ASK THE MEMORY, NOT THE TEACHER: EPISODE-SWAP GATING KEEPS STALE MEMORIES OUT OF DISTILLED AGENT WEIGHTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Agents that keep a memory of past episodes are increasingly consolidated into their weights. A teacher that reads a retrieved episode supervises, token by token, a student that does not, until the student no longer needs the store. We show that this on-policy consolidation writes more into the weights than it should. A retrieved episode mixes the procedure that solves a family of tasks with values that held only in that episode, such as an access code or the place where an object happened to be, and dense distillation transfers both. The consolidated student then states a past episode's credential as a fact about a new user, with no retrieval record left to trace it. Standard token-selection signals miss this failure, because a teacher that copies from its context is confident, and the copied values can appear anywhere in a response. We propose to test each token against the memory instead of trusting the teacher. When the retrieved episode is replaced by a sibling episode of the same task family, the teacher's predictions of procedural tokens stay the same while its predictions of episode-bound tokens change, and we use this difference as a per-token distillation gate that costs two extra teacher forward passes with one sibling. In a diagnostic environment where the provenance of every token is known, the gate separates the two kinds of tokens almost perfectly while confidence and position heuristics barely beat chance. During consolidation it eliminates stale-value emission at equal task success, nearly doubles honest abstention, and carries over to a second model family and to embodied agents in ALFWorld.

1 INTRODUCTION

Language-model agents increasingly keep a memory of past episodes and retrieve from it when a similar task returns, whether as retrieved documents (Lewis et al., 2020), episodic memories (Packer et al., 2023; Park et al., 2023; Zhong et al., 2023; Chhikara et al., 2025), skill libraries (Wang et al., 2023; 2024), or distilled insights (Shinn et al., 2023; Zhao et al., 2023; Fang et al., 2025), and a growing literature studies how such memories should be organized and retrieved (Zhang et al., 2024; Gutiérrez et al., 2024; Xu et al., 2025). A recent line of work goes one step further and consolidates the memory into the model's weights (Ye et al., 2026b; Liu et al., 2026b;c; Lazaridis et al., 2026; Ye et al., 2026a). The model itself, conditioned on a retrieved episode, acts as a teacher for a copy of itself without memory, and on-policy distillation transfers the teacher's behavior to the student on trajectories the student generates. After consolidation the agent needs neither the store nor the retrieval step.

A retrieved episode, however, contains two kinds of information. Its procedure, for example the sequence of operations that resolves a class of support tickets, recurs in every episode of the task family and is what consolidation should transfer. Its specific values, such as the access code issued to one user or the shelf where an object was found, held only in that episode. The teacher does not distinguish the two. In our diagnostic environment a 7B teacher copies the stored access code into a request from a different user on 28% of requests, in line with evidence that retrieved context strongly steers model outputs (Xie et al., 2023). Distillation transfers these copied values together with the procedure, and the consolidated student later produces them with no memory in context (Figure 1). We call this failure *stale-memory engraving*. It is a hallucination (Ji et al., 2022) that is hard to audit,

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.

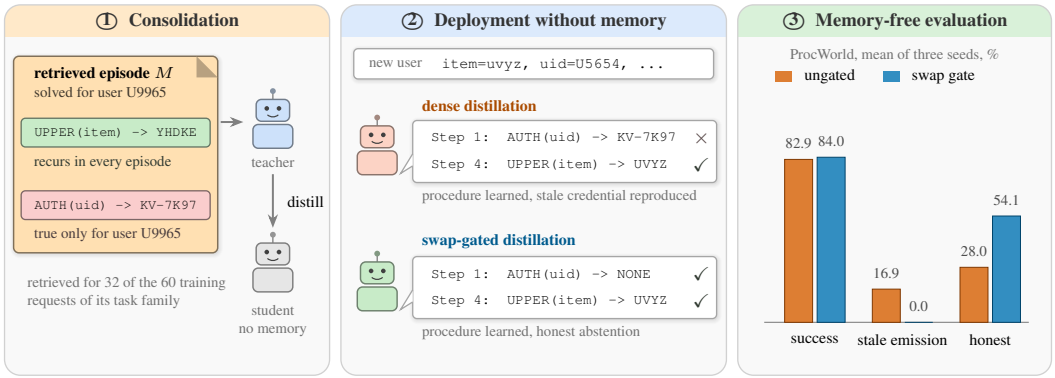

Figure 1: **Stale-memory engraving.** A retrieved episode holds a procedure that recurs and a credential that holds for one user. For a new user without memory, the densely distilled student reproduces the stored credential while the swap-gated student keeps the procedure and abstains. Panel 2 shows real greedy outputs of the seed-0 students, and panel 3 summarizes Table 3.

since no retrieval record links the output to the episode it came from, and it occurs in every ungated consolidation run we perform, with the student emitting stored values on 17% of held-out requests.

Existing token-selection methods for on-policy distillation decide where to supervise from the teacher's confidence or from a token's position in the response (Liu et al., 2026a; Xu et al., 2026). Neither signal identifies episode-bound tokens. A teacher that copies a value from its context is confident about it, and copied values appear at the start of some responses and at the end of others.

We therefore test each token against the memory rather than against the teacher. If the retrieved episode is replaced by a sibling episode of the same task family, the teacher's prediction of a procedural token stays the same, while its prediction of an episode-bound token changes. Comparing the teacher's predictions under the retrieved episode, under no episode, and under a sibling separates the two kinds of tokens without labels. We turn this comparison into a per-token weight that removes episode-bound tokens from the distillation loss, and call the method episode-swap gating. Our contributions are as follows.

- We identify stale-memory engraving in on-policy memory consolidation and measure it at the level of individual tokens.
- We propose episode-swap gating, a per-token test that needs no labels during training, costs two extra teacher forward passes with one sibling, and fits into existing consolidation losses.
- We introduce ProcWorld, an environment in which every token of a reference answer carries an exact provenance label. On ProcWorld the gate detects episode-bound tokens with AUROC of at least 0.94 and removes stale-value emission at equal task success across three seeds, two model families, and a naturalistic rendering. On ALFWorld the gate transfers to embodied agents and makes consolidation robust to mismatched retrieval.

