HIDDEN, NOT ERASED: ATTENTION-SIDE ACCESS CONTROL FOR COMPACTED AGENT MEMORY

Anonymous authors
Paper under double-blind review

ABSTRACT

Compacting an agent's trajectory is usually treated as deletion: once older interactions are summarized, their exact contents stop influencing the model, even though they still exist in an archived tool result. This conflates two separable decisions, storage and visibility. We make visibility the object of control. Triggered Sparse Attention keeps raw pre-compaction spans archived and ineligible by default. A trigger decides whether they may be attended to at all; the model's own sparse indexer then chooses which hidden tokens to read, independently for each layer and decoding position. In the model-native form, opened evidence is never written back into the prompt. On Qwen3-4B with a native sparse indexer, in 1,000 fresh synthetic agent traces whose queried fact survives only in the archive, triggered opening answers 342 while a closed archive and random opening at a higher rate answer none, and with sixteen archived events it answers 123 against 14 for random opening at exactly its opening rate. Reusing DeepSeek-V3.2's pretrained sparse indexer without retraining hidden-token identity, archive access recovers 21 of 26 hidden details where summary-only memory recovers none. In two pre-registered replications on 512 fresh paired WebShop tasks each, with byte-identical prompts across arms, selective recall beats random access at similar archive volume (157 against 134, $p = 0.015$; 163 against 132, $p < 0.001$) and raises mean reward over indiscriminate access at 2.7 times less archive volume.

1 INTRODUCTION

An agent can spend dozens of steps discovering a crucial fact, only to lose access to it when its harness compacts the trajectory. The fact has not vanished: it still exists in an archived tool result or observation, but the model can no longer attend to it. This failure is increasingly common as language agents operate through long sequences of searches, tool calls, code edits, and environment actions (Yao et al., 2023; 2022; Shridhar et al., 2021; Wang et al., 2022; Jimenez et al., 2024). An orchestration layer—the agent *harness*—must keep this growing trajectory within a finite working-memory budget. It therefore preserves the goal and recent interactions, summarizes older episodes, and archives their raw contents. This process makes long-running agents feasible, but changes what the model can know: after compaction, an exact product attribute, command output, or earlier observation may exist in storage while being absent from the prompt.

Existing memory strategies address only part of this problem. Hierarchical agent memories organize or summarize past experience (Packer et al., 2023; Hu et al., 2024), while retrieval-augmented systems paste selected records back into the prompt (Lewis et al., 2020; Jiang et al., 2023; Asai et al., 2024): the former can lose fine-grained evidence, and the latter makes every retrieved token globally visible throughout generation. Exposing the complete archive is also problematic, because old actions, failed plans, and repeated observations compete with the current state. The central question is therefore not whether more history is informative in principle, but how a model should regain *selected* pre-compaction information without destabilizing its current working context.

We formulate this missing capability as *post-compaction access control*. The key insight is that compaction need not destroy raw history: it can remove that history from the default attention candidate

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.

set while retaining an addressable representation. We implement this insight as Triggered Sparse Attention. At a structured decision boundary, a trigger decides whether archived memory should become eligible. A sparse-attention indexer then performs the fine-grained reading step, selecting hidden tokens separately for every layer and decoding position, so different layers may read different archived evidence rather than one block fixed for an entire action.

This separation between *access* and *reading* resolves two failure modes. First, a closed trigger removes archived tokens from the candidate set, so irrelevant archives cannot silently displace working-memory evidence. Second, when recall is needed, the pretrained sparse indexer—rather than a new text-generating controller—identifies useful tokens. Archived blocks remain storage and transfer units, not the model's final semantic decision units. This design is particularly natural for DeepSeek-V3.2, whose native DSA indexer already learns per-query token relevance (DeepSeek-AI, 2025). We also provide a Qwen conversion path for studying the mechanism on a smaller trainable backbone.

Our evaluation compares opening policies while reporting how much of the archive each exposes. On never-scored synthetic agent traces, a Qwen3-4B model with a native sparse indexer answers 342 of 1,000 archive-only queries when its training-free trigger decides where to open, while random opening at a higher rate answers none; with sixteen archived events the trigger answers 123 of 300 against 14 for random opening at exactly the trigger's opening rate. Two pre-registered WebShop studies with byte-identical starting prompts show the same ordering in an agent: selective access beats random access at similar expansion rates, and exceeds opening every record in pooled mean reward at 2.7 times less volume. A complementary DeepSeek-V3.2 study freezes its pretrained native indexer and changes only whether archived tokens may participate in sparse selection, including a fixed-total-token control. Together, the experiments support three contributions:

1. We define post-compaction recall as attention-side eligibility over archived raw history, with explicit closed, random, and all-open controls.
2. We introduce a model-native two-stage mechanism: a NULL/access decision controls archive eligibility, while the sparse indexer performs layer- and token-specific reading.
3. We show that query-conditioned opening beats random opening in a native sparse model, at a higher or exactly matched opening rate, and at similar expansion rates in two controlled agent studies, that it exceeds indiscriminate exposure in pooled agent reward at a fraction of the volume, and that a frozen pretrained native-DSA indexer reads archived evidence without relearning hidden-token identity.

We review related work and fix terminology (Sections 2–3), describe the mechanism and its training (Section 4), and give the protocol (Section 5). Section 6 reports, in order, where the archive opens in a native sparse model, selection over volume in a controlled agent study, what makes access useful, and transfer to a pretrained native indexer; Section 7 discusses implications.

2 Related Work

Agent memory and context compression. Long-horizon agents use summaries, episodic stores, and hierarchical working memory to remain within a context budget (Packer et al., 2023; Hu et al., 2024; Shinn et al., 2023; Park et al., 2023), and long-term memory benchmarks show that precise past facts remain difficult to recover (Maharana et al., 2024; Wu et al., 2024). Context-compression methods instead encode or recurrently summarize long sequences (Rae et al., 2019; Ge et al., 2023; He et al., 2024; Munkhdalai et al., 2024; Chari et al., 2025). Our setting begins after budget-driven agent compaction has already occurred.

Retrieval and memory augmentation. RAG retrieves documents into the input sequence (Lewis et al., 2020); FLARE and Self-RAG additionally learn when to retrieve (Jiang et al., 2023; Asai et al., 2024), while Memorizing Transformers and MemLong retrieve cached representations rather than text (Wu et al., 2022; Liu et al., 2024), which is the closest line to ours. We focus on a narrower interface: after budget-driven compaction, one eligibility decision determines whether archived tokens may compete in an existing sparse indexer at all.

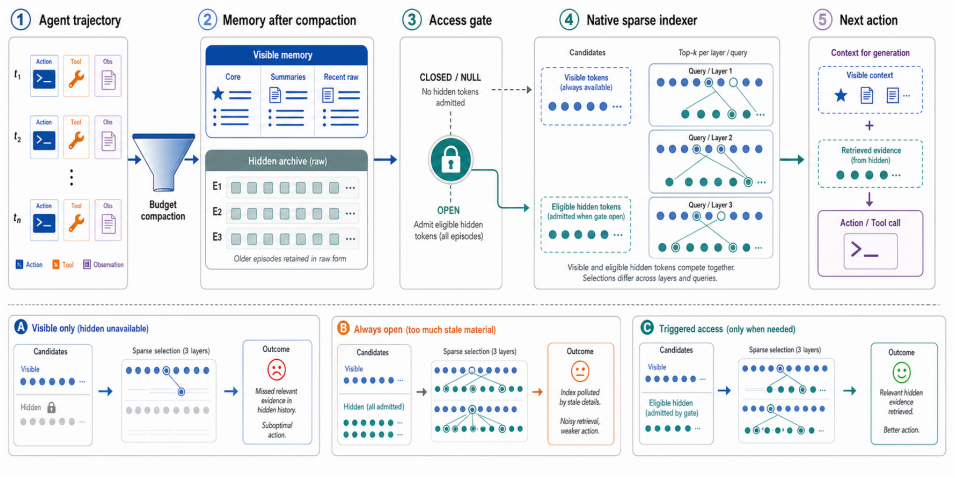


Figure 1: Triggered post-compaction recall. Budget-driven compaction leaves core state, summaries, and recent raw events visible while preserving older raw episodes in a hidden archive. The gate controls only archive *eligibility*: CLOSED preserves the ordinary visible path, whereas OPEN lets hidden tokens enter the native sparse indexer's selection (shown here for the fixed-total contract, in which visible and hidden tokens compete). The winning hidden tokens may differ across layers and queries. The lower strip contrasts missing evidence, indiscriminate exposure, and targeted recall.

