BRANCHSPAN: CONTINUATION-INDEXED REPLAY FOR LONG-HORIZON WEB AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Two browser histories can arrive at nearly indistinguishable pages while leaving different actions executable because an earlier tab, filter, modal choice, or selection changed the active branch. This *branch aliasing* makes replay an addressing problem: chronological context preserves history and semantic retrieval finds relevant-looking states, but neither address determines whether recalled actions still fit the branch on screen. We introduce BranchSpan, which seals decision-coherent trajectory segments and indexes each endpoint by latent relevance, canonical reachable actions, and a history-conditioned locality-sensitive branch code; the selected segment enters an otherwise unchanged agent through a deterministic five-field payload. On the complete 812-task WebArena and 910-task VisualWebArena inventories, BranchSpan improves GPT-4o success over 128k Long-Context by +5.3 [3.6,7.0] and +4.4 [2.8,6.0] points, rising to +7.2 [4.6,9.8] and +5.9 [3.6,8.2] on long-horizon tasks. A validation-tuned Long-Context plus semantic-retrieval hybrid leaves BranchSpan ahead by +2.6 [1.1,4.1] and +2.3 [0.9,3.7] points overall. In a method-blind audit of method-native replay, annotators find valid continuations in 82.1% (197/240) and 76.7% (184/240) of sampled WebArena and VisualWebArena states; on the shared WebArena segment inventory, replacing continuation scoring with cosine-only retrieval reduces success by 1.9 points overall and 2.8 on long tasks, and reduces valid replay by 5.8 points. The frozen interface preserves the Long-Context $<$ hybrid $<$ BranchSpan ordering with Qwen2.5-VL-72B, while BranchSpan uses $0.21\times$ the Raw Frame replay payload with 40.3 ms encoder cost. For long-horizon agents, a useful memory address must recover not only relevant history, but history whose actions remain executable now.

1 INTRODUCTION

A shopping page revisited after choosing a size may look identical to its earlier version, even though the viable next action now depends on that choice. The same ambiguity appears when a dashboard is reached through different filters or a form is reopened after a modal decision. A web agent that retrieves the most similar earlier page can therefore recover an action that was appropriate on another route. We call this collision *branch aliasing*: observation similarity conceals disagreement in executable continuations.

Existing memory addresses emphasize either position or semantic proximity. Long context retains more of the trajectory but leaves selection to prompt position, while episodic retrieval promotes states that resemble the current one; evidence buried in long prompts can also be attenuated (Liu et al., 2024). Stateful browser environments make the missing coordinate consequential because tabs, selections, modal transitions, and prior submissions alter reachable actions without necessarily producing a visually distinct page (Zhou et al., 2024; Koh et al., 2024; Deng et al., 2023). Retention, relevance, and current executability are thus separate properties of replay.

BranchSpan turns executability into an explicit address coordinate. A multimodal encoder represents the DOM, accessibility tree, screenshot, and recent canonical action prefix. A learned detector seals segments at decision events; each endpoint is indexed by a canonical reachable-action set and a locality-sensitive code of interface structure and recent anchor actions. Retrieval combines those coordinates with latent relevance, then renders the chosen trajectory evidence as span, budget progress, affected elements, branch code, and anchor actions. All fields come from the current episode prefix,

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.

and the planner, prompt scaffold, browser tools, action vocabulary, stopping rule, tasks, and resets remain unchanged.

This fixed-agent contract produces a consistent outcome hierarchy. BranchSpan exceeds 128k Long-Context by 5.3/4.4 points overall and 7.2/5.9 points on long WebArena/VisualWebArena tasks. Even after Long-Context receives validation-tuned semantic retrieval, the paired overall gaps remain +2.6 [1.1,4.1] and +2.3 [0.9,3.7]. The clean address intervention fixes every other replay component: on GPT-4o WebArena, changing only continuation scoring to cosine retrieval on the same encoder, segments, five-field payload, allocation, and depth costs 1.9 success points overall, 2.8 on long tasks, and 5.8 points of independently judged replay validity. Method-blind native-unit audits on both suites, separate token and latency operating regimes, and frozen-interface transfer to Qwen2.5-VL-72B (Bai et al., 2025) reinforce the practical consequence: memory improves when it retrieves actions that still belong to the realized route.

The paper contributes:

- **A replay failure model:** branch aliasing identifies executable continuation, rather than retained volume or topical relevance, as the missing coordinate in stateful single-trajectory memory.
- **A continuation-indexed interface:** BranchSpan combines learned decision segmentation, canonical action compatibility, history-conditioned branch structure, and deterministic prefix-only serialization.
- **A controlled evidence hierarchy:** complete two-suite results establish the package-level gain, a shared-inventory WebArena intervention isolates continuation scoring, and blinded continuation judgments, resource operating points, and a second outer agent test the same design consequence from complementary views.

2 Related Work: Memory Units and Addresses

Classic replay stores transitions for policy learning, episodic control retrieves nearby experience, and retrieval-augmented models use latent similarity (Lin, 1992; Schaul et al., 2016; Blundell et al., 2016; Pritzel et al., 2017; Borgeaud et al., 2022; Wu et al., 2022). Summary and reflection memories compress history into prose (Shinn et al., 2023; Park et al., 2023). These approaches vary the stored unit and usually address it by recency or relevance. BranchSpan adds a third coordinate—whether stored actions remain reachable after the current action prefix—and tests it against cosine retrieval on the same learned segment inventory and visible payload.

WebArena and VisualWebArena expose route-dependent browser state; Mind2Web, WorkArena, SeeAct, and WebVoyager cover offline traces, enterprise workflows, multimodal grounding, and end-to-end control (Zhou et al., 2024; Koh et al., 2024; Deng et al., 2023; Drouin et al., 2024; Zheng et al., 2024; He et al., 2024). Recent web-memory systems operate at broader levels: AgentFold edits context, M^2 combines recursive summaries with cross-trajectory insights, AgentSwing routes among context strategies with lookahead, SAM learns access to raw trajectory pages, and ACGM builds a task-adaptive graph over multimodal history (Ye et al., 2025; Yan et al., 2026; Feng et al., 2026; Hu et al., 2026; Forouzandeh et al., 2026). Branch-and-Browse joins tree exploration, state restoration, and page-action memory (He et al., 2026). BranchSpan complements these controller, experience-source, and memory-graph choices by isolating the address applied to one realized episode under a fixed controller.