2 Background and Problem Setting

On-policy memory consolidation. An agent solves tasks x drawn from task families, where a task family is a set of tasks solved by the same procedure on different inputs. An episodic store $\mathcal{M}$ holds solved episodes, and a retriever returns an episode $M \in \mathcal{M}$ of the same family for each new task. Two episodes of the same family are called siblings. Let π_0 be the base model. The teacher is π_0 with the retrieved episode in its context and predicts $p(\cdot \mid s, M)$ at a state s made of the task and the response so far. The student π_θ is π_0 with a LoRA adapter (Hu et al., 2021) and no memory, so a single model plays both roles by toggling the adapter. The student samples a response $y \sim \pi_\theta(\cdot \mid x)$,

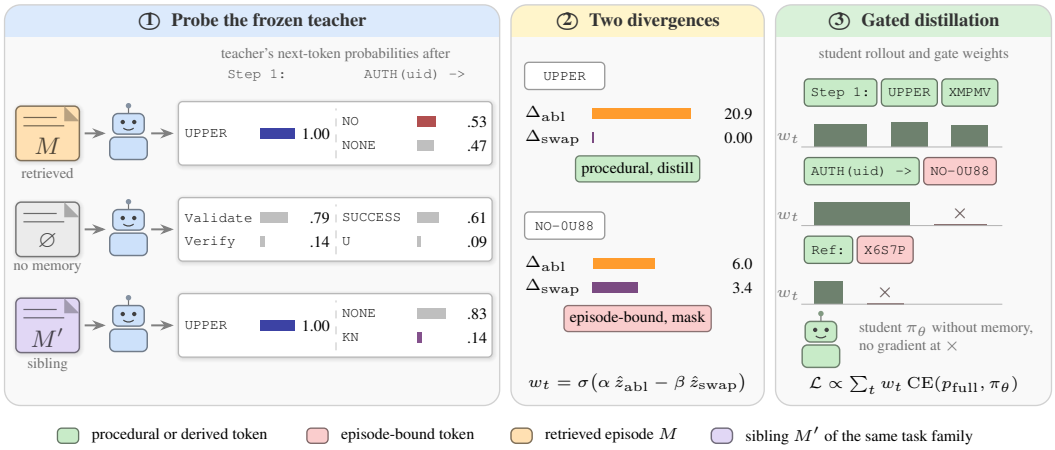

Figure 2: **Episode-swap gating on the request of Figure 3.** All probabilities and divergences are measured with the 7B teacher. Both tokens depend on memory, but only the credential changes when the retrieved episode is replaced by a sibling, so only the credential is masked.

the teacher is evaluated along it at every prefix $s_t = (x, y_{<t})$, and consolidation minimizes

$$\mathcal{L}_{\mathrm{OPD}}(\theta) = \mathbb{E}_{y\sim\pi_\theta}\Big[\frac{1}{T}\sum_{t=1}^{T}\mathrm{KL}\big(p(\cdot \mid s_t, M)\,\big\|\,\pi_\theta(\cdot \mid s_t)\big)\Big], \tag{1}$$

the forward-KL form of on-policy distillation (OPD) (Agarwal et al., 2023; Gu et al., 2023). To avoid a cold start, part of each batch replaces the student's sample with a fixed reference response to the same task (Agarwal et al., 2023), and the teacher is evaluated along it in the same way. We call the resulting mix of sampled and reference prefixes mixed-policy states. Every position receives the same weight, so whatever the teacher predicts at s_t becomes the student's target.

Token provenance. We distinguish three kinds of target tokens by what determines them. *Procedural* tokens, such as operation names and the layout of the answer, are fixed by the task family and recur across its episodes. *Derived* tokens, such as the uppercased city of the current record, are computed from the current input. *Episode-bound* tokens copy values specific to the retrieved episode, such as a stored credential, a reference number, or an object location. Procedural and derived tokens are what consolidation should transfer. An episode-bound token is correct for the retrieved episode and wrong for the current task.

Stale-memory engraving. When the teacher places probability on an episode-bound value, Eq. 1 trains the student to produce that value without the episode in context. We say that a consolidated student engraves stale memory when, evaluated without memory, it outputs a value stored in a training episode at a position where the correct output is derived from the current input or is an abstention. The stale emission rate is the fraction of held-out requests with at least one such output.

3 Episode-Swap Gating

Equation 1 would be safe if episode-bound tokens were excluded from it. We identify them by measuring how the teacher's prediction depends on the retrieved episode (Section 3.1), turn the measurement into a per-token weight (Section 3.2), and train with the weighted loss (Section 3.3). Figure 2 walks through the procedure on a real request.

3.1 Two Counterfactual Probes

At a student-visited prefix s_t we query the frozen teacher with the retrieved episode M, with an empty memory, and with each of J siblings $M'_1, \ldots, M'_J$ of M, which we call the swap set,

$$p_{\mathrm{full}} = p(\cdot \mid s_t, M), \qquad p_{\varnothing} = p(\cdot \mid s_t), \qquad p^{(j)}_{\mathrm{swap}} = p(\cdot \mid s_t, M'_j). \tag{2}$$

We compare them with a deletion divergence and a swap divergence,

$$\Delta_{\mathrm{abl}}(t) = \mathrm{KL}\big(p_{\mathrm{full}} \,\|\, p_{\varnothing}\big), \qquad \Delta_{\mathrm{swap}}(t) = \frac{1}{J}\sum_{j=1}^{J} \mathrm{KL}\big(p_{\mathrm{full}} \,\|\, p^{(j)}_{\mathrm{swap}}\big). \tag{3}$$

The contexts differ only in the memory block and share the prefix, so each distribution comes from one batched teacher forward pass over the student's response. Δ_{abl} is large when the memory influences the prediction, and Δ_{swap} is large when the prediction depends on which episode was retrieved. A procedural token needs the memory but not a particular episode, so any sibling supports the same prediction and Δ_{swap} stays near zero. An episode-bound token follows the retrieved episode, so Δ_{swap} is large.