Sparse attention. Fixed patterns such as Longformer, learned routing, and query-key compression reduce attention cost by selecting source tokens (Beltagy et al., 2020; Kitaev et al., 2020; Roy et al., 2021; Wang et al., 2025). DSA learns lightweight index queries and keys for dynamic top-k selection (DeepSeek-AI, 2025); recent work improves hierarchical selection and cross-layer index reuse (Xu et al., 2026; Bai et al., 2026; Ni et al., 2026). We instead use sparse selection to control which pre-compaction tokens can re-enter a model's information path; Appendix Table 3 compares the resulting interface with summary-only compaction, prompt retrieval, and always-open K/V memory.

3 Problem Setting

At agent step t, a budgeted memory manager partitions the trajectory into

$$M_t = (C_t, S_t, R_t, H_t), \tag{1}$$

where C_t is stable core state, S_t contains visible episode summaries, R_t is a recent raw working set, and $H_t = \{H_{t,b}\}_{b=1}^{B_t}$ stores raw archived episodes. The visible input is $V_t = C_t \cup S_t \cup R_t$. A new episode is archived only when the raw working set exceeds a benchmark-specific token budget; hence compaction is an observed event, not a prompt annotation.

We compare five memory semantics. Visible-Only uses V_t and never exposes H_t. Full-History retains the clean raw trajectory without compaction or summary–raw duplication and serves as an information reference whenever it fits. Always-Open exposes all archived episodes and is a pollution stress test, not an upper bound. Triggered-SA makes hidden access query-conditioned. Random opens the archive at a fixed probability, without conditioning on the current query.

The primary evaluation target is Triggered-SA versus Visible-Only and undirected-exposure controls after real compaction. We report task outcome together with compaction and hidden-exposure counts, which ties every post-compaction result to observed compaction and distinguishes targeted recall from exposure volume.

4 TRIGGERED SPARSE ATTENTION

ARCHIVE ELIGIBILITY AND NATIVE READING

Let $q'_{\ell,i}$ be the sparse index query for layer ℓ and decoding position i, and let k'_j be the index key retained for source token j. The native relevance score is

$$s_{\ell,i,j} = {q'_{\ell,i}}^{\top} k'_j. \tag{2}$$

A NULL/access variable $g_i \in \{0, 1\}$ controls the candidate universe:

$$\mathcal{C}_{\ell,i} = V_t \cup \begin{cases} \varnothing, & g_i = 0, \\ H_t, & g_i = 1. \end{cases} \qquad \mathcal{I}_{\ell,i} = \mathrm{TopK}_{j \in \mathcal{C}_{\ell,i}} s_{\ell,i,j}. \tag{3}$$

The gate may be one bit per decision, shared across layers, or one bit per (ℓ, i), written $g_{\ell,i}$; the native probe in Section 6 uses the per-position form. Hidden identity is induced by native top-k membership: a token from episode $H_{t,b}$ is visible at (ℓ, i) exactly when it wins this sparse competition. Blocks aggregate token memberships only for storage transfer and logging; they do not force all layers to read the same episode.

This formulation gives the model a broader reading surface than an action-fixed block selector: even with a bounded hidden-token budget per layer, the union of selections across depth can cover several archived episodes, and the sparse indexer's job is not duplicated by a separate semantic retriever.

VISIBLE PRESERVATION AND ATTENTION BUDGETS

We use two attention contracts for complementary purposes. The *fixed-total* contract takes one top-k over visible and eligible hidden tokens, keeping the total selected-token count identical to VISIBLE-ONLY. It controls the selected-token count; archive storage and the index computation over eligible tokens still differ. The *visible-preserving* contract retains the exact visible selection $\mathcal{I}^V_{\ell,i}$ and adds a bounded hidden sidecar $\mathcal{I}^H_{\ell,i}$:

$$o_{\ell,i} = o^V_{\ell,i} + \alpha_{\ell,i} o^H_{\ell,i}. \tag{4}$$

The hidden branch is normalized separately. When $g_i = 0$ or $\alpha_{\ell,i} = 0$, its contribution is exactly zero and the visible computation is unchanged. Fixed-total selection answers the capacity question; visible-preserving selection is the safer deployment form because opening the archive cannot evict current evidence.

LEARNING ACCESS AND ADOPTION

The access head should answer "does hidden evidence help now?", not duplicate token identity ranking. It consumes pooled native-index scores, hidden and visible margins, and the current query state, then predicts a NULL/open logit. We train it from same-history counterfactual pairs. For query i, the positive and negative executions share the prefix, current state, candidate actions, and decoding configuration; only archived access changes. Pairwise utility supervision uses

$$\mathcal{L}_{\mathrm{pair}} = -\log \sigma(u_\phi(q_i, H_t) - u_\phi(q_i, \varnothing)), \tag{5}$$

where the first argument is the preferred branch: for explicit NULL examples and harmful stale-history negatives the two arguments swap, so the loss pushes $u_\phi(q_i, H_t) - u_\phi(q_i, \varnothing)$ below zero. The pairwise loss constrains only this difference, so the open/close threshold is set separately by calibration on held-out pairs. A residual-adoption coefficient $\alpha_{\ell,i}$ can be trained with the same pairs when a model selects useful evidence but underuses its values.

DENSE-BACKBONE CONVERSION

Qwen serves as a smaller training platform and requires an additional setup stage because its base attention does not include a pretrained DSA indexer. First, we distill sparse token scores from dense-attention mass:

$$\mathcal{L}_{\mathrm{idx}} = \sum_{\ell,i} \mathrm{KL}\left(a_{\ell,i} \,\|\, \mathrm{softmax}(s_{\ell,i})\right). \tag{6}$$

Second, we train hidden access with block and pairwise supervision. Third, we adapt the modified sparse path with continued next-token training and instruction/tool SFT, updating LoRA or full language projections while keeping the discrete top-k indexer objective separate, because selecting the right tokens does not by itself make the modified path generate well.

The End-to-End Memory Cycle

The mechanism lives at the seam between the harness and the model (Figure 1). Once the raw working set exceeds its budget, the harness leaves a short episode representation in S_t and moves the unabridged token states, K/V tensors, and index keys to H_t. At a later decision, NULL preserves the visible computation, while OPEN admits archived tokens to the sparse candidate set; every layer and position makes its own top-k choice. The gate learns the utility boundary; the indexer keeps the harder job of finding tokens. The interface is one eligibility bit, so any compacting harness with access to archived model states and index keys can carry it and supervise it from its own trajectories.

Supervision from Agent Trajectories

Training examples arise naturally from the memory lifecycle. We replay a decision with an identical visible prefix, action space, archive, and decoding configuration, changing only archive eligibility. A positive pair is created when access increases the likelihood of a reference action, recovers a hidden target, avoids a redundant action, or improves downstream return; NULL examples include decisions already solvable from V_t, and hard negatives pair a query with equal-volume but stale or task-irrelevant archived K/V.

The pair construction matters as much as the loss. Independent positive and negative examples let a classifier exploit superficial differences in trajectory length, episode count, or task template; same-history counterfactuals remove those shortcuts, because both branches share the same story up to the access boundary. We partition by task identity before constructing pairs, so two views of one trajectory cannot straddle train and evaluation, and probability-matched random controls test whether a gain reflects conditional utility rather than a general preference for opening more memory.

5 Experimental Setup

Models. The model-native probes use Qwen3-4B converted to DSA (index distillation and sparse-path adaptation; Appendix B); the agent-side WebShop study uses Qwen3-30B-A3B. The cross-model mechanism study uses DeepSeek-V3.2, whose pretrained native DSA indexer remains frozen; training is confined to lightweight NULL/adoption calibration on its native features.