Options, hierarchical reinforcement learning, world models, and offline sequence policies alter temporal control or the policy itself (Sutton et al., 1999; Nachum et al., 2018; Parr & Russell, 1997; Ha & Schmidhuber, 2018; Hafner et al., 2020; 2023; Chen et al., 2021; Janner et al., 2021; Kumar et al., 2020). BranchSpan uses decision boundaries only to decide which realized actions re-enter the existing agent prompt; it neither executes learned options nor predicts counterfactual dynamics. Appendix F compares the resulting control, replay-unit, and address distinctions.

3 BRANCHSPAN: CONTINUATION-INDEXED REPLAY

3.1 FAILURE MODEL AND OBSERVABLE CONTINUATION ADDRESSES

We model web interaction as a POMDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{O}, T, R, \Omega, \gamma)$. Observation $o_t = (d_t, u_t, s_t)$ contains the DOM, accessibility tree, and screenshot; $h_t = (a_1, \ldots, a_{t-1})$ is the observed action prefix. For evaluation, $\mathcal{C}_t$ contains every next action that is executable in the displayed interface and admits a task-consistent continuation without undoing constraints already established by h_t. Annotators construct this set from raw state/action evidence and may accept multiple actions. It never enters inference; online replay uses the observable reachable-action proxy $\mathcal{A}_t^{\text{reach}}$ defined below.

Definition 1 (Web trajectory)**.** *A trajectory $\tau = (o_1, a_1, r_1, \ldots, o_T, a_T, r_T)$ records one task interaction, with T ranging from 5 to more than 100 steps.*

Definition 2 (Branch aliasing)**.** *States (o, h) and (o', h') are branch-aliased when their rendered observations are similar but their canonical valid-continuation sets differ because the action prefixes produced different reachable futures.*

Definition 3 (Observable continuation address)**.** *The address $\chi_t = (z_t, \mathcal{A}_t^{\text{reach}}, \beta_t)$ combines a latent state key, the currently reachable action signature, and a locality-sensitive code of the realized DOM/accessibility branch and recent anchors. It estimates which stored actions remain executable from agent-observable state.*

We canonicalize a browser action as

$$q(a) = (\text{type}, \text{role}, \text{name}, \text{path}, \text{argclass}). \tag{1}$$

Type is one of click, type, select, scroll, submit, navigate, or browser-tab control. Interface tabs are clicks with role `tab`; browser-level navigation and tab operations replace role/name/path by the command family and an origin-normalized target. Otherwise, role and whitespace-normalized accessible name identify semantics; path starts at the nearest stable ID/name ancestor and then records tag, role, and sibling rank, after stripping session-specific numeric suffixes. Argument classes are free-text, number, date, Boolean, an enumerated option, or a browser-control target; literal typed text is retained only in the executed anchor payload. Duplicate keys collapse to a set, making $\mathcal{A}_t^{\text{reach}} = \{q(a) : a \text{ is exposed and currently legal at } t\}$ finite. If stable attributes are absent, the viewport-rooted role/tag path is used.

Definition 4 (BranchSpan segment)**.** *A segment $\sigma_i = (z_i^{\text{start}}, z_i^{\text{end}}, \mathcal{I}_i, \mathcal{P}_i)$ contains encoded endpoints, a non-prompt retrieval index $\mathcal{I}_i = (z_i^{\text{end}}, \mathcal{A}_i^{\text{reach}}, \beta_i)$, and a prompt payload $\mathcal{P}_i = (\mathcal{T}_i, p_i, \mathcal{E}_i, \beta_i, \mathcal{U}_i)$. Here $\mathcal{T}_i = ((o_t, a_t))_{t=t_i^s}^{t_i^e}$ is the stored action–observation excerpt with explicit span bounds. The other four payload fields are budget progress, affected element identifiers, branch code, and executed anchor actions. The reachable-action signature is index-only and is never rendered to the outer agent.*

Definition 5 (Continuation-indexed segmentation)**.** *A segmentation is continuation-indexed when each sealed segment exposes an address that supports retrieval simultaneously by task relevance, current action compatibility, and realized-branch agreement.*

3.2 MULTIMODAL ENCODING AND DECISION BOUNDARIES

Each state is encoded as $z_t = f_\theta(o_t, \bar{h}_t) \in \mathbb{R}^{768}$ by a 12-layer, 12-head transformer, where $\bar{h}_t$ is the sequence of up to four most recent canonical anchor actions since the previous boundary. Linearized DOM tokens include depth and sibling-order embeddings; accessibility nodes retain role, name, and state attributes; screenshot patches come from a frozen ViT-B/16 (Dosovitskiy et al., 2021); and $\bar{h}_t$ enters as typed action tokens. Cross-modal attention produces the fused state.

The two-layer boundary detector consumes consecutive states and a discretized action-set change:

$$b_t = g_\phi([z_t; z_{t-1}], \Delta_t^{\text{act}}), \qquad \Delta_t^{\text{act}} = \frac{|\mathcal{A}_t^{\text{reach}} \triangle \mathcal{A}_{t-1}^{\text{reach}}|}{|\mathcal{A}_t^{\text{reach}} \cup \mathcal{A}_{t-1}^{\text{reach}}|}. \tag{2}$$

We set $\Delta_t^{\text{act}} = 0$ when both action sets are empty and 1 when exactly one is empty. Its training target y_t is one when the training-side browser event stream records a page transition, completed option,

modal/branch change, or an explicit plan-revision event, and zero otherwise. These prefix-observable event rules are the supervision source; the separately human-labeled diagnostic trajectories are not used for fitting. The detector minimizes class-balanced binary cross entropy

$$\mathcal{L}_{\text{bdry}} = -\sum_t w_{y_t} \left[y_t \log b_t + (1 - y_t) \log(1 - b_t) \right]. \tag{3}$$

A boundary is emitted when $b_t > 0.56$ or when a segment reaches 12 steps. The threshold is selected for boundary F1 on the training-side validation partition, then frozen.