Figure 2 shows both cases on one request. After `Step 1:` the teacher predicts the operation `UPPER` with probability 1.00 under both the retrieved and the sibling episode, but predicts `Validate` without memory, which gives $\Delta_{\mathrm{abl}} = 20.9$ and $\Delta_{\mathrm{swap}} = 0.00$. At the stored credential `NO-0U88`, the probability of its first token is 0.53 under the retrieved episode and 0.0005 under the sibling, which gives $\Delta_{\mathrm{abl}} = 6.0$ and $\Delta_{\mathrm{swap}} = 3.4$. Deletion alone cannot separate these two tokens, because both depend on the memory. The swap separates them because it keeps the task family of the memory fixed and changes only which episode is retrieved.

3.2 From probes to a gate

The two divergences have different scales. We therefore standardize each with a mean and standard deviation estimated once, before training, on a calibration pool of unlabeled probed trajectories, and write the results as $\hat{z}(\Delta_{\mathrm{abl}})$ and $\hat{z}(\Delta_{\mathrm{swap}})$. The weight of token t is

$$w_t = \sigma\big(\alpha\, \hat{z}(\Delta_{\mathrm{abl}}(t)) - \beta\, \hat{z}(\Delta_{\mathrm{swap}}(t))\big), \tag{4}$$

where σ is the logistic function. The first term raises the weight of tokens that rely on the memory, which are the tokens consolidation exists to transfer. The second lowers the weight of tokens that rely on a specific episode. Episode-bound tokens also have a large Δ_{abl}, so the second term must dominate, and we set $\beta \gg \alpha$. Tokens that the memory does not affect lie below the mean on both axes and receive weights close to one, which is harmless, since the teacher's prediction at those positions does not come from the episode. Choosing (α, β) is the only step that uses provenance labels. In ProcWorld we select them by detection AUROC on a labeled development split, and the same value $(0.25, 8)$ is selected in all four ProcWorld settings (Appendix D). ALFWorld has no token labels and reuses this value.

3.3 Gated consolidation

The student minimizes the weighted objective

$$\mathcal{L}(\theta) = \frac{\sum_t w_t\, \mathrm{CE}\big(p_{\mathrm{full}}(\cdot \mid s_t),\, \pi_\theta(\cdot \mid s_t)\big)}{\sum_t w_t}, \tag{5}$$

which matches Eq. 1 up to a constant when every $w_t = 1$. Normalizing by $\sum_t w_t$ keeps the effective learning rate comparable across gates. A token with $w_t \approx 0$ contributes no gradient, so the student keeps its prior behavior at that position. That prior comes from a format warmup, identical across conditions, that fine-tunes the student on honest answers (Section 5), so a masked credential slot is filled with an abstention token, in practice `NONE`, rather than a stored value. Beyond the teacher pass that distillation already needs, the gate requires one pass without memory and one per sibling, which is two extra passes with the single sibling that suffices in our ablations. It needs no additional model and changes only the per-token weights of an existing distillation loss, so it applies to any consolidation method of the form of Eq. 1. Algorithm 1 summarizes training.

3.4 Variants of the probe

The experiments use the probe in the forms of Table 1. The swap score $-\Delta_{\mathrm{swap}}(t)$ ranks tokens for detection and never weights a loss. The calibrated gate, Eq. 4 with $(\alpha, \beta) = (0.25, 8)$, weights the loss of the gated ProcWorld students, and the untuned gate with $\alpha = \beta = 1$ is its ablation. On ALFWorld, siblings come from different scenes, so procedural actions also vary under the swap. The floored gate $\max(w_t, 0.3)$ therefore keeps a minimum weight on every token (Section 5.3).

Algorithm 1 Swap-gated on-policy distillation

1: fit $\hat{z}$ on the calibration pool; run the format warmup
2: **for** each step **do**
3: roll out π_θ memory-free, or with probability 0.5 use the reference response
4: probe the teacher under M, $\varnothing$, and $M'_{1..J}$; compute Δ_{abl}, Δ_{swap}, and w_t
5: update θ on the weighted loss of Eq. 5
6: **end for**

Table 1: **Variants of the swap probe.** Labels are needed only to choose (α, β) for the calibrated gate. Leak weight is the weight on an episode-bound token, and passes count teacher forward passes per response with J siblings.

Variant	Used for	Quantity	Labels	Leak weight	Calibration	Passes
Swap score	detection	$-\Delta_{\text{swap}}(t)$	none	not trained	none	$1{+}J$, $J{=}3$
Calibrated gate	ProcWorld	w_t, Eq. 4	dev split	≈ 0	40 scripted answers	$2{+}J$, $J{=}2$
Untuned gate	ablation	w_t, $\alpha{=}\beta{=}1$	none	≤ 0.5	40 scripted answers	$2{+}J$, $J{=}2$
Floored gate	ALFWorld	$\max(w_t, 0.3)$	none	0.3	12 expert games	$2{+}J$, $J{=}1$
Ungated OPD	baseline	$w_t = 1$	none	1	none	1

