k}) \in [0, 1]$, we also have $1 - p_k(C_{<k}) \in [0, 1]$. Therefore, every factor in the above product lies in $[0, 1]$, and hence

$$P(C \mid S) \in [0, 1].$$

It remains to prove normalization. Summing over all continuations is equivalent to summing over the binary decision variables indicating whether each future step is retained. Let

$$b_k = \mathbf{1}[j_k \in C], \qquad k = 1, \ldots, K.$$

Then each continuation C uniquely corresponds to a binary sequence $(b_1, \ldots, b_K) \in \{0, 1\}^K$, and $C_{<k}$ is determined by $(b_1, \ldots, b_{k-1})$. Therefore,

$$\sum_{C \in \mathcal{C}(S)} P(C \mid S) = \sum_{b_1 \in \{0,1\}} \sum_{b_2 \in \{0,1\}} \cdots \sum_{b_K \in \{0,1\}} \prod_{k=1}^{K} [p_k(C_{<k})]^{b_k} \, [1 - p_k(C_{<k})]^{1-b_k}.$$

We now eliminate the sums from b_K to b_1. For any fixed prefix $(b_1, \ldots, b_{K-1})$, the term involving b_K satisfies

$$\sum_{b_K \in \{0,1\}} [p_K(C_{<K})]^{b_K} \, [1 - p_K(C_{<K})]^{1-b_K} = p_K(C_{<K}) + \big(1 - p_K(C_{<K})\big) = 1.$$

Thus, after summing over b_K, the last factor is removed. Applying the same argument successively to $b_{K-1}, b_{K-2}, \ldots, b_1$, we obtain

$$\sum_{C \in \mathcal{C}(S)} P(C \mid S) = \sum_{b_1 \in \{0,1\}} \cdots \sum_{b_{K-1} \in \{0,1\}} \prod_{k=1}^{K-1} [p_k(C_{<k})]^{b_k} \, [1 - p_k(C_{<k})]^{1-b_k} = \cdots = 1.$$

Therefore,

$$\sum_{C \in \mathcal{C}(S)} P(C \mid S) = 1.$$

□

E.3 Proof of Theorem 2

Proof. By the definition of Sequential Shapley, we have

$$\phi_{\text{seq}}(i) = \sum_S \Phi(i, S) \widetilde{V}(S) = \sum_S \Phi(i, S) \sum_C P(C \mid S) V(S \cup C).$$

Using the decomposition

$$V(S) = V(\emptyset) + \sum_{k \in S} \phi_{\text{standard}}(k) + R(S),$$

we obtain

$$V(S \cup C) = V(\emptyset) + \sum_{k \in S \cup C} \phi_{\text{standard}}(k) + R(S \cup C).$$

Substituting it into the above expression gives

$$\begin{aligned}
\phi_{\text{seq}}(i) &= \sum_S \Phi(i,S) \sum_C P(C \mid S) \left[V(\emptyset) + \sum_{k \in S \cup C} \phi_{\text{standard}}(k) + R(S \cup C) \right] \\
&= V(\emptyset) \sum_S \Phi(i,S) \sum_C P(C \mid S) \\
&\quad + \sum_S \Phi(i,S) \sum_C P(C \mid S) \sum_{k \in S \cup C} \phi_{\text{standard}}(k) \\
&\quad + \sum_S \Phi(i,S) \sum_C P(C \mid S) R(S \cup C).
\end{aligned}$$

We first consider the constant term. Since $P(C \mid S)$ is a normalized continuation distribution, $\sum_C P(C \mid S) = 1$. Moreover, the signed Shapley coefficients satisfy $\sum_S \Phi(i,S) = 0$, because a constant value function has zero Shapley value. Therefore,

$$V(\emptyset) \sum_S \Phi(i,S) \sum_C P(C \mid S) = V(\emptyset) \sum_S \Phi(i,S) = 0.$$

Next, we decompose the additive term as

$$\sum_{k \in S \cup C} \phi_{\text{standard}}(k) = \sum_{k \in S} \phi_{\text{standard}}(k) + \sum_{j \in C} \phi_{\text{standard}}(j),$$

where C only contains future steps not included in S. Hence,

$$\begin{aligned}
&\sum_S \Phi(i,S) \sum_C P(C \mid S) \sum_{k \in S \cup C} \phi_{\text{standard}}(k) \\
&= \sum_S \Phi(i,S) \sum_C P(C \mid S) \sum_{k \in S} \phi_{\text{standard}}(k) + \sum_S \Phi(i,S) \sum_C P(C \mid S) \sum_{j \in C} \phi_{\text{standard}}(j).
\end{aligned}$$

For the first part, since $\sum_C P(C \mid S) = 1$, we have

$$\begin{aligned}
\sum_S \Phi(i,S) \sum_C P(C \mid S) \sum_{k \in S} \phi_{\text{standard}}(k) &= \sum_S \Phi(i,S) \sum_{k \in S} \phi_{\text{standard}}(k) \\
&= \phi_{\text{standard}}(i).
\end{aligned}$$

The last equality follows from the fact that $\sum_{k \in S} \phi_{\text{standard}}(k)$ is an additive value function, whose Shapley value for player i is exactly $\phi_{\text{standard}}(i)$.