Benchmarks. The model-native probe is a synthetic agent execution log of 20 tool events in which one archived event records a queried code that must be copied exactly; the event is attention-ineligible unless the trigger opens it. WebShop Text (Yao et al., 2022) is the primary paired environment: earlier queries, product pages, and option values may become useful evidence or stale actions after compaction. Paired arms share task ids, memory budget, summary policy, recent window, model checkpoint, and decoding rule.

Compaction protocol. Working-memory budgets are selected per benchmark because one token limit does not create comparable pressure across environments. An episode is archived only after the raw window crosses its declared budget. We retain the task instruction, structured core state, episode summaries, and most recent raw interactions in every compacted condition. We record final compaction count, hidden expansions, selected hidden records, visible tokens, tool calls, repeated actions, and format validity. The DeepSeek detail suite instead uses three scheduled folds per query, a controlled setting rather than budget-triggered compaction.

Pre-registration and separation. Every endpoint comparison (both native probes and both WebShop waves) was registered, with its primary contrast and population, before any of its outputs existed, and was monitored only through checks that print no outcomes. Thresholds were set on calibration tasks and never retuned on scored tasks, and no task or slice was excluded. The DeepSeek-

V3.2 detail suite evaluates frozen native membership, with NULL/adoption calibration on identity-disjoint pairs.

Metrics and statistics. WebShop outcomes are *nonzero-reward outcomes* ($R > 0$, some attributes matched), not full matches ($R = 1$), which are rare and similar in every arm (5–10 of 512 per arm). Native-probe accuracy is exact match of the copied code. Paired binary outcomes use two-sided exact McNemar tests. Mechanism studies report hidden-token selection, content controls, and NULL/open discrimination; efficiency is reported separately as archive bytes and warm admission latency.

Implementation details. The Qwen and DeepSeek paths use bf16 archived states, and any forced recent tokens remain inside rather than on top of the declared sparse-selection budget. Qwen streams selected K/V per head so that large candidate budgets do not materialize all selected heads at once, and DeepSeek generation uses the ordinary chat opening rather than a visible reasoning trace. For fixed-total controls, visible-only and triggered conditions select the same total number of tokens; for visible-preserving runs, hidden admission is reported as an explicit side budget, and the two contracts are never mixed.

6 RESULTS

The evaluation asks three questions in order: whether the place where the archive opens matters in a native sparse model, whether selection beats undirected access in a real agent under controlled prompts, and what makes access useful and whether it transfers to a pretrained native indexer.

WHERE THE ARCHIVE OPENS, IN A NATIVE SPARSE MODEL

The cleanest test of the model-native mechanism is Qwen3-4B with native DSA on synthetic agent traces whose queried fact lies only in archived, attention-ineligible events (Table 1, Figure 2a). The trigger is the training-free score-margin rule: a (layer, query) position opens when its best archived index score comes within 3.65 of its best visible one, a threshold set on 2,000 calibration traces and then frozen. Every comparison was registered before any trace was scored, and each uses never-scored traces.

With one archived event, a closed archive answers none of 1,000 traces and TRIGGERED-SA answers 342. Random opening at a *higher* rate than the trigger (0.54 against 0.33 of positions) also answers none: 342 wins and no losses for TRIGGERED-SA (exact McNemar $p = 2{\times}10^{-103}$). Opening a larger share of positions at random recovers nothing; which positions open decides whether the fact is recovered. With sixteen archived events (1,702 hidden tokens), the result is unchanged: random opening answers none of 300 and TRIGGERED-SA 123 (123 wins, 0 losses, $p = 2{\times}10^{-37}$). Matching random opening exactly to the trigger's realized rate on these traces (0.90 of positions, 917 against 918 hidden tokens selected per trace on average) raises it only to 14 (113 wins, 4 losses for TRIGGERED-SA, $p = 9{\times}10^{-29}$). Inside the opened archive the native indexer is selective: at open positions it retains 95% of the answer tokens under the trigger and 74% under random opening, against 60% for a blind choice of the same number of hidden tokens. Making the full trace visible, which serves as the information reference, answers 658 and 197; a training-free trigger thus recovers 52% and 62% of the reference's answers while leaving 67% and 10% of (layer, query) positions on the visible-only path.

SELECTION OVER VOLUME IN A CONTROLLED AGENT STUDY

The agent-side study asks whether selection over volume holds in a real environment, with a different archive representation and backbone (Qwen3-30B-A3B). Old tool observations are folded into hidden records while an exact action ledger stays visible; an opened record's raw observations are restored to the agent's context, so this study isolates the learned access decision. The trigger threshold 0.005 and the random opening probability 0.332 were fixed on calibration tasks and never retuned. Every arm receives a byte-identical prompt until the first archive opening, and all four arms take identical actions up to that point on every task, and each wave was registered, with its primary contrast, before any job ran.

Table 1: Qwen3-4B with native DSA on never-scored synthetic agent traces (exact-copy answers). Open rate is the fraction of (layer, query) positions whose archive is eligible. W/L counts traces that TRIGGERED-SA answers and the row does not, and the reverse; p is exact McNemar. Full access makes the whole trace visible and is the information reference, not a memory policy.

	One archived event ($n = 1{,}000$)			Sixteen archived events ($n = 300$)		
Policy	Correct	Open rate	W/L, p	Correct	Open rate	W/L, p
VISIBLE-ONLY	0	0.00	342/0, 2×10^{-103}	0	0.00	123/0, 2×10^{-37}
RANDOM	0	0.54	342/0, 2×10^{-103}	0	0.54	123/0, 2×10^{-37}
RANDOM, rate-matched	–	–	–	14	0.90	113/4, 9×10^{-29}
TRIGGERED-SA	**342**	0.33	–	**123**	0.90	–
Full access (reference)	658	1.00	–	197	1.00	–

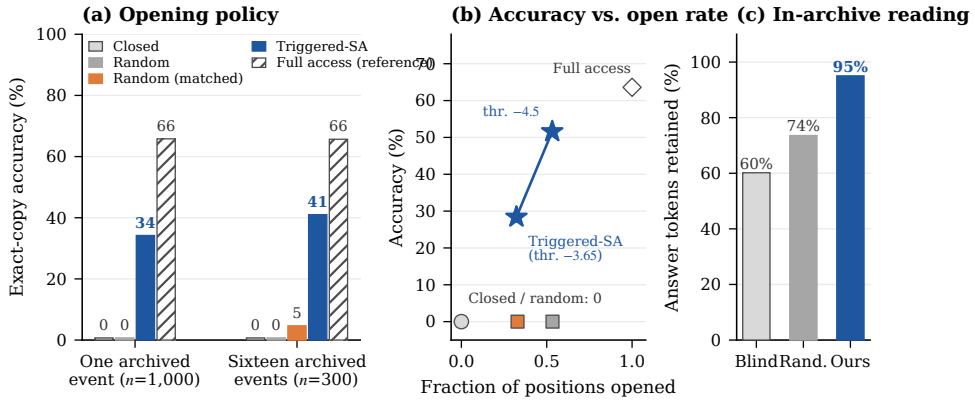


Figure 2: Model-native results (Qwen3-4B, native DSA, never-scored traces). (a) Random opening recovers almost no archive-only facts, even at exactly the trigger's opening rate, while TRIGGERED-SA recovers 34% and 41%. (b) The trigger threshold traces an accuracy–openness frontier that none of the random settings reaches (250 traces). (c) At open positions of the sixteen-event probe, the native indexer retains 95% of the answer tokens under TRIGGERED-SA, against 74% under random opening and 60% for a blind choice.

Figure 3 and Appendix Table 7 report two independent waves of 512 never-scored tasks. Against random opening at similar volume (0.63 against 0.70 expansions per step), TRIGGERED-SA obtains a nonzero reward on 157 against 134 tasks (53 wins, 30 losses, $p = 0.015$) and, in the confirmatory wave, on 163 against 132 (56 wins, 25 losses, $p = 7.5\times10^{-4}$); pooled over 1,024 tasks, mean reward rises from 0.144 to 0.171 (paired $\Delta R = +0.027$, 95% CI $[+0.014, +0.041]$). Against opening every record, at 2.7 times fewer expansions (0.70 against 1.86 per step), TRIGGERED-SA raises mean reward from 0.154 to 0.171 ($\Delta R = +0.017$, CI $[+0.005, +0.028]$). Undirected access is actively harmful here: random opening falls below the closed archive (266 against 300, $p = 0.0045$), whereas TRIGGERED-SA retains the closed archive's reliability and adds to it (320 against 300).