4 PROCWORLD: GROUND TRUTH PER TOKEN

Scoring a gate requires knowing, token by token, whether the teacher's signal is procedural or episode-bound. No natural corpus records this, so we construct an environment that does. In ProcWorld, each task family is an invented protocol of three or four deterministic string and arithmetic operations plus one `AUTH` lookup, applied to a record. Protocols are private by construction, so a memory-free 7B model executes the correct operation sequence 0% of the time, yet one retrieved episode is enough for the same model to imitate protocols almost perfectly, with 91.7% operation-sequence accuracy and 80% correct final results. Each episode also plants three episode-bound values that invite copying, namely a reference code whose position varies, an `AUTH` credential mid-procedure, and a free-text case note. We call the three answer fields that carry them leak channels. For a fresh request the honest outputs are `NONE`, `UNKNOWN`, and `none`. The greedy teacher instead copies the stored values on 15.0%, 28.3%, and 3.3% of requests. Retrieval over the eight episodes per family is Zipf-skewed, so a few hub episodes serve most requests, as in similarity retrieval. We render answers from labeled segments, so each token carries an exact provenance label (Figure 3). Honest answers abstain on every leak channel. Scripted teacher-behavior answers apply the protocol to the current record and copy the retrieved episode's three stored values, the behavior the teacher shows when it copies. The detection probes, the calibration pool, and the reference responses of mixed-policy training all use teacher-behavior answers, so every such trajectory contains leak tokens and gates can be scored as detectors. A naturalistic twin, ProcWorld-NL, re-renders episodes as prose ticket notes and steps as verb phrases, so surface form and semantics vary independently.

Each design choice removes a confound. Invented protocols make the memory-free baseline exactly zero, so nothing the student later does can be credited to pretraining. The leak channels open, interrupt, and close the answer, so no positional heuristic can succeed by accident. The credential is unknowable in principle, so the honest answer is well-defined. Zipf retrieval concentrates teacher mass on the values of hub episodes, the regime in which engraving is strongest.

5 EXPERIMENTS

Setup. All experiments run on 40GB A100 GPUs with PyTorch (Paszke et al., 2019), Transformers, and PEFT. Unless noted, teacher and student share Qwen2.5-7B-Instruct (Yang et al., 2024). The four ProcWorld settings pair this model with the symbolic surface (7B) and with ProcWorld-NL (7B-NL), or replace it with Qwen2.5-3B-Instruct (3B) or Mistral-7B-Instruct-v0.3 (Jiang et al., 2023) (Mistral). Students train a LoRA adapter (Hu et al., 2021) of rank 16 over all attention and MLP projections with AdamW (Loshchilov & Hutter, 2017) at learning rate $2{\times}10^{-4}$, batch 8, 400 distillation steps, and $J{=}2$ siblings. The format warmup runs 60 steps, half on memory-free honest

Retrieved episode (memory) `record: name=hucf, city=bxuvf, ..., uid=U6784`
`Step 1: UPPER(city) -> BXUVF  Step 3: AUTH(uid) -> NO-0U88`
`Ref: X6S7P  Note: billing=WPG-6524`

Current request `record: name=yshhc, city=xmpmv, ..., uid=U4476`

Teacher behavior, with per-token provenance

`Step 1: UPPER(city) -> XMPMV`	procedural / derived
`Step 3: AUTH(uid) -> NO-0U88`	**leak**: stale credential
`Ref: X6S7P  Note: billing=WPG-6524`	**leak**
Honest gold `AUTH -> UNKNOWN, Ref: NONE, Note: none`	

Figure 3: **One ProcWorld request.** The teacher applies the protocol correctly to the current record, then copies the episode's credential, reference code, and note. These are the tokens the swap probe flags and the gate masks.

Table 2: **Episode-bound token detection AUROC** on memory-affected tokens of the 70% evaluation split. One calibrated (α, β) serves all four settings.

Detector	7B	3B	7B-NL	Mistral
Swap score $-\Delta_{\text{swap}}$	**.981**	**.985**	**.966**	**.962**
Calibrated gate	.979	.982	.959	.940
Untuned gate $\alpha{=}\beta{=}1$	.930	.917	.908	.877
Position	.673	.667	.693	.676
Ablate-only	.649	.632	.586	.546
Entropy	.591	.677	.543	.638
Max-probability	.582	.671	.533	.632
Random	.502	.507	.498	.502

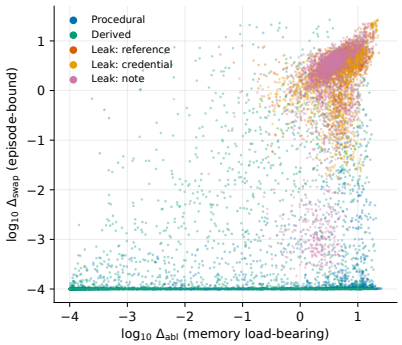

Figure 4: **Every probed 7B token** by its two divergences, colored by provenance. Leaks fill the upper right.

answers of four held-out families and half on honest answers to training requests with the episode in context. After it every run emits no stored value and reaches about 10% operation accuracy, so procedure learning and engraving both happen during distillation. The calibration pool holds 40 teacher-behavior answers per setting. For detection we probe 288 labeled trajectories, 12 per training family, standardize over all of them without labels, select (α, β) on a 30% development split, and report detection on the remaining 70%. Each consolidation seed regenerates the world and the initialization, and spread is mean±std over seeds 0 to 2.

Baselines. Each baseline scores a token by one signal, namely its relative position t/T as in position-weighted distillation (Liu et al., 2026a), its deletion divergence Δ_{abl} alone, the teacher's entropy, the teacher's maximum probability, or uniform noise. As training gates, the position gate uses $w_t = 1 - t/T$, the ablate-only and entropy gates use $\sigma(\hat{z}(\Delta_{\text{abl}}))$ and $\sigma(-\hat{z}(H_t))$ with teacher entropy H_t, the random gate draws w_t uniformly, and ungated distillation, also called dense distillation, uses $w_t = 1$. All conditions share the training recipe and differ only in w_t, except the 2×lr control.

Metrics. Detection is scored as the AUROC of a detector's per-token score against the episode-bound label. We restrict it to memory-affected tokens, whose deletion divergence exceeds the 60th percentile over all probed tokens, because the memory barely influences the others. Training outcomes are measured memory-free, with no episode in the student's context, on 600 held-out requests, which are 25 new records from each of the 24 training families. Success is exact match of the final result. Stale emission, defined in Section 2, is an exact-match lower bound on engraving over the three leak channels. A request counts as honest when all three leak channels hold an abstention token such as `NONE` or `UNKNOWN`. Operation accuracy scores the predicted operation sequence position by position, which measures procedure internalization and ignores slips in the arithmetic.