For the second part, we exchange the order of summation:

$$\begin{aligned}
\sum_S \Phi(i,S) \sum_C P(C \mid S) \sum_{j \in C} \phi_{\text{standard}}(j) &= \sum_j \left[\sum_S \Phi(i,S) \sum_C P(C \mid S) \mathbf{1}_{j \in C} \right] \phi_{\text{standard}}(j) \\
&= \sum_j B_{i,j} \phi_{\text{standard}}(j).
\end{aligned}$$

By the definition of B, this term is the i-th component of $B\phi_{\text{standard}}$.

Finally, the residual term is

$$\sum_S \Phi(i,S) \sum_C P(C \mid S) R(S \cup C) = \Delta_i.$$

Combining the above terms, we obtain

$$\phi_{\text{seq}}(i) = \phi_{\text{standard}}(i) + \sum_j B_{i,j} \phi_{\text{standard}}(j) + \Delta_i.$$

In vector form, this gives

$$\phi_{\text{seq}} = (I + B)\phi_{\text{standard}} + \Delta.$$

This completes the proof. □

E.4 Proof of Corollary 1

Proof. Define the residual-induced value function

$$f(S) = \widetilde{R}(S) = \sum_{C \in \mathcal{C}(S)} P(C \mid S) R(S \cup C).$$

Normalization and nonnegativity of the continuation probabilities imply

$$|f(S)| \le \sum_{C \in \mathcal{C}(S)} P(C \mid S) |R(S \cup C)| \le M.$$

The decomposition theorem gives $\Delta = \Phi f$.

Absolute aggregate error. By the efficiency property of the Shapley value,

$$\sum_{i \in N} \Delta_i = f(N) - f(\emptyset).$$

Consequently,

$$\left| \sum_{i \in N} \Delta_i \right| \le |f(N)| + |f(\emptyset)| \le 2M.$$

Total absolute error. We next bound the Shapley value of any game f satisfying $|f(S)| \le M$. For $0 < t < 1$, let $X_1, \dots, X_n$ be independent Bernoulli random variables with parameter t, and set $S_t = \{i : X_i = 1\}$. The beta-integral identity

$$\int_0^1 t^k (1-t)^{n-k-1} \, dt = \frac{k!(n-k-1)!}{n!}$$

and conditioning on X_i give the following integral representation:

$$(\Phi f)_i = \int_0^1 \frac{\mathbb{E}[f(S_t)(X_i - t)]}{t(1-t)} \, dt.$$

For any fixed $\varepsilon \in \{-1, 1\}^n$, independence implies

$$\mathbb{E}\left[\left(\sum_i \varepsilon_i (X_i - t) \right)^2 \right] = nt(1-t).$$

Therefore, by Cauchy–Schwarz,

$$\begin{aligned} \varepsilon^\top \Phi f &\le M \int_0^1 \frac{\mathbb{E}|\sum_i \varepsilon_i (X_i - t)|}{t(1-t)} \, dt \\ &\le M\sqrt{n} \int_0^1 \frac{dt}{\sqrt{t(1-t)}} = \pi \sqrt{n} \, M. \end{aligned}$$

Taking the maximum over ε gives

$$\sum_{i \in N} |\Delta_i| = \|\Phi f\|_1 = \max_{\varepsilon \in \{-1,1\}^n} \varepsilon^\top \Phi f \le \pi \sqrt{n} \, M.$$

Finally, whenever $R = 0$, the decomposition is exact: $\phi_{\text{seq}} = (I + B)\phi_{\text{standard}}$. □

A sharper bound on the total absolute error. In fact, the second bound in Corollary 1 can be strengthened to

$$\sum_{i \in N} |\Delta_i| \le \kappa_n M \le \pi \sqrt{n} \, M, \qquad \kappa_n = \max_{\varepsilon \in \{-1,1\}^n} \sum_{S \subsetneq N} \left| \sum_{i \in N} \varepsilon_i \Phi(i, S) \right|.$$