3.3 Learning action-bearing addresses

The encoder is trained with inverse-dynamics, reachable-action, branch-code prediction, and contrastive objectives:

$$\mathcal{L}_{\text{inv}} = -\mathbb{E}_\tau \sum_t \log p_\psi(a_t \mid z_t, z_{t+1}), \tag{4}$$

$$\mathcal{L}_{\text{reach}} = \mathbb{E}_\tau \sum_t \text{BCE}\Big(\hat{A}_\xi(z_t), \mathbf{1}[\mathcal{A}_t^{\text{reach}}] \Big), \tag{5}$$

$$\mathcal{L}_{\text{branch}} = \mathbb{E}_\tau \sum_t \text{BCE}(\text{sigmoid}(W_\omega z_t), \beta_t), \tag{6}$$

$$\mathcal{L}_{\text{con}} = -\mathbb{E} \log \frac{\exp(z \cdot z^+/\kappa)}{\exp(z \cdot z^+/\kappa) + \sum_j \exp(z \cdot z_j^-/\kappa)}. \tag{7}$$

The branch target is a 256-bit locality-sensitive code, so $B = 256$. Let c_t be the canonical preorder serialization of the smallest subtree containing the focused element and all currently enabled action targets; when no element is focused, the visible viewport root is used. Nodes retain normalized tag, role, accessible name, state flags, and action keys. A sparse feature vector v_t counts features from c_t, $\mathcal{A}_t^{\text{reach}}$, and the last four canonical anchor actions since the previous boundary. With a fixed seeded signed-projection matrix R,

$$\beta_t = \mathbf{1}[R v_t \geq 0]. \tag{8}$$

This signed random-projection code is locality-sensitive: normalized Hamming agreement estimates feature-space angular agreement rather than comparing cryptographic digests. Query and stored endpoint codes are both computed directly by Eq. 8; the prediction head in $\mathcal{L}_{\text{branch}}$ is an auxiliary representation objective, not the source of the retrieval code. The canonical byte string is retained in the non-prompt index, so distinct strings with the same 256-bit code remain separate entries; for that exact-code collision, s_β is capped at $(B-1)/B$ and the latent/action terms decide the ranking. If the active subtree is missing, its mask removes s_β and the remaining retrieval weights are renormalized.

The four encoder objectives shape different coordinates of the continuation address. Inverse dynamics preserves action-bearing transitions, reachable-action prediction exposes current affordances, and branch-code prediction retains continuation-relevant structure and recent route information. Positives are re-renderings of the same page state under the same canonical action prefix; hard negatives have similar appearance and different reachable-action sets (Chen et al., 2020). The total encoder loss is $\mathcal{L} = \mathcal{L}_{\text{inv}} + 0.5\mathcal{L}_{\text{reach}} + 0.3\mathcal{L}_{\text{branch}} + 0.2\mathcal{L}_{\text{con}}$.

3.4 Address compatibility and prompt interface

For query state t and sealed segment i, define $s_z = (1 + \cos(z_t, z_i^{\text{end}}))/2$, $s_A = |\mathcal{A}_t^{\text{reach}} \cap \mathcal{A}_i^{\text{reach}}| / |\mathcal{A}_t^{\text{reach}} \cup \mathcal{A}_i^{\text{reach}}|$, and $s_\beta = 1 - \text{Hamming}(\beta_t, \beta_i)/B$. Retrieval uses

$$S(t, i) = 0.45 s_z + 0.35 s_A + 0.20 s_\beta. \tag{9}$$

If both action sets are empty, $s_A = 1$; if only one is empty, $s_A = 0$. The query code uses (o_t, h_t), while segment code $\beta_i = \beta_{t_i^e}$ and $\mathcal{A}_i^{\text{reach}} = \mathcal{A}_{t_i^e}^{\text{reach}}$ are computed once when the boundary seals the segment. No stored field is updated using a later state. Ties are resolved by the latest segment end and then segment index. Up to five segments with $S(t, i) \geq 0.35$ are returned. If no segment reaches the threshold, the replay block is empty.

The buffer contains only sealed segments from the current episode prefix and is reset before every task. A segment becomes retrievable after its closing boundary; the open segment, future states, successful reference trajectories, and evaluation annotations never enter inference. The buffer holds at most 32 segments (median 18); when full, it evicts the oldest segment with zero current reachable-action overlap ($s_A = 0$), falling back to the oldest segment. Segments contain at most 12 steps with median length 6. Index fields use endpoint t_i^e. The ordered set $\mathcal{E}_i$ contains unique canonical keys for elements targeted or changed during the span, and $\mathcal{U}_i$ contains the executed anchor actions in temporal order. Progress is $p_i = \min(1, t_i^e / T_{\max})$, where $T_{\max}$ is the fixed episode step cap; it uses neither rewards nor completion labels.

The prompt-visible payload has exactly five fields in one order: the action–observation excerpt with its span bounds, budget progress, affected elements, branch code, and anchor actions. The outer-model tokenizer gives a median 94-token header and a 160-token cap. Overflow removes oldest anchor actions and then oldest affected-element entries, while span bounds, progress, and branch code remain. Payloads are processed in score order under the validation-fixed replay allocation; if the final excerpt exceeds the residual allocation, oldest complete action–observation pairs are removed, and an item with no complete pair remaining is omitted. The fixed outer agent receives its ordinary current observation followed by this replay block. Long-Context receives a chronological raw window; episodic methods receive raw steps or summaries. The strongest hybrid retrieves completed raw steps by semantic similarity and appends them to a chronological window; it receives neither branch codes nor reachable-action signatures. Appendix A gives the complete information-access contract and edge-case behavior.

4 Experiments

The evaluation separates package-level utility from address-level attribution. The full replay comparison asks whether continuation-indexed segments improve completion beyond chronology and semantic retrieval. A shared-inventory intervention then changes only the address score, while blinded continuation judgments test the applicability of delivered evidence. Transfer and resource operating points measure how the package behaves when the outer model or serving constraint changes.

4.1 Setup

WebArena (Zhou et al., 2024) contributes the complete 812-task reporting inventory across shopping, CMS, Reddit, GitLab, and maps, including 247 long-horizon tasks requiring at least 25 actions. VisualWebArena (Koh et al., 2024) contributes the complete 910-task inventory across classifieds, Reddit, and shopping, including 284 requiring at least 20 actions. Deterministic reset scripts restore every self-hosted site before each episode.

The primary outer agent is the pinned GPT-4o-2024-05-13 snapshot (OpenAI, 2024) with temperature 0, a common anonymized ReAct prompt, action vocabulary, browser interface, retry policy, and stopping rule. Baselines are: No Replay; Raw Frame Replay, a recency queue of full multimodal steps; Text Summary Replay, a temperature-zero GPT-4o summary updated after each completed step with one frozen instruction (Shinn et al., 2023); Episodic Buffer, cosine retrieval of completed raw steps (Pritzel et al., 2017); Option-Level Replay, event-delimited spans ranked by endpoint goal similarity; 128k Long-Context, a suffix of raw trajectory tokens; and Long-Context + retrieval. Option spans close under the same page-transition, completed-option, and modal-change event rules used to construct boundary labels; their labels and actions are serialized chronologically. For Episodic Buffer and the hybrid, the query concatenates the current task and observation, and each candidate is one completed raw action–observation step. The frozen endpoint encoder embeds both; cosine similarity ranks candidates without a score threshold, and the top $\min(k, N)$ completed steps are serialized chronologically. The hybrid appends them after a chronological suffix and has no access to BranchSpan segments, branch codes, reachable-action signatures, or typed payloads.