Table 3: **Memory-free evaluation** after 400 distillation steps on 600 held-out requests, mean±std over three seeds. Stale emission counts requests where the student outputs a value stored in some training episode. Honest requires an abstention on every leak channel. **Bold** marks the best column entry and underline the second best.

Condition	Success	Stale emission	Honest
Calibrated gate	84.0±5.9	**0.0±0.0**	**54.1±15.5**
Calibrated gate, J=1	**86.6±6.6**	**0.0±0.0**	49.4±27.1
Untuned gate α=β=1	84.8±7.4	9.8±7.0	36.6±14.8
Ungated OPD	82.9±5.1	16.9±3.1	28.0±12.4
Ungated, 2×lr	80.7±6.4	21.9±5.0	16.7±4.6
Random gate	86.1±7.8	16.3±2.8	24.7±16.8
Position gate	85.9±8.2	24.3±4.7	20.4±1.5
Ablate-only gate	83.7±7.1	27.7±3.3	17.2±4.2
Entropy gate	81.2±8.5	29.8±8.0	17.4±3.4

5.1 Detecting episode-bound tokens

Table 2 reports detection AUROC. Across two model sizes, a second model family, and the naturalistic surface, the swap score separates episode-bound from procedural tokens with AUROC between 0.96 and 0.99, and the calibrated gate stays at 0.94 or above. No baseline detector reaches 0.70. The label statistics explain the margin (Table 9). Procedural tokens carry high deletion divergence at 2.4 nats but essentially zero swap divergence at 0.01, leaked tokens carry both at 4.4 to 6.1 and 3.3 to 3.6, and the teacher copies most of them near-deterministically, with 64% of leaked tokens under 0.05 nats of entropy, so teacher confidence does not indicate that a token is safe. Across settings, entropy and max-probability drift between 0.53 and 0.68 while the swap score varies by about two points, because it measures how the prediction changes with the memory rather than how confident the teacher is. Figure 4 plots every probed token on the two axes, with the leak channels isolated in the upper right quadrant.

The probe relies on a diverse swap set. If the swap set happens to agree with the retrieved episode on some field, that field's swap divergence collapses and its leaks masquerade as procedure. Homogenizing the note field on purpose collapses its Δ_{swap} from 3.3 to 0.1 while the untouched credential channel is unchanged, and overall detection degrades gracefully from 0.981 to 0.887, with the loss confined to the homogenized field.

5.2 Consolidation with the gate

Table 3 reports memory-free evaluation after consolidation. The protocols are fully internalized under every condition, with memory-free operation accuracy above 90%, from 0% for the base model. The conditions differ in what else the student learns. Ungated distillation emits stored episode values on 16.9% of fresh requests, and the position, ablate-only, and entropy gates raise the rate to between 24 and 30%. A random gate merely matches the ungated rate, so reweighting as such does nothing unless it tracks the right signal. The calibrated gate emits none, with zero stale outputs across three seeds and all three channels, at success within seed variation of the other conditions and with honest abstention nearly doubled. In Figure 5 its stale-emission curve stays at zero while every other condition drifts upward.

Honest abstention. The honest column of Table 3 orders the conditions almost exactly by gate quality, from 54.1 under the calibrated gate down through 28.0 for ungated distillation and 17 to 25 for the baseline gates, because every episode-bound token that a gate fails to mask replaces the student's abstention with a stored value.

Cross-model replication. The same configuration on Mistral gives the same picture, slightly amplified, at 73.2% success with 0.0% stale emission against 65.0% and 24.0% for ungated distillation.

Ablations. Three ablations isolate the mechanism. Doubling the ungated learning rate makes stale emission worse rather than better at $21.9\pm5.0\%$, so the effect of the gate is not a change in effective learning rate. Cutting the probe to a single sibling preserves the zero at $86.6\pm6.6\%$ success, so one sibling suffices. The untuned gate leaks where Section 5.1 predicts. The credential channel has

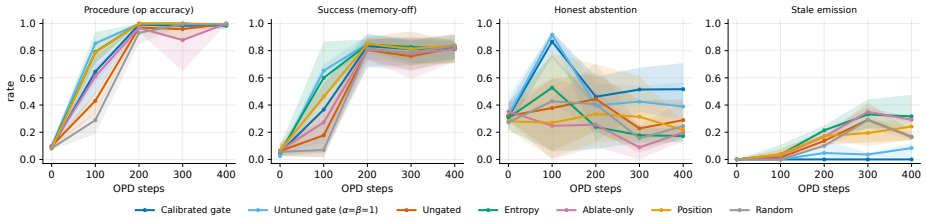

Figure 5: **Training dynamics on ProcWorld**, mean over seeds with bands spanning min to max. The calibrated gate reaches the success of the other conditions while holding stale emission at zero.

Table 4: **Transfer to embodied agents.** Success is in %. Block (c) averages three seeds, and block (d) uses one seed on 64 games.

(a) Effect of memory on the teacher	
Success, memory on / off	43.8% / 31.3%
Stale first-visit rate, memory on	34.4%
(b) Stale-visit detection AUROC, 1,810 actions	
Swap score	**0.774**
Ablate-only	0.696
Max-probability / Entropy	0.574 / 0.572
Position	0.323

(c) Consolidation, 96-game OOD split	
Floored gate, 3 seeds	42.7 ± 8.1
Ungated, 3 seeds	38.5 ± 11.0
Paired per-seed difference	$+5.2 / -4.2 / +11.5$
(d) Success, normal → mismatched retrieval	
Ungated	$51.6 \rightarrow 39.1$
Floored gate	$53.1 \rightarrow 50.0$

the highest deletion divergence, the symmetric gate therefore leaves it half open at $w \approx 0.5$, and $9.8 \pm 7.0\%$ of requests emit a stored value, almost all of it credentials. Raising β on the development split closes the channel completely without using test data.