To derive this refinement, note that the residual in Theorem 2 satisfies

$$R(\emptyset) = R(N) = 0, \qquad \phi_{\text{standard}}(R) = 0,$$

by the efficiency and additivity of the standard Shapley value. Moreover, $\mathcal{C}(N) = \{\emptyset\}$ implies $f(N) = R(N) = 0$. Using the dual representation of the ℓ_1 norm before applying the triangle inequality, we obtain

$$\begin{aligned}\|\Delta\|_1 &= \max_{\varepsilon\in\{-1,1\}^n} \sum_{S\subsetneq N} f(S) \sum_{i\in N} \varepsilon_i \Phi(i,S) \\ &\le M \max_{\varepsilon\in\{-1,1\}^n} \sum_{S\subsetneq N} \left|\sum_{i\in N} \varepsilon_i \Phi(i,S)\right| = \kappa_n M.\end{aligned}$$

This retains cancellation between Shapley coefficients belonging to different players, which is lost when bounding each $|\Delta_i|$ separately.

The coefficient κ_n can be evaluated by considering only the number p of positive entries in ε. For a coalition of size k, write $h = |\{i \in S : \varepsilon_i = 1\}|$. For $1 \le k \le n-1$,

$$\sum_{i\in N} \varepsilon_i \Phi(i,S) = \frac{2(nh - kp)}{\binom{n}{k}\, k(n-k)}.$$

For $S = \emptyset$, the coefficient is $-(2p-n)/n$. Consequently,

$$\kappa_n = \max_{p\in\{0,\dots,n\}} \left\{ \frac{|2p-n|}{n} + 2\sum_{k=1}^{n-1} \frac{\mathbb{E}|nH_{k,p} - kp|}{k(n-k)} \right\}, \tag{3}$$

where $H_{k,p}$ is the number of marked players in a uniformly sampled k-element coalition when p of the n players are marked. Explicitly,

$$\Pr(H_{k,p} = h) = \frac{\binom{p}{h}\binom{n-p}{k-h}}{\binom{n}{k}}, \qquad \max(0, k-n+p) \le h \le \min(k,p).$$

Thus Eq. (3) involves only finite scalar sums. It gives $\kappa_2 = 2$, $\kappa_3 = 3$, $\kappa_4 = 7/2$, and $\kappa_8 = 157/28$.

For the class of all games satisfying $|f(S)| \le M$ and $f(N) = 0$, the coefficient κ_n is exact: for a maximizing sign vector ε, choose $f(S) = M\,\mathrm{sign}(\sum_i \varepsilon_i \Phi(i,S))$ for $S \ne N$ and $f(N) = 0$. Hence the previously proved $\pi\sqrt{n}$ bound implies $\kappa_n \le \pi\sqrt{n}$. The original residual and continuation structure can impose additional constraints, so κ_n need not be the smallest coefficient for a particular continuation distribution.

E.5 Supplementary Explanation to the Main Text

Lemma 3 (Directional Structure of the Redistribution Matrix)**.** *Let*

$$B_{i,j} = \sum_S \Phi(i,S) \sum_C P(C \mid S) \mathbf{1}_{j\in C}$$

and suppose that for any $i < j$ and any coalition $T \subseteq N \setminus \{i,j\}$, adding the earlier step i does not decrease the probability of inducing the later step j, i.e.,

$$P(j \in C \mid T \cup \{i\}) \ge P(j \in C \mid T).$$

Then the redistribution matrix B has the following directional structure.

$$B_{i,j} \le 0, \text{ if } i \ge j \text{ else } B_{i,j} \ge 0.$$

Thus, under the monotonicity assumption, the B-induced term mainly redistributes credit from later steps to earlier ones.

Proof. For simplicity, define

$$P(j \in C \mid S) = \sum_C P(C \mid S) \mathbf{1}_{j\in C}.$$

Then

$$B_{i,j} = \sum_S \Phi(i, S) P(j \in C \mid S).$$

We first consider the case $i \geq j$. By the definition of valid future continuations, a future continuation C after coalition S can only contain steps whose indices are larger than $\max(S)$. Therefore, $j \in C$ is possible only when

$$\max(S) < j.$$

If $i \geq j$, this condition implies $i \notin S$. Hence, every coalition S that can make $P(j \in C \mid S)$ nonzero must satisfy $i \notin S$. For such coalitions, the signed Shapley coefficient is negative:

$$\Phi(i, S) = -\frac{|S|!(n - |S| - 1)!}{n!} < 0.$$

For all other coalitions, $P(j \in C \mid S) = 0$. Since $P(j \in C \mid S) \geq 0$, every nonzero term in the sum defining $B_{i,j}$ is non-positive. Therefore,