CONTENT, PLACEMENT, AND GATE TRAINING

Matched content and placement controls isolate why access changes the model. With placement and K/V amount fixed, correct archived K/V beats both NULL and equal-length incorrect K/V in all eight reciprocal dyads of a controlled Qwen3-4B probe (two-sided sign test $p = 0.0078$), so the effect is carried by the archived content, not by extra attention mass. At a matched WebShop decision, attention-side admission produces a 1.96-nat larger complete-action likelihood margin than prompt insertion, so access changes action preference even when the recovered evidence is held constant.

Pairwise recall/no-recall training improves the access decision (Appendix Table 10). On identity-disjoint ScienceWorld validation pairs, at matched data, seed, parameter count, steps, and two for-

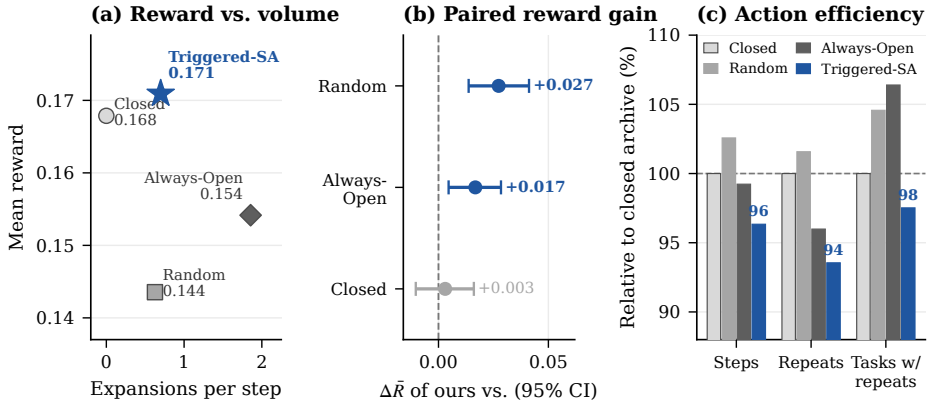


Figure 3: Controlled WebShop study (1,024 paired tasks). (a) TRIGGERED-SA reaches the highest mean reward at 0.70 expansions per step. (b) Its paired reward gain is significant against random and indiscriminate access (95% task-bootstrap CIs). (c) It also takes the fewest steps and repeated actions.

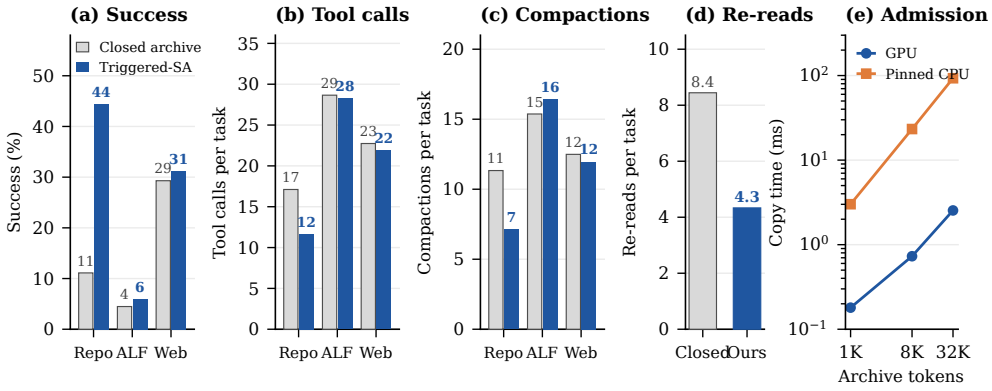


Figure 4: Across surfaces (repository multihop, ALFWorld `valid_unseen`, WebShop; Qwen3-30B-A3B, paired tasks, same compaction policy per surface), with the same frozen trigger, TRIGGERED-SA raises success (a) with fewer tool calls (b), fewer compactions (c) and fewer re-reads of already visited files (d); copying archived K/V and index keys into attention scales linearly with archive size (e).

ward passes per pair, an access head trained pairwise reaches 87.20% AUROC against 75.84% for independently scored pointwise examples, an 11.4-point gain that closes 47% of the remaining gap to perfect ranking, and $85.28\% \pm 1.67\%$ across three seeds; disabling this stage leaves 68.00%. AUROC is threshold-independent ranking quality (50% is chance).

NATIVE DSA TRANSFER

DeepSeek-V3.2 asks a transfer question: when token identity is already learned at scale, does archive access still need new token-level training? We freeze the pretrained native DSA indexer and let its union top-k membership determine hidden-token identity. In a 26-query post-fold detail suite, VISIBLE-ONLY recovers 0 details, native archive access recovers 21, and the all-visible reference recovers 24. The access-versus-visible paired difference is significant under exact McNemar testing ($p = 9.54 \times 10^{-7}$). Each query undergoes three scheduled compactions and the archive contains three raw episodes, so the comparison specifically tests post-compaction evidence.

We next isolate task utility under a fixed attention budget. From a common three-step prefix, VISIBLE-ONLY selects 2,048 tokens and obtains zero reward in 19 steps. Native union top-k keeps the same total 2,048-token budget, selects archived tokens in all 61 layers of the final decision, and reaches reward 1.0 in 12 steps. The model's original full-context top-k path reproduces the same trajectory. Across the matched transfer suite, native membership gives three reward wins, one tie, and no loss, with mean reward 0.443 versus 0 and 18.6% fewer steps (14.25 versus 17.50). A pretrained native indexer therefore reads archived evidence without relearning token identity.

Finally, a lightweight NULL/adoption head calibrates frozen native features. The 897-parameter head over seven DSA layers reaches AUROC $82.52\% \pm 0.54\%$, balanced accuracy $80.73\% \pm 0.90\%$, false-open rate 12.5%, and **93.8% pair accuracy** (30/32) across three seeds. This supports the design choice to reuse the pretrained indexer for identity and spend trainable capacity only on NULL/access and value adoption. Appendix Table 6 lists the model, gate, opening unit, and reading path of every experiment, and Appendix Tables 12 and 13 give a matched trajectory and archive storage and admission cost.

7 ANALYSIS AND DEPLOYMENT IMPLICATIONS

Retrieval solves location; access control solves placement. A text-insertion retrieval pipeline that finds the right archived record serializes it back into the prompt, where it becomes visible to every layer and every later token; cached-representation retrieval need not. Holding the recovered evidence fixed, admitting it through attention gives a 1.96-nat larger action-likelihood margin than inserting it as text. The distinction is not "retrieval versus no retrieval" but whether recovered history becomes prompt text or a conditional attention candidate.

The best memory is often a closed one. Opening the archive has asymmetric risk: a false opening can revive an obsolete plan across many subsequent tokens. In both controlled agent waves, random access falls below a permanently closed archive (266 against 300 of 1,024, $p = 0.0045$) while selective access stays above it. NULL is a first-class action protecting an already sufficient visible state.

Deployment follows suit: use benchmark-specific memory budgets and apply ordinary access and deletion controls to archived tool results.

8 CONCLUSION

Compaction removes history from view; it need not remove it from reach. Triggered Sparse Attention makes raw archived interactions ineligible by default, lets a heuristic or learned trigger decide when they become eligible, and delegates fine-grained reading to the model's sparse indexer. With a native indexer, where the archive opens decides whether archive-only facts are recovered (342 of 1,000 fresh traces against none for random opening, and 123 of 300 with sixteen archived events against 14 at an exactly matched opening rate); selection beats random access in both of two pre-registered agent waves and exceeds indiscriminate access in pooled mean reward; and a pretrained DeepSeek-V3.2 indexer reads archived evidence without retraining hidden-token identity. Attention-side access control is thus a concrete path toward language models that participate in the memory lifecycle of the agents deploying them.

LIMITATIONS

The native probes use synthetic traces and a training-free trigger, and the agent study evaluates the learned access decision; combining the learned gate with native reading in a live agent is the natural next step. Gates and budgets should be calibrated per backbone and benchmark, and costs are reported for one warm Qwen3-4B bf16 configuration on H200. The archive assumes access to model-side K/V states and index keys, which is natural for local inference but storage-intensive. A false negative withholds needed evidence and a false positive revives stale output, and the content controls establish causal influence only under controlled conditions.