Hybrid allocation is selected from raw-window sizes $\{32, 64, 96, 128\}$k and retrieval depths $\{1, 3, 5\}$ by validation success subject to the active cap. The headline/latency point uses a 64k raw suffix and up to five retrieved steps (82.6k complete mean prompt); the 40k-token point uses 32k and up to three steps (39.1k). Before the first completed step, its retrieval addition is empty and the chronological

Table 1: Task success rate (%) when the same outer agent receives different replay units and addresses. "Long" means at least 25 actions on WebArena (WA; 247 tasks) and 20 on VisualWebArena (VWA; 284 tasks). Means and standard errors use three seeds over 812/910 tasks. The hybrid adds a validation-tuned episodic retriever to Long-Context. Paired 95% intervals hierarchically resample tasks and seeds.

Method	WA All	WA Long	VWA All	VWA Long
No Replay	12.3±0.7	5.7±1.1	9.1±0.6	4.2±0.9
Raw Frame Replay	14.8±0.9	8.1±1.2	11.4±0.8	6.1±1.1
Text Summary Replay	17.2±0.8	10.4±1.1	13.8±0.7	8.3±1.0
Episodic Buffer	18.6±1.0	12.8±1.4	14.9±0.9	10.1±1.3
Option-Level Replay	17.9±0.9	11.2±1.2	14.2±0.8	8.8±1.1
Long-Context (128k)	20.1±0.7	13.9±1.3	16.3±0.8	11.4±1.2
Long-Context + retrieval	22.8±0.8	16.7±1.2	18.4±0.8	13.8±1.1
BranchSpan	25.4±0.8	21.1±1.1	20.7±0.9	17.3±1.2
BranchSpan − Long-Context, 95% CI	+5.3 [3.6,7.0]	+7.2 [4.6,9.8]	+4.4 [2.8,6.0]	+5.9 [3.6,8.2]

suffix is unchanged. This is the strongest semantic-address baseline under the same outer-agent contract.

BranchSpan is implemented in PyTorch 2.3.1 and CUDA 12.4. Its 12,000 training-side WebArena/MiniWoB++ trajectories (Liu et al., 2018) are grouped by (suite, site, intent template), deduplicated by normalized goal and initial-state signature, and divided into 10,800 training and 1,200 validation trajectories. Any group sharing a normalized goal template with the 812/910 reporting tasks is excluded before splitting; VisualWebArena reporting trajectories are never used for fitting. The encoder runs for 50k pretraining steps followed by 20k AdamW steps at 3×10^{-4} with 2k-step warmup and cosine decay. The validation partition selects the boundary threshold, retrieval weights and threshold, buffer size, and baseline allocations once. Evaluation tasks, audit states, and task outcomes do not affect optimization or selection. Training takes 18.4 hours on 8×A100 80GB GPUs.

Headline results average three seeds; for both outer agents, a seed fixes task order, tie resolution, and backend decoding randomness under temperature-zero decoding, while task inventories and reset states remain paired across methods. ± is the standard error across seed-level rates. Paired 95% intervals use 10,000 hierarchical bootstrap replicates that resample task IDs and then the three within-task seeds, preserving method pairs. Pooled task-seed counts are reported alongside rates.

The deterministic affordance extractor enumerates the canonical exposed, currently legal actions in Eq. 1. Branch Err. compares top-1 branch identity with one successful reference suffix and is reported only as a method-coupled diagnostic. The independent audit uses disjoint task IDs and one sampled decision state per task, shared by all methods. It hides method identity, typed fields, and reference suffixes, and permits every continuation supported by raw state/action evidence.

4.2 Outcome effect under a fixed agent

BranchSpan leads every baseline in Table 1. Replacing chronological raw-history addressing with continuation indexing raises WebArena success from 20.1 to 25.4 and VisualWebArena from 16.3 to 20.7. Adding semantic retrieval to Long-Context recovers part of the difference, yet the paired BranchSpan advantages remain +2.6 [1.1,4.1] and +2.3 [0.9,3.7] points overall. Pooled successful task-seed episodes for BranchSpan, Long-Context, and the hybrid are, respectively, 619/2436, 490/2436, and 555/2436 on WebArena, and 565/2730, 445/2730, and 502/2730 on VisualWebArena. The corresponding long-horizon counts are 156/741, 103/741, and 124/741 on WebArena, and 147/852, 97/852, and 118/852 on VisualWebArena. Table 3 supplies the matched address intervention behind this package-level result: on the shared learned-segment inventory, cosine-only selection lowers WebArena success from 25.4 to 23.5 overall and from 21.1 to 18.3 on long tasks, while validity of the selected raw segment falls from 82.1% to 76.3%. The encoder, candidate segments, five-field payload, allocation rule, and top-five depth are identical; only Eq. 9 is replaced by latent cosine.

Table 2: WebArena operating points for two separate constraints. Decision latency is the pooled median warm-cache time from the start of local replay selection through the first syntactically complete browser action; prompt tokens are per-step means. The fixed 38.6k-token, 887 ms BranchSpan point appears in both blocks without retuning. Appendix B separately reports replay payload.

Cap	Method	SR All	SR Long	Tok./step	ms/step
Median decision latency ≤ 890 ms	Long-Context (128k)	20.1±0.7	13.9±1.3	127.4k	847
	Episodic Buffer	18.6±1.0	12.8±1.4	41.3k	391
	Long-Context + retrieval	22.8±0.8	16.7±1.2	82.6k	861
	BranchSpan (fixed point)	25.4±0.8	21.1±1.1	38.6k	887
Mean prompt tokens ≤ 40k	Long-Context (40k)	17.6±0.9	10.8±1.4	38.2k	412
	Episodic Buffer ($k = 3$)	17.1±1.0	11.4±1.5	37.8k	378
	Long-Context + retrieval	20.5±0.9	14.1±1.3	39.1k	498
	BranchSpan (same fixed point)	25.4±0.8	21.1±1.1	38.6k	887

Table 3: Two complementary continuation tests. Panel A uses GPT-4o WebArena and a shared learned-segment inventory; all rows preserve the encoder, candidates, task states, top-five depth, and allocation rule. Cosine-only changes only the address score; removing the action coordinate drops s_A while preserving the five-field payload; removing the branch code drops s_β and its visible payload field. Panel B audits each complete method's native replay on one shared state from each of 240 WebArena and 240 VisualWebArena task IDs.