$$B_{i,j} \leq 0.$$

We next consider the case $i < j$. We group the terms in $B_{i,j}$ by pairs of coalitions that differ only by whether they contain i. Let $T \subseteq N \setminus \{i\}$. The two paired coalitions are T and $T \cup \{i\}$. Using the signed Shapley coefficients, we can write

$$B_{i,j} = \sum_{T \subseteq N \setminus \{i\}} w(|T|) \left[P(j \in C \mid T \cup \{i\}) - P(j \in C \mid T) \right],$$

where

$$w(|T|) = \frac{|T|!(n - |T| - 1)!}{n!} \geq 0.$$

Here, $w(|T|)$ is the standard Shapley weight associated with adding player i to coalition T.

Under the monotonicity assumption,

$$P(j \in C \mid T \cup \{i\}) - P(j \in C \mid T) \geq 0.$$

Since each weight $w(|T|)$ is non-negative, every term in the above sum is non-negative. Therefore,

$$B_{i,j} \geq 0.$$

Combining the two cases, entries with $i \geq j$ are non-positive, while entries with $i < j$ are non-negative under the monotonicity assumption. This indicates that the positive part of the B-induced redistribution mainly assigns credit from later steps j back to earlier steps i, completing the proof. □

F Pseudo Code

Our method is shown in Algorithm 1. Here we provide a brief explanation to illustrate its correctness.

Correctness of the Prefix-Reuse Algorithm. The correctness of Algorithm 1 follows from an equivalent reformulation of the closed value function under the sequential Bernoulli selection process. Let $N = \{1, 2, \cdots, n\}$ be the player-index set, and let $A_N = (a_i)_{i \in N}$ be its ordered action sequence. Let $m(T) = \max T$ denote the largest original index in T, with $m(\emptyset) = 0$. For any index subset $T \subseteq N$, define $T_{<k} = \{i \in T : i < k\}$. We first define the probability of reaching exactly the retained/skipped pattern of T up to its last retained action as

$$U(T) = \prod_{k=1}^{m(T)} \rho_\theta(a_k \mid q, A_{T_{<k}})^{\mathbb{I}[k \in T]} \big(1 - \rho_\theta(a_k \mid q, A_{T_{<k}})\big)^{\mathbb{I}[k \notin T]}.$$

Thus, $U(T)$ includes not only the retention parameters of retained actions, but also the complementary gate probabilities of skipped actions before $m(T)$.

For a coalition S and a future continuation $C \subseteq \{m(S)+1, \dots, n\}$, the continuation probability is

$$P(C \mid S) = \prod_{k=m(S)+1}^{n} \rho_\theta(a_k \mid q, A_S, A_{C_{<k}})^{\mathbb{I}[k \in C]} \big(1 - \rho_\theta(a_k \mid q, A_S, A_{C_{<k}})\big)^{\mathbb{I}[k \notin C]},$$

where $C_{<k} = \{i \in C : i < k\}$. Equivalently, if $T = S \cup C$ and $S \preceq T$, where $S \preceq T$ means that $S \subseteq T$ and every index in $T \setminus S$ is larger than $m(S)$, then

$$P(C \mid S) = \frac{U(T)}{U(S)} \prod_{k=m(T)+1}^{n} \big(1 - \rho_\theta(a_k \mid q, A_T)\big).$$

To see this, the ratio $U(T)/U(S)$ exactly contains all retain/skip gate decisions from $m(S)+1$ to $m(T)$, while the remaining product contains the probability of skipping every action after $m(T)$. We therefore define

$$P_{\text{stop}}(T) = \prod_{k=m(T)+1}^{n} \big(1 - \rho_\theta(a_k \mid q, A_T)\big)$$

and obtain

$$P(C \mid S) = P_{\text{stop}}(T) \frac{U(T)}{U(S)}.$$

This identity is the key reason why prefix reuse is still valid under the revised definition: once $U(T)$, $P_{\text{stop}}(T)$, and $V(T)$ are computed for each index subset T, their contribution to all prefix subsets $S \preceq T$ can be reused through the normalization factor $1/U(S)$.