Reproducibility statement

The runtime contract, training recipe, evaluation protocol, expanded result tables, hardware accounting, and the paper-to-code map are given in Appendix A, Appendix B, Appendix C, Appendix D, Appendix E, and Appendix F. Every endpoint record retains task and artifact identity, the working-memory budget, compaction and expansion counts, hidden selections, tool and format statistics, and the final memory state, and policies are paired by task identity. Every reported statistic is recomputed from per-task records by fail-closed audit scripts; binary outcomes use exact McNemar testing and reward differences use paired task-bootstrap intervals. Aggregate result records and SHA-256 hashes accompany the anonymous artifact; model weights and benchmark environments are resolved through ordinary local path arguments, and third-party assets remain under their original licenses.

Ethics statement

The experiments use public agent benchmarks and do not involve newly collected personal data or human participants. A deployed memory archive may nevertheless contain sensitive user text, credentials, copyrighted material, or malicious instructions returned by tools. Compaction must not be treated as deletion: hidden records should inherit the same authorization, retention, provenance, deletion, and content-safety policies as visible context, and a learned gate is not a privacy or security boundary. Training and evaluating sparse-access models consumes accelerator resources, and retaining K/V increases storage use; we report model size, archive accounting, and admission measurements to make this cost visible. Generative AI tools were used across the research and writing workflow; this use is disclosed in the AI use statement below, and the authors remain responsible for every claim, citation, and artifact.

AI use statement

In this work, we used generative AI tools for the following tasks: implementing and modifying the attention, gate, harness, and evaluation code used in the experiments; orchestrating, monitoring, and auditing experiment runs; searching, summarizing, and formatting the related literature and references; proposing and refining hypotheses and experimental designs; interpreting intermediate results; creating and editing figures; and drafting and revising parts of the manuscript. All AI-assisted work was reviewed by the authors. Every numerical claim is bound to a result artifact and was recomputed from its per-task records, and AI-assisted code was executed and tested before its results were used. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

Yushi Bai, Qian Dong, Ting Jiang, Xin Lv, Zhengxiao Du, Aohan Zeng, Jie Tang, and Juanzi Li. Indexcache: Accelerating sparse attention via cross-layer index reuse. *arXiv preprint arXiv:2603.12201*, 2026. URL https://arxiv.org/abs/2603.12201.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020. URL https://arxiv.org/abs/2004.05150.

Vivek Chari, Guanghui Qin, and Benjamin Van Durme. Kv-distill: Nearly lossless learnable context compression for llms. *arXiv preprint arXiv:2503.10337*, 2025. URL https://arxiv.org/abs/2503.10337.

DeepSeek-AI. DeepSeek-V3.2-Exp: Boosting long-context efficiency with DeepSeek Sparse Attention. Technical report, DeepSeek-AI, 2025. URL https://github.com/deepseek-ai/DeepSeek-V3.2-Exp/blob/main/DeepSeek_V3_2.pdf.

Tao Ge, Jing Hu, Lei Wang, Xun Wang, Si-Qing Chen, and Furu Wei. In-context autoencoder for context compression in a large language model. *arXiv preprint arXiv:2307.06945*, 2023. URL https://arxiv.org/abs/2307.06945.

Zifan He, Yingqi Cao, Zongyue Qin, Neha Prakriya, Yizhou Sun, and Jason Cong. Hmt: Hierarchical memory transformer for efficient long context language processing. *arXiv preprint arXiv:2405.06067*, 2024. URL https://arxiv.org/abs/2405.06067.

Mengkang Hu, Tianxing Chen, Qiguang Chen, Yao Mu, Wenqi Shao, and Ping Luo. Hiagent: Hierarchical working memory management for solving long-horizon agent tasks with large language model. *arXiv preprint arXiv:2408.09559*, 2024. URL https://arxiv.org/abs/2408.09559.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. *arXiv preprint arXiv:2305.06983*, 2023. URL https://arxiv.org/abs/2305.06983.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/2001.04451.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, 2020. URL https://arxiv.org/abs/2005.11401.

Weijie Liu, Zecheng Tang, Juntao Li, Kehai Chen, and Min Zhang. Memlong: Memory-augmented retrieval for long text modeling. *arXiv preprint arXiv:2408.16967*, 2024. URL https://arxiv.org/abs/2408.16967.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of llm agents. *arXiv preprint arXiv:2402.17753*, 2024. URL https://arxiv.org/abs/2402.17753.

Tsendsuren Munkhdalai, Manaal Faruqui, and Siddharth Gopal. Leave no context behind: Efficient infinite context transformers with infini-attention. *arXiv preprint arXiv:2404.07143*, 2024. URL https://arxiv.org/abs/2404.07143.

Wentao Ni, Kangqi Zhang, Zhongming Yu, Oren Nelson, Mingu Lee, Hong Cai, Fatih Porikli, Jongryool Kim, Zhijian Liu, and Jishen Zhao. Double-p: Hierarchical top-p sparse attention for long-context llms. *arXiv preprint arXiv:2602.05191*, 2026. URL https://arxiv.org/abs/2602.05191.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*, 2023. URL https://arxiv.org/abs/2310.08560.

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 *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023. URL https://arxiv.org/abs/2304.03442.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling. *arXiv preprint arXiv:1911.05507*, 2019. URL https://arxiv.org/abs/1911.05507.

Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. Efficient content-based sparse attention with routing transformers. *Transactions of the Association for Computational Linguistics*, 9:53–68, 2021. doi: 10.1162/tacl_a_00353. URL https://aclanthology.org/2021.tacl-1.4/.

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. URL https://arxiv.org/abs/2303.11366.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2010.03768.

Haoyu Wang, Tong Teng, Tianyu Guo, An Xiao, Duyu Tang, Hanting Chen, and Yunhe Wang. Unshackling context length: An efficient selective attention approach through query-key compression. *arXiv preprint arXiv:2502.14477*, 2025. URL https://arxiv.org/abs/2502.14477.

Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022. URL https://arxiv.org/abs/2203.07540.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. Longmemeval: Benchmarking chat assistants on long-term interactive memory. *arXiv preprint arXiv:2410.10813*, 2024. URL https://arxiv.org/abs/2410.10813.

Yuhuai Wu, Markus N. Rabe, DeLesley Hutchins, and Christian Szegedy. Memorizing transformers. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TrjbxzRcnf-.

Yufei Xu, Fanxu Meng, Fan Jiang, Yuxuan Wang, Ruijie Zhou, Jiexi Wu, Zhixin Pan, Zhaohui Wang, Xiaojuan Tang, Wenjie Pei, Tongxuan Liu, Di Yin, Xing Sun, and Muhan Zhang. Hisa: Efficient hierarchical indexing for fine-grained sparse attention. *arXiv preprint arXiv:2603.28458*, 2026. URL https://arxiv.org/abs/2603.28458.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. In *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=R9KnuFlvnU.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

A Complete Runtime Contract

Memory State and Compaction Invariants

At decision step t, the memory manager maintains

$$M_t = (C_t, S_t, R_t, H_t), \tag{7}$$

where C_t is stable core state, S_t contains compact visible episode representations, R_t is the recent raw working set, and $H_t = \{H_{t,b}\}_{b=1}^{B_t}$ is the raw hidden archive. The visible sequence is $V_t = C_t \cup S_t \cup R_t$. The following invariants define a post-compaction evaluation:

1. **Budget trigger.** An episode moves from R_t to H_t only after the declared benchmark-specific working-memory budget is exceeded.
2. **Raw fidelity.** The hidden record retains the unabridged episode tokens and their model-side K/V and index-key representations; the visible representation in S_t is not substituted for the raw archive.
3. **Narrative continuity.** Core state, episode representations, and the recent raw window remain visible in every compacted policy.

Table 2: Algorithm 1: one complete compaction-and-recall cycle. The gate controls eligibility; the sparse indexer controls token identity.