Panel A: shared-inventory WebArena intervention

Variant	SR All	SR Long	Valid top-1 segment
BranchSpan	25.4±0.8	21.1±1.1	82.1% (197/240)
Cosine-only address	23.5±0.8	18.3±1.2	76.3% (183/240)
Without action coordinate	23.2±0.9	17.9±1.3	74.6% (179/240)
Without branch code	23.8±0.9	18.6±1.2	75.8% (182/240)

Panel B: independent valid-continuation audit

Method	WA valid	WA Wilson CI	VWA valid	VWA Wilson CI
Episodic Buffer	64.2% (154/240)	[57.9,70.0]	60.4% (145/240)	[54.1,66.4]
Long-Context	68.8% (165/240)	[62.6,74.3]	64.6% (155/240)	[58.3,70.4]
Long-Context + retrieval	74.6% (179/240)	[68.7,79.7]	70.0% (168/240)	[63.9,75.4]
BranchSpan	82.1% (197/240)	[76.7,86.4]	76.7% (184/240)	[70.9,81.6]

4.3 Cost of selecting applicable history

Latency is measured with one streamed API request in flight, provider prefix caching enabled for the invariant system/task prefix, one warm-up request per task, and identical cache policy across methods. The interval includes local retrieval, serialization, encoder execution, network service, and generation through the first complete action; page execution, loading, and reset are outside it. Each task-seed contributes its median decision, and the table pools those medians. Component profiling assigns 40.3 ms to BranchSpan encoding, 4.8% of the 847.2 ms Long-Context median. Under the median-latency constraint, BranchSpan gains 5.3/7.2 points over Long-Context. Under the separate mean-token constraint, it exceeds the 39.1k-token hybrid by 4.9/7.0 points while using 38.6k tokens. These are independent serving views rather than a joint token–latency constraint.

4.4 Address intervention and continuation applicability

Panel A gives the matched address-level estimate: replacing the complete score by cosine alone changes no representation, candidate unit, visible field, or allocation, yet reduces SR All/Long by 1.9/2.8 points and valid selected replay by 5.8 points. Removing the index-only reachable-action coordinate is also an address intervention; removing the branch code is broader because the code participates in both ranking and the visible payload.

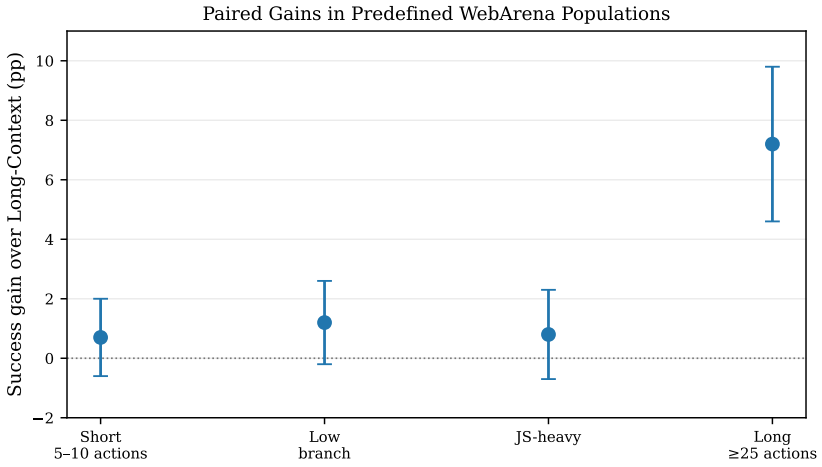


Figure 1: Paired success-rate gains over Long-Context in four predefined, overlapping WebArena populations. Bars are 95% task/seed hierarchical-bootstrap intervals; each population is interpreted independently.

Panel B asks whether the evidence delivered by each complete replay interface supports a valid continuation. All methods use the same query states, stratified by horizon and prior branch events. Episodic Buffer and the hybrid contribute their highest-ranked raw step, Long-Context contributes the latest complete action–observation pair in its visible window, and BranchSpan contributes its highest-scored raw segment with typed fields removed. A candidate is valid when its raw evidence contains an action or transition that is executable at the query state and consistent with the task goal and established constraints. Candidates are randomly ordered and shown without method identity; empty retrieval is unsupported. Annotators train on 20 held-out examples, label independently, and adjudicate after the blinded pass. Paired gap intervals resample the shared task states.

This native-unit audit yields $\kappa = 0.79$ before adjudication. BranchSpan exceeds the hybrid by +7.5 [2.5,12.4] points on WebArena and +6.7 [1.7,11.5] on VisualWebArena. Panel B establishes the applicability of the complete delivered replay, while Panel A isolates the score on a common unit; both agree with end-task success.

4.5 Transfer, component dependence, and operating regimes

Without retraining the replay encoder, a Qwen2.5-VL-72B ReAct agent served in bfloat16 with tensor parallelism across 8 A100 80GB GPUs preserves the Long-Context $<$ hybrid $<$ BranchSpan ordering on both suites. Relative to the hybrid, continuation indexing gains +2.7 / +3.2 points on WebArena All/Long and +2.3 / +3.1 on VisualWebArena. The paired intervals are +2.7 [1.1,4.3] / +3.2 [0.8,5.8] and +2.3 [0.9,3.7] / +3.1 [0.8,5.4], respectively; Appendix D reports means, standard errors, and per-seed counts. The unchanged encoder and interface preserve the ordering under a materially different outer model.

Component interventions explain where the outcome difference enters. Replacing learned boundaries with fixed cuts reduces WebArena SR All by 3.6 points; random cuts reduce it by 6.2; removing branch-code prediction costs 2.8. On the shared inventory, removing the index-only reachable-action coordinate or changing only the scorer to cosine affects selection, whereas removing the branch code also changes the visible payload (Appendix C).

Four predefined WebArena operating regimes provide complementary views of the aggregate gain. Short tasks contain 5–10 reference actions, low-branch tasks at most two page/modal/option events in the benchmark trace, and JS-heavy tasks a majority of action targets introduced after client-side DOM mutation. These overlapping populations yield gains over Long-Context of +0.7 points on 132 short tasks with [-0.6,2.0] paired interval, +1.2 on 241 low-branch tasks with [-0.2,2.6], and +0.8 on 176 JS-heavy tasks with [-0.7,2.3]. The long-horizon population gains 7.2 points, [4.6,9.8]. Figure 1 presents each population independently.

5 Discussion

The experiments support two distinct conclusions. At the package level, continuation-indexed replay improves completion over chronological and semantic-retrieval alternatives on both suites, and the ordering persists under a second outer model. At the address level, the GPT-4o WebArena shared-inventory intervention fixes representation, candidates, payload, allocation, and depth; its 1.9-point overall, 2.8-point long-task, and 5.8-point validity differences provide matched point estimates for continuation scoring. Learned boundaries contribute separately by keeping decision-coherent evidence together.