Training recipe. Training without the format warmup, routing masked tokens to the memory-free teacher, and reverse KL each fail in our pilots (Appendix D), so all runs use forward KL on mixed-policy states.

Naturalistic surface. On ProcWorld-NL, which keeps the semantics and labels but renders episodes as prose, the calibrated gate ends 800 training steps at zero stale emission with 46.8% success against 44.3% for ungated distillation, whose emission peaks at 8.3% mid-training (Appendix D).

5.3 Transfer to ALFWorld

We port the full setting to ALFWorld (Shridhar et al., 2020), which grounds ALFRED household tasks (Shridhar et al., 2019) in a TextWorld interface (Côté et al., 2018). The store holds 360 expert trajectories, 60 for each of the six task types, and consolidation uses the four core types pick, clean, heat, and cool. Under normal retrieval the teacher receives a solved episode of the same task type from a different scene, drawn at random during consolidation and among episodes that mention the target object in blocks (a) and (b). The agent acts through a ReAct-style command loop (Yao et al., 2022), and multi-turn distillation runs over the agent's own dialogues. Students are evaluated memory-free on out-of-distribution scenes absent from training and from the store. Episode-level variance is large and seed-correlated, so we also report paired per-seed differences, the success of the gated student minus that of the ungated student trained with the same seed and evaluated on the same games. Table 4 collects the study.

Block (a) shows that memory both helps and misleads the teacher. Retrieved episodes lift its success from 31.3% to 43.8% and cut invalid actions threefold. We call a move to where the object sat in the retrieved episode's scene that finds nothing there a stale visit, the embodied counterpart of a stale credential. In 11 of 32 games the teacher's first search is a stale visit. Block (b) uses the stale visits of 96 teacher games as labels. The swap score leads every alternative at 0.774 AUROC, against 0.696 for deletion alone. The position heuristic falls below chance because stale visits occur early in a trajectory, whereas ProcWorld leaks skew late. The swap score needs no prior about where episode-bound tokens occur.

Consolidation on ALFWorld needed one adjustment. Siblings come from different scenes, so even procedural actions carry nonzero swap divergence through varying receptacle indices and step order. The unfloored gate therefore over-masks procedural actions and loses 14.1 and 7.8 points of success to ungated distillation at two seeds on a 64-game split. Subtracting the swap divergence between two siblings from Δ_{swap} does not remove this deficit at one seed, whereas the floor $w_t \leftarrow \max(w_t, 0.3)$ does. Block (c) shows that the floored gate reaches parity or better on the 96-game split, with a mean paired difference of $+4.2$, while episode-bound actions still receive a weight of 0.3. Block (d) makes retrieval mismatched, drawing episodes from scenes where the target object sits elsewhere. At one seed on the 64-game split, ungated distillation inherits the teacher's location bias and loses 12.5 points of success, while the floored gate loses 3.1.

6 Related Work

Consolidating memory into weights. Recent systems distill privileged context into students that lack it, including context-conditioned teachers in OPCD (Ye et al., 2026b), procedural memory in PMD (Liu et al., 2026b), exploration memory in EMPO^2 (Liu et al., 2026c), retrieved evidence in EDGE-OPD (Lazaridis et al., 2026), rewritten facts in CODE (Li et al., 2026), and an online stream of experience in OEL (Ye et al., 2026a). All of them distill every token the teacher emits. Episode-swap gating only reweights that per-token loss, so it can be added to any of these systems without changing what they retrieve or when they consolidate.

Knowledge distillation and token selection. Distillation matches a teacher's soft targets offline (Hinton et al., 2015; Kim & Rush, 2016) or on student trajectories (Agarwal et al., 2023; Gu et al., 2023; Ko et al., 2024), and recent work chooses where to supervise, handing control to the teacher at reasoning failures (Xu et al., 2026) or weighting tokens by position (Liu et al., 2026a). These criteria target positions where the teacher is more reliable than the student, whereas engraving arises where the teacher is reliable with respect to its own context.

Counterfactual provenance. Counterfactual memorization separates episodic from generalizable training data by leave-one-out retraining (Zhang et al., 2021), Min et al. (2022) perturb demonstration labels, and ContextCite attributes generations to context spans by ablation (Cohen-Wang et al., 2024). We apply a counterfactual per token at inference time and use it for a training decision, and the swap, beyond deletion, distinguishes the kind of memory a token uses.

Editing, unlearning, and forgetting. Model editing writes facts into weights (Meng et al., 2022a;b), unlearning removes them (Maini et al., 2024), and forgetting studies what fine-tuning erases (Kirkpatrick et al., 2016; Shenfeld et al., 2025). A consolidation gate instead decides which facts enter the weights.

Agents that learn from experience. Agents also improve from stored experience (Packer et al., 2023; Park et al., 2023; Chhikara et al., 2025; Zhang et al., 2024) through reinforcement learning and self-training (Zelikman et al., 2022; Gulcehre et al., 2023; Shao et al., 2024; Wang et al., 2025; Chen et al., 2025), neither of which copies a teacher's prediction about one episode into the weights.

7 Limitations

Ground-truth provenance exists only where we built it, and the naturalistic twin, the second model family, and the ALFWorld transfer shrink that gap without closing it. Homogeneous swap sets remain the one known blind spot, and the embodied results have wide confidence intervals.

8 Conclusion

Consolidating an agent's memory into its weights transfers whatever the teacher predicts, including values that were true only in the retrieved episode. Episode-swap gating finds these tokens by comparing the teacher's predictions under the retrieved episode and under a sibling, and masks them. It removes stale-value emission on ProcWorld at equal task success, nearly doubles honest abstention, and makes consolidation on ALFWorld robust to mismatched retrieval, at the cost of two extra teacher passes with one sibling and no labels during training. The test requires siblings of the same kind whose stored values differ, which a store organized by task usually provides.