1	Append the latest action, observation, and tool result to the recent raw working set R_t.
2	If R_t is within its benchmark-specific budget, retain the ordinary visible path and continue generation.
3	Otherwise, close the oldest complete episode. Write its compact representation to S_t and its raw tokens, K/V tensors, index keys, source interval, and content hash to H_t.
4	At the next structured decision boundary, compute the ordinary visible query state and the access logit $u_\phi(q_i, H_t)$.
5	If NULL wins, set $g_i = 0$ and execute exactly the visible-only attention path.
6	If OPEN wins, stage the eligible archive pages and set $g_i = 1$; do not insert recalled text into the prompt.
7	At each layer and decoding position, let the native indexer select top-k tokens from $\mathcal{C}_{\ell,i}$. Apply the fixed-total or visible-preserving contract declared for the experiment.
8	Generate and execute the next action. Record gate state, eligible archive ids, selected hidden tokens by layer, action/tool validity, reward, and final memory state.

4. **Stable identity.** Each archived episode retains a stable id, source-step interval, and content hash for paired analysis and audit.
5. **Stable positions.** Archived K/V and index keys keep the rotary positions they were computed with; in the native probes an archived span stays in place in the sequence and only its eligibility changes, so opening it requires no re-encoding or position remapping.
6. **Observed pressure.** Post-compaction metrics are tied to recorded compaction counts, expansions, selected hidden tokens, and final memory state.

Eligibility and Native Sparse Reading

For query token i in layer ℓ, let $q'_{\ell,i}$ be the native sparse index query and k'_j the retained index key for source token j. Native relevance is

$$s_{\ell,i,j} = {q'_{\ell,i}}^\top k'_j. \tag{8}$$

The access decision $g_i \in \{0, 1\}$ changes only the candidate universe:

$$\mathcal{C}_{\ell,i} = V_t \cup \begin{cases} \varnothing, & g_i = 0, \\ H_t, & g_i = 1, \end{cases} \qquad \mathcal{I}_{\ell,i} = \mathrm{TopK}_{j \in \mathcal{C}_{\ell,i}} s_{\ell,i,j}. \tag{9}$$

The gate therefore answers whether hidden evidence should be searchable. Token identity remains a native top-k decision at each layer and query position. Archived episodes are storage and transfer units; they are not fixed semantic choices shared by the full forward pass.

Two Attention Contracts

The *fixed-total* contract performs one top-k operation over visible and eligible hidden tokens. Its selected-token width is identical to VISIBLE-ONLY, so paired gains isolate information placement rather than added attention capacity.

The *visible-preserving* contract retains the visible selection $\mathcal{I}^V_{\ell,i}$ and adds a bounded hidden sidecar $\mathcal{I}^H_{\ell,i}$:

$$o_{\ell,i} = o^V_{\ell,i} + \alpha_{\ell,i} o^H_{\ell,i}. \tag{10}$$

The branches are normalized separately. When $g_i = 0$ or $\alpha_{\ell,i} = 0$, the hidden residual is exactly zero. This contract protects working-memory evidence from eviction when the archive opens.

Closed-Path Equivalence

Closed-path equivalence is an implementation contract, not an approximate behavioral expectation. The hidden branch receives an empty eligibility mask, its separately normalized output is zero, and

Table 3: The design boundary relative to common memory interfaces. "Prompt insertion" means that recalled records become ordinary input tokens visible throughout the forward pass. Our gate controls archive eligibility, while the native sparse indexer retains token-level reading.

Interface	Raw past retained	Query-triggered	Prompt insertion	Hidden eligibility	Final reading unit
Summary-only compaction	No	–	No	None	Summary token
Prompt retrieval / RAG	Yes	Yes	Yes	Retrieved records	Visible prompt token
Always-open K/V memory	Yes	No	No	Entire archive	Sparse token
TRIGGERED-SA	**Yes**	**Yes**	**No**	**NULL / OPEN**	**Per-layer, per-query token**

Table 4: Training stages and their separate responsibilities.

Stage	Updated component	Objective	Observable success criterion
1: index distillation	Sparse index query/key projections	KL distillation from dense-attention mass	High target-token hit under a fixed selection width
2: access learning	NULL/open head and optional adoption coefficient	Same-history pairwise utility, explicit NULL, and stale-history negatives	High AUROC and low false-open rate under identity-disjoint calibration
3a: sparse adaptation	LoRA or language projections	Ordinary next-token continued training on the modified sparse path	Stable generation and reduced target-token misses
3b: tool adoption	Language/tool projections	Instruction and tool SFT	Valid action format and improved paired endpoint completion

the visible selection and residual are unchanged. With fixed-width selection tensors, closed positions of a partially opened layer are padded to a common width; the padded slots carry a validity mask and are excluded before the softmax, so a closed position computes exactly the visible-only attention. Tests in the anonymous artifact cover empty archives, closed gates with nonempty archives, forced-open traces, selection-budget accounting, and raw-episode fidelity.

B TRAINING THE GATE AND SPARSE PATH

SAME-HISTORY COUNTERFACTUAL SUPERVISION

The access target is conditional utility, not semantic similarity. For a decision query q_i, paired executions share the visible prefix, archive, candidate actions, model checkpoint, decoding configuration, and environment snapshot. Only archive eligibility changes. Pairwise supervision is

$$\mathcal{L}_{\text{pair}} = -\log \sigma \left(u_\phi(q_i, H_t) - u_\phi(q_i, \varnothing) \right). \tag{11}$$

A pair is positive when access recovers a hidden target, increases the likelihood of a reference action, avoids a redundant action, or improves downstream return. NULL examples are decisions already solvable from V_t. Hard negatives use equal-volume stale or task-irrelevant archived K/V. Task-identity partitioning occurs before pair construction so two branches of one trajectory never cross training and evaluation pools.

DENSE-BACKBONE CONVERSION

Qwen3-4B supplies a compact platform for studying the complete conversion. Stage 1 distills sparse relevance from dense attention:

$$\mathcal{L}_{\text{idx}} = \sum_{\ell,i} \mathrm{KL} \left(a_{\ell,i} \,\|\, \mathrm{softmax}(s_{\ell,i}) \right). \tag{12}$$

Stage 2 learns NULL/access from matched recall/no-recall pairs. Stage 3 adapts language and tool behavior to the modified attention path using continued next-token training followed by instruction/tool SFT. The discrete hard top-k indexer is trained by distillation, while LoRA or full language projections receive the generation objective.

Table 5: Evaluation surfaces and the evidence each contributes.

Surface	Tasks	Primary metric	Role
Native trace probe, one archived event (Qwen3-4B)	1,000	Exact-copy accuracy	Where to open: trigger versus random opening in a native sparse model
Native trace probe, sixteen archived events (Qwen3-4B)	300	Exact-copy accuracy	The same question with a larger archive and in-archive token selection
WebShop Text, observation folding (Qwen3-30B-A3B)	2×512	Nonzero reward; mean reward	Selection over volume in an agent
DeepSeek detail suite	26	Hidden-detail recovery	Frozen native-DSA identity under three scheduled compactions
DeepSeek transfer suite	4	Reward; action steps	Fixed-total end-to-end transfer with a pretrained native indexer

NATIVE-DSA TRANSFER

DeepSeek-V3.2 already contains a pretrained sparse indexer. We therefore freeze token-identity scoring and fit the compact access/adoption boundary on native index features. This preserves the model's pretrained relevance structure and isolates the new learning problem: deciding when archived evidence should participate in a post-compaction decision. The fixed-total contract holds attention width constant, while the visible-preserving contract ensures that opening adds evidence without evicting the visible selection.

C EVALUATION PROTOCOL

BENCHMARK ROLES

POLICY SEMANTICS

- VISIBLE-ONLY uses core state, visible episode representations, and the recent raw window; archived raw tokens remain ineligible.
- FULL-HISTORY replays the clean uncompressed raw trajectory and serves as an information reference when it fits.
- ALWAYS-OPEN removes query-conditioned selection by opening every archived record allowed by the same expansion budget; in the agent study it measures stale-history pollution. In the native probe, making the whole trace visible serves as the information reference.
- RANDOM replaces the query-conditioned decision with an independent coin flip at a fixed probability per candidate. It is probability-matched, not exposure-matched, so it is a no-conditioning control. In the agent study the probability is $P^* = 0.332$ with salt `obsfold_exposure_20260924`, calibrated on development slices 00184–00207 and then frozen; realized exposure is 0.63 expansions per post-compaction step, against 0.70 for TRIGGERED-SA and 1.86 for ALWAYS-OPEN. In the native probes it opens each (layer, query) position independently at probability 0.5406.
- TRIGGERED-SA controls eligibility with NULL/OPEN. In the model-native experiments it keeps archived text outside the prompt and delegates token reading to the sparse indexer; in the agent study the opened record's raw observations are restored to the agent's context (Table 6).