The audit clarifies the practical consequence. Native BranchSpan replay contains a valid next continuation more often than the strongest hybrid on matched states in both suites, and the exact same-unit score intervention preserves that direction on WebArena. The large long-horizon gain, compact replay payload, and frozen Qwen interface together make executable continuation a concrete design target rather than a proxy for context volume.

The evaluated regime is single-trajectory replay from agent-observable DOM, accessibility, screenshot, and action-prefix state on self-hosted WebArena and VisualWebArena. Within that regime, two outer agents, a semantic-retrieval hybrid, matched address interventions, and independent continuation judgments establish the result. Browser state outside those observations remains outside the address, canonical matching depends on exposed interface elements, and the resource comparison reports warm-cache median decision latency on WebArena. This scope preserves the central implication: continuation compatibility accounts for a residual that additional context and semantic relevance do not absorb in the tested fixed interfaces.

6 Conclusion

BranchSpan retrieves realized web experience by latent relevance, reachable actions, and history-conditioned branch structure. It surpasses both 128k chronological rereading and retrieval-augmented long context on WebArena and VisualWebArena, delivers more valid method-native replay, preserves its package ordering with Qwen2.5-VL-72B, and uses $0.21\times$ the Raw Frame payload. On the matched GPT-4o WebArena inventory, the score-only intervention produces point differences of 1.9 overall success points, 2.8 long-task points, and 5.8 points of valid replay. The resulting design principle is direct: long histories become useful when retrieval distinguishes which remembered actions still belong to the branch now on screen.

Reproducibility Statement

Sections 3 and 4 specify action and branch canonicalization, boundary supervision, objectives, retrieval, split isolation, baseline allocations, caps, metrics, annotation, and uncertainty; Appendix A records the per-baseline information-access contract and complete result details.

AI Use Statement

Generative language tools assisted with drafting and editing portions of this manuscript, including presentation, title and frontmatter revision, and formatting. This statement is limited to the documented manuscript-preparation record.

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Charles Blundell, Benigno Uria, Alexander Pritzel, Yazhe Li, Avraham Ruderman, Joel Z Leibo, Jack Rae, Daan Wierstra, and Demis Hassabis. Model-free episodic control. *arXiv preprint arXiv:1606.04460*, 2016. URL https://arxiv.org/abs/1606.04460.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 2206–2240, 2022. URL https://proceedings.mlr.press/v162/borgeaud22a.html.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. In *Advances in Neural Information Processing Systems*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, 2020.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2Web: Towards a generalist agent for the web. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2306.06070.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=MzTdZhMjeC.

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H Laradji, Manuel Del Verme, Tom Marty, David Vazquez, Nicolas Chapados, and Alexandre Lacoste. WorkArena: How capable are web agents at solving common knowledge work tasks? In *International Conference on Machine Learning*, pp. 11642–11662, 2024.

Zhaopeng Feng, Liangcai Su, Zhen Zhang, Xinyu Wang, Xiaotian Zhang, Xiaobin Wang, Runnan Fang, Qi Zhang, Baixuan Li, Shihao Cai, Rui Ye, Hui Chen, Yong Jiang, Joey Tianyi Zhou, Chenxiong Qian, Pengjun Xie, Bryan Hooi, Zuozhu Liu, and Jingren Zhou. AgentSwing: Adaptive parallel context management routing for long-horizon web agents. *arXiv preprint arXiv:2603.27490*, 2026. URL https://arxiv.org/abs/2603.27490.

Saman Forouzandeh, Kamal Berahmand, and Mahdi Jalili. Task-adaptive retrieval over agentic multimodal web histories via learned graph memory. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3727–3732, 2026. doi: 10.1145/3805712.3809978. URL https://doi.org/10.1145/3805712.3809978.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, pp. 2451–2463, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1912.01603.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. WebVoyager: Building an end-to-end web agent with large multimodal models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6864–6890, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.371. URL https://aclanthology.org/2024.acl-long.371/.

Shiqi He, Yue Cui, Xinyu Ma, Yaliang Li, Bolin Ding, and Mosharaf Chowdhury. Branch-and-browse: Efficient and controllable web exploration with tree-structured reasoning and action memory. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 18407–18418, 2026. doi: 10.18653/v1/2026.acl-long.838. URL https://aclanthology.org/2026.acl-long.838/.

Yuyang Hu, Hongjin Qian, Shuting Wang, Jiongnan Liu, Ziliang Zhao, Jiejun Tan, Zheng Liu, and Zhicheng Dou. SAM: State-adaptive memory for long-horizon reasoning agent. *arXiv preprint arXiv:2605.24468*, 2026. URL https://arxiv.org/abs/2605.24468.

Michael Janner, Qiyang Li, and Sergey Levine. Offline reinforcement learning as one big sequence modeling problem. In *Advances in Neural Information Processing Systems*, 2021.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. VisualWebArena: Evaluating multimodal agents on realistic visual web tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 881–905, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.50. URL https://aclanthology.org/2024.acl-long.50/.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-Learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, 2020.

Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine Learning*, 8(3–4):293–321, 1992.

Evan Zheran Liu, Kelvin Guu, Panupong Pasupat, Tianlin Shi, and Percy Liang. Reinforcement learning on web interfaces using workflow-guided exploration. In *International Conference on Learning Representations*, 2018.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

Ofir Nachum, Shixiang Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. In *Advances in Neural Information Processing Systems*, 2018.

OpenAI. GPT-4o system card. https://cdn.openai.com/gpt-4o-system-card.pdf, 2024. Accessed 2026-04-24.

Joon Sung Park, Joseph C O'Brien, Carrie J Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Annual ACM Symposium on User Interface Software and Technology*, 2023.

Ronald Parr and Stuart J. Russell. Reinforcement learning with hierarchies of machines. In *Advances in Neural Information Processing Systems*, pp. 1043–1049, 1997.

Alexander Pritzel, Benigno Uria, Sriram Srinivasan, Adrià Puigdomènech Badia, Oriol Vinyals, Demis Hassabis, Daan Wierstra, and Charles Blundell. Neural episodic control. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 2827–2836, 2017. URL https://proceedings.mlr.press/v70/pritzel17a.html.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. In *International Conference on Learning Representations*, 2016.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, 2023.

Richard S Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1–2):181–211, 1999.

Yuhuai Wu, Markus N Rabe, DeLesley Hutchins, and Christian Szegedy. Memorizing transformers. In *International Conference on Learning Representations*, 2022.