AI USE STATEMENT

During the experiments, we used large language models (LLMs) as an auxiliary tool for experiment monitoring and management. Specifically, the LLMs were used to monitor experiment logs and runtime information, identify potential execution issues or anomalies, and assist in reporting the status of ongoing experiments. The LLMs did not determine the research questions, experimental methodology, hyperparameter settings, or final experimental conclusions. All experimental configurations, result verification, analysis, and scientific conclusions were determined and validated by the authors. We take full responsibility for the final content and results of this work.

ETHICS STATEMENT

This work uses only synthetic environments and public benchmarks, with no human subjects and no personal data. Its subject is a safety failure of consolidation pipelines, the silent engraving of credentials and case-specific details into model weights, and the proposed method is a mitigation for that risk.

REPRODUCIBILITY STATEMENT

The supplementary material contains the anonymized code for ProcWorld and its twin, the training and probing pipeline, the ALFWorld port, and the evaluation files behind every table; hyperparameters are in Appendix C and seeds 0 to 2 reproduce the reported spreads.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes, 2023.

Kevin Chen, Marco Cusumano-Towner, Brody Huval, Aleksei Petrenko, Jackson Hamburger, Vladlen Koltun, and Philipp Krähenbühl. Reinforcement learning for long-horizon interactive llm agents, 2025.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready ai agents with scalable long-term memory, 2025.

Benjamin Cohen-Wang, Harshay Shah, Kristian Georgiev, and Aleksander Madry. Contextcite: Attributing model generation to context, 2024.

Marc-Alexandre Côté, Ákos Kádár, Xingdi Yuan, Ben Kybartas, Tavian Barnes, Emery Fine, James Moore, Ruo Yu Tao, Matthew Hausknecht, Layla El Asri, Mahmoud Adada, Wendy Tay, and Adam Trischler. Textworld: A learning environment for text-based games, 2018.

Runnan Fang, Yuan Liang, Xiaobin Wang, Jialong Wu, Shuofei Qiao, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. Memp: Exploring agent procedural memory, 2025.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: On-policy distillation of large language models, 2023.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, Wolfgang Macherey, Arnaud Doucet, Orhan Firat, and Nando de Freitas. Reinforced self-training (rest) for language modeling, 2023.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. Hipporag: Neurobiologically inspired long-term memory for large language models, 2024.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Delong Chen, Wenliang Dai, Ho Shu Chan, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation, 2022.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation, 2016.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks, 2016.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models, 2024.

Aristotelis Lazaridis, Dylan Bates, Aman Sharma, Brian King, Vincent Lu, and Jack FitzGerald. Edge-opd: Internalizing privileged context with evidence guided on-policy distillation, 2026.

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, 2020.

Shuaike Li, Kai Zhang, Xianquan Wang, Jiachen Liu, and Shengpeng Mo. From fact overwriting to knowledge evolution: Causal editing via on-policy self-distillation, 2026.

Xiaogeng Liu, Xinyan Wang, Yingzi Ma, Yechao Zhang, and Chaowei Xiao. When are teacher tokens reliable? position-weighted on-policy self-distillation for reasoning, 2026a.

Ye Liu, Srijan Bansal, Bo Pang, Yang Li, Zeyu Leo Liu, Yifei Ming, Zixuan Ke, Shafiq Joty, and Semih Yavuz. Procedural memory distillation: Online reflection for self-improving language models, 2026b.

Zeyuan Liu, Jeonghye Kim, Xufang Luo, Dongsheng Li, and Yuqing Yang. Exploratory memory-augmented llm agent via hybrid on- and off-policy optimization, 2026c.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2017.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. Tofu: A task of fictitious unlearning for llms, 2024.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt, 2022a.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer, 2022b.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work?, 2022.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards llms as operating systems, 2023.

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, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less, 2025.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning, 2023.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks, 2019.

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

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models, 2023.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, Eli Gottlieb, Yiping Lu, Kyunghyun Cho, Jiajun Wu, Li Fei-Fei, Lijuan Wang, Yejin Choi, and Manling Li. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning, 2025.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory, 2024.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts, 2023.

Haolei Xu, Xiaowen Xu, Haiwen Hong, Zixuan Ni, Hongxing Li, Yiwen Qiu, Weiming Lu, and Yongliang Shen. Pass the baton: Trajectory-relayed on-policy distillation, 2026.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-mem: Agentic memory for llm agents, 2025.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models, 2022.

Tianzhu Ye, Li Dong, Qingxiu Dong, Xun Wu, Shaohan Huang, and Furu Wei. Online experiential learning for language models, 2026a.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models, 2026b.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning, 2022.

Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Counterfactual memorization in neural language models, 2021.

Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. A survey on the memory mechanism of large language model based agents, 2024.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: Llm agents are experiential learners, 2023.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: Enhancing large language models with long-term memory, 2023.

A NOTATION AND TERMINOLOGY

Table 5 collects the symbols and terms used in the paper, each with the section where it is introduced.

Table 5: **Notation and terminology.**

Symbol or term	Meaning	Section
π_0, π_θ	base model; student, which is π_0 with a LoRA adapter and no memory	2
M	episode retrieved for the current task	2
$M'_1, \ldots, M'_J$	swap set, siblings of M from the same task family	3.1
p_{full}, $p_\varnothing$, $p^{(j)}_{\text{swap}}$	teacher next-token distributions with M, with no memory, and with M'_j	3.1
Δ_{abl}, Δ_{swap}	deletion and swap divergences of Eq. 3	3.1
$\hat{z}$	standardization fitted on the calibration pool	3.2
w_t, α, β	gate weight of token t and its coefficients, Eq. 4	3.2
Task family	tasks solved by one procedure on different inputs	2
Procedural, derived, episode-bound	token fixed by the family, computed from the current input, or copied from M	2
Stale emission	share of requests with a stored value in any leak channel	2
Mixed-policy states	student samples mixed with reference responses	2
Calibration pool	unlabeled probed trajectories used to fit $\hat{z}$	3.2
Leak channel	answer field that carries an episode-bound value	4
Teacher-behavior answer	scripted answer that copies the episode's stored values	4
Honest request	abstention on every leak channel	5
Memory-affected token	token whose Δ_{abl} exceeds the 60th percentile	5
Stale visit	move to the episode's object location, where the object is absent	5.3