PAIRING AND STATISTICS

Endpoint policies share task ids, benchmark-specific memory budgets, episode representation policy, recent window, model checkpoint, and decoding rule. WebShop and native-probe outcomes are binary per task and are paired by task identity with two-sided exact McNemar tests, and reward differences use paired task-bootstrap intervals with 10^4 resamples. Each endpoint comparison has one registered primary contrast. The hidden-content dyads use a two-sided exact sign test. Post-fold

Table 6: What each experiment executes. "Opening unit" is the granularity of the access decision; "reading" is how an opened archive reaches the model.

Experiment	Model	Gate	Opening unit	Reading	Status
Native probes (Table 1)	Qwen3-4B, converted DSA + sparse-path LoRA	Score-margin rule, threshold -3.65 (training-free)	(layer, query) position	Native top-k over visible + eligible hidden; top-$k = 1024$	Registered; 1,000 + 300 fresh traces
WebShop agent study (Table 7)	Qwen3-30B-A3B	Logistic gate on 8 structural features, threshold 0.005	Candidate archived record per step	Raw observations restored to context	Two registered waves, 2×512 tasks
DeepSeek detail suite	DeepSeek-V3.2, frozen indexer	Archive opened at the query	Query	Native top-k	26 queries, 3 scheduled folds
DeepSeek trajectory and four-task suite	DeepSeek-V3.2, frozen indexer	Native top-k membership (no separate gate)	(layer, query)	Fixed-total union top-k, 2,048 tokens	4 matched tasks
DeepSeek NULL/adoption head	DeepSeek-V3.2 features	897-parameter head	Decision	– (classification only)	32 identity-disjoint pairs
Content dyads	Qwen3-4B	None (correct block supplied)	–	Attention admission	8 dyads
Gate-training AUROC	Qwen3-4B	Layer-9 access head	Decision	– (classification only)	25 identity-disjoint pairs

detail recovery uses exact McNemar testing. Multi-seed gate calibration reports mean and sample standard deviation across three seeds.

Reported Observability

Each endpoint record contains the task id, model and artifact hashes, memory budget, compaction count, expanded episodes, selected hidden-record count, selected hidden tokens by layer when available, visible-token count, tool calls, repeated actions, format validity, reward, and final memory state. This ledger separates memory-state effects from truncated generation, malformed tool syntax, or runs without observed compaction.

D Expanded Results

Throughout this appendix, a WebShop "completion" is a nonzero-reward outcome ($R > 0$).

Experiment–Implementation Map

Native Probes

Each trace is a synthetic agent execution log of 20 tool events, in which one event records the queried code; the answer must be copied exactly. In the one-event probe only that event is archived (107 hidden tokens on average); in the sixteen-event probe the 16 oldest events are archived (1,702 hidden tokens on average, the target always among them) and the 4 most recent stay visible. The model is Qwen3-4B with the converted DSA indexer and sparse-path adaptation (one checkpoint throughout), top-$k = 1024$, a forced recent window of 256 tokens, and the score-margin trigger at -3.65, set on 2,000 calibration traces. Random opening draws each (layer, query) position independently with probability 0.5406, the trigger's opening rate measured on earlier held-out traces. Both probes were registered before any output existed and use traces never scored before (offsets 4000–4999 and 5000–5299).

Closed archive, random opening and TRIGGERED-SA answer 0, 0 and 342 of 1,000 in the one-event probe and 0, 0 and 123 of 300 in the sixteen-event probe; the registered primary contrast, random $\rightarrow$ TRIGGERED-SA, gives 342 wins and 0 losses ($p = 2.2\times10^{-103}$) and 123 and 0 ($p = 1.9\times10^{-37}$). Realized opening rates are 0.33 and 0.90 for TRIGGERED-SA and 0.54 for random. On the first 250 one-event traces, random opening at exactly the trigger's realized rate (0.33) also answers none (71 wins, 0 losses for TRIGGERED-SA), and lowering the trigger threshold from -3.65 to -4.5 opens 0.53 of positions and raises accuracy from 71 to 129 of 250, so the threshold trades visible-path preservation against recall. A further registered control on the same 300 sixteen-event traces opens at random with probability 0.896, the trigger's realized rate there: it opens 0.896 of positions and selects 917 hidden tokens against the trigger's 918, yet answers 14 against 123

Table 7: Controlled observation-folding evaluation (Qwen3-30B-A3B, never-scored WebShop tasks, byte-identical prompts across arms). $R > 0$ counts tasks with nonzero reward out of 512 per wave, with a two-sided exact McNemar p against TRIGGERED-SA; $\bar{R}$ is mean reward, Rep. repeated actions per task, and Steps actions per task, over both waves; E is archive expansions per post-compaction step.

	Wave 1		Wave 2 (confirm.)		Pooled, $n = 1{,}024$			
Policy	$R > 0$	p	$R > 0$	p	$\bar{R}\uparrow$	Rep.↓	Steps↓	E
VISIBLE-ONLY	148	0.37	152	0.27	0.168	17.1	22.7	0.00
RANDOM	134	**0.015**	132	7.5×10^{-4}	0.144	17.4	23.3	0.63
ALWAYS-OPEN	144	0.14	147	0.10	0.154	16.4	22.6	1.86
TRIGGERED-SA	**157**	–	**163**	–	**0.171**	**16.0**	**21.9**	0.70

Table 8: Paired reward differences of TRIGGERED-SA over each comparison in the controlled WebShop waves (95% task-bootstrap CIs, 10^4 resamples).

Comparison	Wave 1	Wave 2	Pooled ($n = 1{,}024$)
RANDOM	$+0.027\,[+0.007, +0.047]$	$+0.028\,[+0.010, +0.047]$	$+0.027\,[+0.014, +0.041]$
ALWAYS-OPEN	$+0.012\,[-0.004, +0.027]$	$+0.022\,[+0.003, +0.040]$	$+0.017\,[+0.005, +0.028]$
VISIBLE-ONLY	$+0.007\,[-0.011, +0.025]$	$-0.001\,[-0.020, +0.018]$	$+0.003\,[-0.010, +0.016]$

(113 wins, 4 losses, $p = 9.2\times10^{-29}$; audit `qwen3_native_dsa_distractor_matched_random_audit_20260926.json`, SHA-256 `94ab4c0cf3d40dff...`). At open positions of the sixteen-event probe the indexer retains 95% of the answer tokens under TRIGGERED-SA and 74% under random opening, against 60% for a blind choice of the same number of hidden tokens, so selection inside the archive is itself query-conditioned. Random opening nevertheless answers none: 233 of its 300 answers copy a code from a visible event instead, consistent with scattered openings failing to sustain the access that exact copying of a six-digit code needs across decoding steps. A duplicate run of the TRIGGERED-SA arm reproduced every prediction on all 1,000 traces. The audits are `qwen3_native_dsa_random_control_audit_20260926.json` (SHA-256 `9c572f02d24ad80d...`) and `qwen3_native_dsa_distractor_probe_audit_20260926.json` (`f68921bfeb80c50a...`).

CONTROLLED WEBSHOP WAVES

Both waves use the observation-folding archive, Qwen3-30B-A3B, and the frozen threshold 0.005 and random probability 0.332, on never-scored tasks 03448–03959 and 03960–04471 (64 eight-task slices each). The generation prompt carries no memory-policy statistics, and the audit verifies that the four arms' first-step prompts are byte-identical and that all arms take identical actions up to the first archive opening on every task; after that, contexts differ only in the archived records each policy opened. Each audit checks that every registered task is present in every arm, that per-task evaluation seeds and instruction hashes match across arms, that the visible arm never expands an archived record, and that every executed source file matches a build receipt taken during the run; no slice was excluded.