Dawei Yan, Haokui Zhang, Guangda Huzhang, Yang Li, Yibo Wang, Qing-Guo Chen, Zhao Xu, Weihua Luo, Ying Li, Wei Dong, and Chunhua Shen. M^2: Dual-memory augmentation for long-horizon web agents via trajectory summarization and insight retrieval. *arXiv preprint arXiv:2603.00503*, 2026. URL https://arxiv.org/abs/2603.00503.

Rui Ye, Zhongwang Zhang, Kuan Li, Huifeng Yin, Zhengwei Tao, Yida Zhao, Liangcai Su, Liwen Zhang, Zile Qiao, Xinyu Wang, Pengjun Xie, Fei Huang, Siheng Chen, Jingren Zhou, and Yong Jiang. AgentFold: Long-horizon web agents with proactive context management. *arXiv preprint arXiv:2510.24699*, 2025.

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. GPT-4V(ision) is a generalist web agent, if grounded. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 61349–61385, 2024. URL https://proceedings.mlr.press/v235/zheng24e.html.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=oKn9c6ytLx.

A Replay Addressing and Information-Access Contract

Table 4: Fixed-agent replay contract. Every method receives the same current observation, prompt scaffold, action interface, stopping rule, and task reset. The address column identifies how stored experience is selected for the prompt. "Allocation" describes the replay portion; per-step cost excludes shared outer-model computation.

Method	Replay unit	Memory address and serialization	Allocation and incremental cost
No Replay	None	None	No replay tokens; constant assembly
Raw Frame	Full multimodal step	Recency; raw DOM, accessibility, screenshot references, and actions	Chronological cap; linear in retained steps
Text Summary	Free-form summary	Recency; compressed semantic text	Summary block; linear generation and compact prompt
Episodic Buffer	Completed raw step	Goal/current-state cosine; chronological order after top-k	Up to five items; empty block if none
Long-Context	Raw trajectory tokens	Chronological position; truncated raw history	Remaining window; linear prompt growth
Option-Level	Option spans	Active-goal similarity; option labels and actions	Up to five spans; boundary inference
Long-Context + retrieval	Raw window plus raw steps	Chronology plus goal/current-state cosine	64k+k=5 or 32k+k=3; validation-selected
BranchSpan	Sealed decision segment	Non-prompt latent/action/branch index; five-field visible payload	Up to five snippets; deterministic truncation and masked-coordinate renormalization

All methods see the same current observation outside the replay block. Candidate stores contain only the current episode prefix and reset at task boundaries. Semantic candidates are completed raw steps; their addition is empty only before the first candidate exists. BranchSpan candidates are sealed decision segments, and its addition is empty when no segment clears the validation-fixed retrieval threshold. Its reachable-action signature is used only by the non-prompt address, whereas the branch

code appears in both the address and the five-field payload. The shared-inventory control changes the score over identical BranchSpan candidates; the native-unit audit instead evaluates the evidence each complete interface actually delivers. Both protocols use matched query states and exclude future or replacement evidence.

B Outcome and Systems Details

Table 5: Outcome-side systems metrics corresponding to Table 1. Avg. Steps is computed over all episodes. Branch Err. is lower-is-better. Replay footprint is the serialized replay block normalized to Raw Frame (=1.00×), whereas Table 2 reports the complete prompt tokens per step.

	WebArena			VisualWebArena		
Method	Steps	Branch Err.	Footprint	Steps	Branch Err.	Footprint
No Replay	44.1	—	—	47.8	—	—
Raw Frame Replay	38.4	38.2	1.00×	41.2	41.7	1.00×
Text Summary Replay	35.7	31.6	0.12×	38.9	34.3	0.12×
Episodic Buffer	34.2	27.3	0.45×	37.1	29.8	0.45×
Option-Level Replay	36.1	29.8	0.31×	39.3	32.1	0.31×
Long-Context (128k)	32.8	25.1	1.00×	35.4	27.4	1.00×
Long-Context + retrieval	30.9	22.7	0.62×	33.1	24.8	0.62×
BranchSpan	28.3	19.4	0.21×	30.6	21.2	0.21×

The footprint column isolates serialized replay content, while Table 2 counts the entire prompt and reports median decision latency. BranchSpan therefore combines the smallest action-bearing payload among the high-performing methods with the lowest average step count and branch error on both suites. The latency comparison remains specific to the warm-cache WebArena protocol described in Section 4.

C Component Controls

Table 6: WebArena component controls. Panel A changes segmentation or training objectives. Panel B holds the encoder, learned segment inventory, prompt scaffold, task set, allocation rule, and retrieval depth fixed. Cosine-only changes only Eq. 9; removing the action coordinate drops index-only s_A; removing the branch code also removes its visible payload field. Cells report SR All / SR Long / independently valid top-1 replay.

Panel A: boundary and objective interventions

Variant	SR All	SR Long	Branch Err.
BranchSpan	25.4	21.1	19.4
Fixed-length boundaries	21.8	15.3	24.7
Without branch-code loss	22.6	16.4	28.6
Random boundaries	19.2	13.1	31.4

Panel B: shared-inventory replay interventions

Variant	SR All / SR Long / independent validity
BranchSpan	25.4±0.8 / 21.1±1.1 / 82.1% (197/240)
Without branch code	23.8±0.9 / 18.6±1.2 / 75.8% (182/240)
Without action coordinate	23.2±0.9 / 17.9±1.3 / 74.6% (179/240)
Cosine-only address	23.5±0.8 / 18.3±1.2 / 76.3% (183/240)
Fixed segments, typed headers	22.3±1.0 / 16.5±1.4 / 72.1% (173/240)

Panel A shows that decision-aligned segmentation and branch-sensitive representation both contribute to the complete system. Panel B supplies the tighter score comparison: full scoring versus cosine-only

changes no candidate, visible field, allocation, or encoder, yet improves SR All/Long by 1.9/2.8 points and valid top-1 replay by 5.8 points. The action-coordinate row is also index-only; the branch-code row jointly changes ranking evidence and one visible field.

D Outer-Agent Transfer

Table 7: Transfer to Qwen2.5-VL-72B ReAct with the same task inventories, deterministic resets, action interface, replay depths and allocations, replay encoder, serialization, and three seeds. The encoder is used without retraining. Cells report SR All / SR Long; intervals are paired 95% hierarchical-bootstrap intervals.

Method	WebArena All / Long	VisualWebArena All / Long
Long-Context	17.8±0.9 / 11.6±1.2	14.7±0.8 / 9.7±1.1
Long-Context + retrieval	20.2±1.0 / 14.7±1.3	16.8±0.9 / 12.3±1.2
BranchSpan	22.9±0.9 / 17.9±1.3	19.1±0.9 / 15.4±1.1
BranchSpan − hybrid, 95% CI All / Long	+2.7 [1.1,4.3] / +3.2 [0.8,5.8]	+2.3 [0.9,3.7] / +3.1 [0.8,5.4]

The frozen interface preserves the Long-Context < hybrid < BranchSpan ordering under Qwen2.5-VL-72B. This experiment isolates outcome transfer from one proprietary planner's behavior; the GPT-4o runs supply the separate latency analysis.