Substituting the above identity into the definition of the closed value gives

$$\tilde{V}(S) = \sum_C P(C \mid S) V(S \cup C) = \frac{1}{U(S)} \sum_{T : S \preceq T} U(T) P_{\text{stop}}(T) V(T).$$

Algorithm 1 directly implements this reformulation. In Phase I, it caches the terminal value $V(T)$ and all context-conditioned retention parameters needed by the Bernoulli selection process. In particular, parameters for skipped intermediate actions are obtained by reusing the teacher-forced passes of other subsets in which the corresponding indexed action is retained. In Phase II, it forms the reusable coefficient

$$\text{coef}(T) = U(T) P_{\text{stop}}(T) V(T).$$

Thus, each coefficient corresponds to the unnormalized contribution of the index subset T to the closed values of all its prefixes. In Phase III, for each prefix S, the algorithm sums the coefficients of all supersets T satisfying $S \preceq T$ and divides by $U(S)$, thereby recovering exactly the closed value $\tilde{V}(S)$:

$$\tilde{V}(S) = \frac{1}{U(S)} \sum_{T : S \preceq T} \text{coef}(T).$$

Finally, after obtaining all closed values $\tilde{V}(S)$, Algorithm 1 applies the standard Shapley marginal contribution formula

$$\phi_{\text{seq}}(k) = \sum_{S \subseteq N \setminus \{k\}} w(S) \big(\tilde{V}(S \cup \{k\}) - \tilde{V}(S)\big), \quad w(S) = \frac{|S|!(n - |S| - 1)!}{n!}.$$

Therefore, when all index subsets are enumerated, Algorithm 1 exactly computes the Sequential Shapley value defined over the closed value function $\tilde{V}(\cdot)$. When index subsets are sampled instead, the same derivation yields a Monte Carlo estimator of the Sequential Shapley value.

G LIMITATIONS

Our current study mainly focuses on search-based LLM agents, where both the continuation distribution $P(C \mid S)$ and the counterfactual evaluation can be naturally instantiated through the likelihood of generating future tool-call steps and masking selected search results in the prompt

Algorithm 1 Prefix-Reuse Sequential Shapley Computation

Input: question q, player-index set $N = \{1, 2, \cdots, n\}$, ordered actions $A_N = (a_i)_{i \in N}$ and their responses $(e_i)_{i \in N}$, LLM policy π_θ
Initialize $\phi_i^{\text{seq}} \leftarrow 0,\ \forall i \in N$
Define $m(T) = \max T$ with $m(\emptyset) = 0$, and $T_{<k} = \{i \in T : i < k\}$
// Phase I: cache terminal values and context-conditioned retention parameters
for each index subset $T \subseteq N$ **do**
  Construct the serialized trajectory from A_T and the corresponding responses
  Run one teacher-forced forward pass
  Cache the terminal value $V(T)$
  for each retained index $k \in T$ **do**
    Cache

$$r(T_{<k}, k) \leftarrow \rho_\theta(a_k \mid q, A_{T_{<k}})$$

  end for
end for
// After enumeration, $r(H, k)$ is available for every $H \subseteq \{1, \ldots, k-1\}$
// Phase II: compute subset coefficients
for each index subset $T \subseteq N$ **do**
  Compute the induced probability of the retained/skipped pattern up to $m(T)$:

$$U(T) = \prod_{k=1}^{m(T)} r(T_{<k}, k)^{\mathbb{I}[k \in T]} \big(1 - r(T_{<k}, k)\big)^{\mathbb{I}[k \notin T]}$$

  Compute the stopping probability after T:

$$P_{\text{stop}}(T) = \prod_{k=m(T)+1}^{n} \big(1 - r(T, k)\big)$$

  Compute

$$\text{coef}(T) = U(T) P_{\text{stop}}(T) V(T)$$

end for
// Phase III: compute closed values and Sequential Shapley values
for each index subset $S \subseteq N$ **do**
  Compute

$$\tilde{V}(S) \leftarrow \frac{1}{U(S)} \sum_{T:\, S \preceq T} \text{coef}(T)$$

end for
for each player index $k \in N$ **do**
  Compute

$$\phi_{\text{seq}}(k) \leftarrow \sum_{S \subseteq N \setminus \{k\}} w(S) \big(\tilde{V}(S \cup \{k\}) - \tilde{V}(S)\big)$$

  where

$$w(S) = \frac{|S|!(n - |S| - 1)!}{n!}$$

end for
Optionally normalize $\{\phi_i^{\text{seq}}\}_{i=1}^n$ for advantage estimation during policy optimization.
return $\{\phi_i^{\text{seq}}\}_{i=1}^n$

context. Although Sequential Shapley is formulated as a general principle for ordered decision-making problems, extending it to broader agent settings may require task-specific designs of continuation modeling and counterfactual evaluation. For example, in environments with strong sequential constraints where future actions become nearly deterministic conditioned on previous states (i.e., $P(C \mid S) \approx 1$), defining meaningful counterfactual conditions may become more challenging.

Extending Sequential Shapley beyond search-agent scenarios is an important direction for future work.