Over the 128 slices, TRIGGERED-SA completes more tasks than random opening in 56, ties in 48 and completes fewer in 24. Full matches ($R = 1$) are rare in every arm (8, 10, 9 and 9 of 512 for closed, TRIGGERED-SA, random and indiscriminate access in the first wave; 6, 5, 7 and 6 in the second). The audits are `obsfold_blinded30b_03448_03959_audit_20260926.json` (SHA-256 `e9780afd60a555c7...`) and `obsfold_blinded30b_w2_03960_04471_audit_20260926.json` (`f8fb8f135f435a06...`).

Table 9: Per-arm metrics over both controlled WebShop waves ($n = 1{,}024$ paired tasks). Rep. is repeated actions per task and "Tasks rep." the number of tasks with any repeated action; Comp. is compactions per task; Vis. tok. is visible tokens per post-compaction step; E is archive expansions per post-compaction step. No arm produced a format error.

Policy	$R > 0$	$R \geq 0.5$	$R = 1$	$\bar{R}$	Steps	Rep.	Tasks rep.	Comp.	Vis. tok.	E
VISIBLE-ONLY	300	211	14	0.168	22.7	17.1	766	12.5	860	0.00
RANDOM	266	177	16	0.144	23.3	17.4	801	13.0	1172	0.63
ALWAYS-OPEN	291	190	15	0.154	22.6	16.4	815	12.6	1747	1.86
TRIGGERED-SA	320	206	15	0.171	21.9	16.0	747	12.0	1135	0.70

Table 10: Access supervision on 25 identity-disjoint ScienceWorld validation pairs (layer-9 head; 50 validation rows). All rows share data, seed, parameter count, steps, and forward passes per pair.

Condition	Gate AUROC
Stage 2 disabled	68.00%
Compute-matched pointwise access	75.84%
Pairwise recall vs. NULL	**87.20%**
Pairwise access, three seeds	$85.28\% \pm 1.67\%$

GATE TRAINING

DEEPSEEK-V3.2 NATIVE TRANSFER

MATCHED DECISION TRACE

The fixed width holds attention capacity constant, the absence of prompt reinsertion isolates attention-side access, and reproduction by the original full-context top-k path confirms that the selected hidden evidence follows the model's pretrained relevance structure. Together, the detail suite and the fixed-total trajectory connect native token recovery to action-level utility without retraining token identity.

E STORAGE, HARDWARE, AND ADMISSION COST

For the Qwen3-4B bf16 configuration, standard K/V requires 144 KiB per token and the shared DSA index key adds 9 KiB per token. Table 13 reports binary units. The 32K admission measurements use NVIDIA H200 GPUs; the recorded software environment uses CUDA 12.8, PyTorch 2.8.0, and Transformers 4.56.2.

The copy-only column scales linearly with archive size, as it should for a bandwidth-bound transfer: pinned-CPU copies run at 53–55 GB/s at every size, and GPU-resident copies stay below 3 ms even at 32K tokens. The end-to-end 32K admission is higher than the copy floor on both tiers, so most of GPU-resident admission is work other than moving bytes, which is where an optimised implementation has headroom. The explicit NULL path makes admission conditional. Closed decisions avoid archive transfer and hidden attention, while storage tier and staging policy can be selected independently of the gate and sparse indexer. The copy benchmark is `archive_admission_latency_20260925.json` (SHA-256 `2b96b36f1ec5bb50...`; PyTorch 2.9.1, CUDA 12.8, H200).

F ANONYMOUS ARTIFACT MAP

The accompanying code ZIP follows the paper's verification path from memory management to sparse selection and endpoint analysis. It contains no Git metadata, cluster credentials, user paths, model weights, or third-party benchmark assets.

Table 11: Native-DSA transfer results with the pretrained indexer frozen.

Test	Result	Interpretation
Post-fold detail recovery	0 / 21 / 24	VISIBLE-ONLY / native archive access / all-visible reference, of 26 queries
Exact paired test	$p = 9.54 \times 10^{-7}$	Native access versus VISIBLE-ONLY under exact McNemar testing
Fixed-total WebShop decision	$R : 0 \to 1.0$; $19 \to 12$ steps	The same 2,048-token attention width is used in both arms; hidden tokens are selected across all 61 layers
Four-task matched suite	3 wins, 1 tie, 0 losses	Mean reward $0 \to 0.443$ and 18.6% fewer steps
NULL/adoption calibration	897 parameters	$82.52\% \pm 0.54\%$ AUROC, $80.73\% \pm 0.90\%$ balanced accuracy, 12.5% false-open rate, and 93.8% pair accuracy

Table 12: A matched DeepSeek-V3.2 trajectory across the compaction fold (common three-step prefix).

Moment	VISIBLE-ONLY	TRIGGERED-SA
Common prefix	Reads the decisive product evidence and follows the same trajectory.	Reads the decisive product evidence and follows the same trajectory.
After repeated folds (15 and 8)	Core, episode representations, and recent actions remain visible; earlier raw evidence is ineligible.	The same visible state remains intact; earlier raw evidence is addressable but closed.
Final decision	Selects 2,048 visible tokens; archived evidence cannot enter the competition.	Selects 2,048 tokens in total; native top-k admits archived evidence across all 61 layers.
Outcome	No reward after 19 steps.	**Reward 1.0 after 12 steps (36.8% fewer).**

SUGGESTED VERIFICATION ORDER

1. Run the lightweight unit suite for layered memory and visibility masks.
2. Run closed-path equivalence tests with a nonempty synthetic archive.
3. Run the native synthetic-trace probe with the trigger and the rate-matched random arm, and check that padded selection slots are masked (closed positions must reproduce the visible-only output).
4. Run gate evaluation on the included derived-label split and verify task-identity separation.
5. With local model weights, run the DeepSeek visibility smoke test and inspect per-layer selection receipts.
6. Run the paired analysis scripts to reproduce tables from the included aggregate result records.

G REPRODUCTION PACKAGE

The artifact exposes the complete paper-to-code map, focused behavioral tests, benchmark entry-points, example configurations, aggregate result records, and SHA-256 hashes. Model weights and benchmark environments are resolved through ordinary local path arguments, so the included tests and table-regeneration workflow run without private infrastructure. This separation keeps third-party assets under their original licenses while preserving a direct verification path for compaction, visibility, sparse selection, access calibration, and reported aggregate results.

Table 13: Absolute archive accounting and warm admission cost. "Admission" is the end-to-end measurement reported for the 32K archive; "copy" is a separate micro-benchmark of only the host-to-device or device-to-device transfer of the same K/V and index-key bytes (median of 50 warm, CUDA-event-timed repetitions), which isolates the data-movement floor and extends it to every archive size.

				Admission		Copy only	
Archive tokens	Standard K/V	Index keys	Combined	GPU	Pinned CPU	GPU	Pinned CPU
1,024	144.0 MiB	9.0 MiB	153.0 MiB	–	–	0.18 ms	3.01 ms
8,192	1.125 GiB	72.0 MiB	1.195 GiB	–	–	0.73 ms	23.3 ms
32,768	4.500 GiB	288.0 MiB	4.781 GiB	12.2 ms	105.8 ms	2.54 ms	92.9 ms

Table 14: Map from paper concepts to the anonymous code artifact.

Concept	Principal implementation	Included verification
Layered budget compaction	`src/mini_swe_agent.py`	Profiles, raw fidelity, and segmented-policy contracts
Qwen sparse attention and visibility	`src/dsa_qwen.py`	Masks, selected-token budgets, and artifact loading
DeepSeek triggered visibility	`src/deepseek_v32_exp_hierarchical.py` (`triggered_visibility_*`)	Eligibility, fixed-total top-k, and native-indexer patch contracts
Visible-preserving hidden residual	`src/deepseek_v32_visibility_fusion.py`	Normalization, zero residual, and runtime integration
Layer/query observability	`src/deepseek_v32_layer_token_runtime.py`	Per-layer selection and coverage accounting
Access/adoption training	`src/open_gate_utility_training.py`	Pair inventory, calibration, and binary metrics
Dense index distillation	`src/layerwise_block_distillation.py` and training scripts	Selection diagnostics and artifact checks