Table 8: Qwen2.5-VL-72B successes by seed. Each entry is WebArena / VisualWebArena. Per-seed denominators are 812/910 for All and 247/284 for Long.

Method	All successes by seed, WA / VWA	Long successes by seed, WA / VWA
Long-Context	132,145,157 / 121,134,146	24,28,34 / 22,28,33
Long-Context + retrieval	150,164,178 / 139,153,167	31,36,42 / 29,35,41
BranchSpan	173,186,199 / 160,174,187	39,44,50 / 38,44,49

The per-seed ledger exposes the denominators behind the means. Absolute success changes with the outer model, while the method ranking and larger long-horizon advantage persist.

E Predefined WebArena Operating Regimes

Table 9: Pre-outcome, overlapping WebArena slices from the same runs as Table 1. Short uses 5–10 reference actions; low branch has at most two page/modal/option branch events; JS-heavy has a majority of action targets introduced by client-side DOM mutation. Gains are BranchSpan minus 128k Long-Context with paired 95% hierarchical-bootstrap intervals.

Slice	Tasks	Gain (points)	95% CI
Short (5–10 actions)	132	+0.7	[-0.6,2.0]
Low branch pressure	241	+1.2	[-0.2,2.6]
JS-heavy pages	176	+0.8	[-0.7,2.3]
Long horizon ($\geq$ 25 actions)	247	+7.2	[4.6,9.8]

The estimates are small in the short, low-branch, and JS-heavy populations and largest in the long-horizon population. Because the populations overlap, each row is a separate operating-regime estimate rather than a level on a common scale.

F Memory-Address Design Space

Table 10: Architectural distinctions among adjacent memory systems. The address is the key used to select experience for the current decision; the final column identifies the object changed by each family.

System/family	Controller and replay unit	Memory address	Changed object
Long context	Fixed; trajectory tokens	Chronological position	Capacity only
Episodic retrieval	Fixed; raw states	Latent similarity	Semantic address
M^2 / SAM	Summary, cue, and external/raw pages	Summary or intent-driven recall	Experience source and payload (Yan et al., 2026; Hu et al., 2026)
AgentFold / AgentSwing	Controller-managed contexts or lookahead branches	Controller-selected utility	Context policy/controller (Ye et al., 2025; Feng et al., 2026)
Branch-and-Browse	Tree exploration, state restoration, page actions	URL/page and search branch	Planner and multi-branch experience (He et al., 2026)
ACGM	Fixed history, learned graph units	Task-adaptive graph relevance	Graph inventory and learned retriever (Forouzandeh et al., 2026)
BranchSpan	Fixed; typed realized-trajectory segments	Relevance, reachable actions, branch code	Continuation address

The closest empirical row is therefore Episodic retrieval, instantiated both alone and inside Long-Context. The shared-inventory cosine intervention narrows the comparison further by preserving BranchSpan's candidates and payload while deleting only continuation compatibility. Broader systems are scientifically adjacent but change the controller, experience source, or memory graph that this fixed-agent experiment holds constant.

G Decision-Boundary Trace

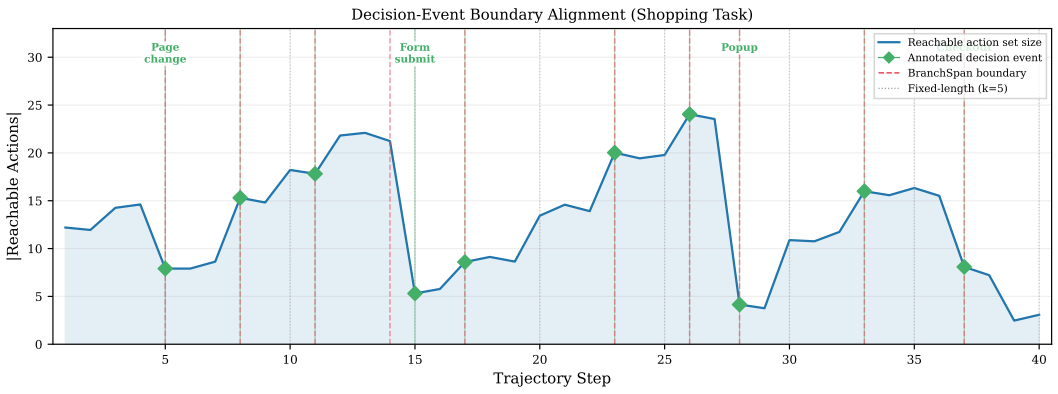


Figure 2: Decision-event boundary alignment on a 40-step shopping task. BranchSpan boundaries closely track page, form, modal, and checkout transitions, while fixed-length cuts split coherent action stretches.

The trace illustrates the supervision target in Section 3: boundaries follow interface events rather than periodic cuts. It complements the success intervention in Appendix C by showing the local

errors—one one-step offset for BranchSpan versus repeated within-decision cuts for the fixed-length construction.

H Claim-to-Evidence Matrix

Table 11: Evidence matrix connecting the continuation-address claim, operating assumptions, strongest controls, measured properties, and paper-internal evidence.

Claim	Stressor and trust assumptions	Property, strongest control, and evidence trace
Continuation indexing delivers applicable history	Similar displayed states after different action prefixes	Native-unit continuation audit on matched states; Table 3, Panel B
Continuation-indexed replay changes outcomes	Fixed outer agent, prompt, tools, tasks, and resets	Package-level success and paired gaps against chronological and semantic replay; Table 1
Addressing gain survives resource pressure	Separate mean-token and median-latency regimes	Success, tokens/step, and decision latency; Table 2
Continuation score changes WebArena point estimates	Identical encoder, inventory, fields, allocation, and depth	Success and selected-segment validity; Table 3, Panel A
Interface transfers across agents	Frozen replay encoder and serialization	Qwen2.5-VL-72B task success; Appendix D
Long-horizon replay gains 7.2 points	Predefined 247-task WebArena population	Paired interval $[4.6, 9.8]$; Appendix E, Fig. 1
Online cost is measured	Warm-cache median decision timing	40.3 ms encoder cost and $0.21\times$ payload; Section 4, Appendix B

The matrix distinguishes package-level outcomes from the address-only intervention and keeps the systems result tied to its measured latency statistic.