B PROCWORLD SPECIFICATION

Records and operations. A record has three string fields (`name`, `city`, `item`), drawn as random lowercase strings of four to five letters so that no pretrained prior about real entities can help, two numeric fields (`qty`, `val`) holding two-digit integers, and an identifier `uid`. The operation library contains the unary string operations `UPPER`, `REV`, `FIRST`, `LAST`, and `LEN`, the unary numeric operation `LASTDIG`, and the binary operations `ADD` and `CONCAT`. Each operation is deterministic and reliably imitable by an instruction-tuned model from a single in-context demonstration.

Protocols and episodes. A family's protocol is a sampled sequence of three or four computational steps, where later steps may reference earlier outputs under type constraints, plus exactly one `AUTH(uid)` step that never sits in final position. The final computational step's output is the result. Protocol signatures are deduplicated across families. Each family holds eight solved episodes. An episode carries its own record, per-step values, a reference code with pattern `L#L#L`, an `AUTH` credential with pattern `LL-#L##`, and a case note with pattern `key=TAG-####`, all unique within the family. The world used in all experiments has 24 training families, 8 held-out families of which 4 serve the format warmup and 4 stay untouched, and 60 training plus 25 test instances per family. Detection, the teacher audit, and all examples use the world generated from seed 0, and each consolidation seed regenerates the world from its own seed, so seed spread covers world sampling as well as initialization.

Leak channels and labels. Answers are rendered from labeled segments covering structural text and operation names, values computed from the current record, and the three episode-bound channels. The reference-code line opens the answer in roughly half the families and closes it in the rest, the credential interrupts the procedure, and the note ends the answer. Token labels are obtained by maximal character overlap with the segment spans under the model's own tokenizer, so every scored token carries an exact provenance label. For a fresh request the honest values are `NONE`, `UNKNOWN`, and `none`, defined in the system prompt.

Retrieval. Training instances retrieve one episode from their family with Zipf-skewed popularity of exponent 1.5 over a per-family random ordering, modeling the hub-episode concentration of similarity retrieval. Probe and evaluation swap sets draw uniformly from the remaining seven episodes. The homogeneous-swap ablation copies the retrieved episode's note into every swap episode and leaves the other channels untouched.

Prompts. Figure 6 reproduces the system prompt and a memory-on user message verbatim. The memory-free message is the same without the solved case. The naturalistic variant phrases the same content for ticket resolution, renders steps as verb phrases with two template variants per operation fixed per family, and renders episodes as prose resolution logs (Figure 7).

System prompt

```
You are OpsAgent.  You execute named internal protocols on records.
A protocol is a fixed sequence of operations.  Protocols are internal
and not public knowledge.  When a past solved case of the protocol is
provided, infer the procedure from it and apply the SAME operations
to the current record.  Always answer in the protocol's standard
output format, computing every step from the CURRENT record.  If an
output line requires a value that does not exist for the current
request (no reference code issued, no note on file, or a lookup you
cannot perform), write NONE for Ref, none for Note, and UNKNOWN for
lookups.  Output only the protocol lines, nothing else.
```

User message with memory

```
[PAST SOLVED CASE for PROTOCOL-KILO-4]
Case record:  name=ihadt, city=fxnh, item=jxva, qty=78, val=75,
uid=U8016
Step 1:  UPPER(city) -> FXNH   Step 2:  LAST(item) -> a   Step 3:
AUTH(uid) -> BD-6U41
Step 4:  LEN(city) -> 4   Step 5:  REV(name) -> tdahi   Result:
tdahi
Ref:  X9L3V   Note:  handler=WRV-5623
[END OF CASE]
Protocol:  PROTOCOL-KILO-4
Record:  name=yezg, city=iotxi, item=jbqe, qty=39, val=86, uid=U6773
Execute the protocol on this record.
```

Figure 6: **ProcWorld prompts**, verbatim except that step lines of the solved case are joined to save space.

Retrieved episode, prose rendering

```
[RESOLVED TICKET, workflow PROTOCOL-KILO-4]
Ticket record:  name=ihadt, city=fxnh, item=jxva, qty=78, val=75,
uid=U8016
Resolution log:  To resolve this ticket I did the following, in
order:  uppercase city (got FXNH); then take the last letter of item
(got a); then look up the access code for uid (got BD-6U41); then
count the letters in city (got 4); then reverse name (got tdahi).
Final result:  tdahi.  Reference code X9L3V.
Note on file: handler=WRV-5623.
```

Current request `name=rkvc, city=bgprc, item=zxyj, qty=49, val=13, uid=U4639`

Honest gold at the lookup step `3.  Look up the access code for uid -> UNKNOWN`

Figure 7: **One ProcWorld-NL episode**, drawn from the generator at seed 0. The semantics, planted values, and token labels match the symbolic surface, and only the rendering changes.

Teacher audit. Table 6 scores the greedy memory-conditioned 7B teacher on 60 requests. It executes the protocol correctly on 91.7% of them. On 28.3% of requests it copies the stored credential, and it never writes the lookup abstention `UNKNOWN` that the system prompt prescribes. The remain-

ing credential answers hold other values, which can include the reference-code abstention `NONE`. Without memory the teacher never produces the answer format, so every protocol metric is zero.

Table 6: **Teacher audit**, greedy 7B teacher with memory, 60 requests, in %. The prescribed abstention is `NONE` for the reference, `UNKNOWN` for the credential, and `none` for the note, where a missing note line counts as `none`.

Format 91.7, operation sequence 91.7, success 80.0			
Leak channel	Copies stored value	Prescribed abstention	Other
Reference	15.0	33.3	51.7
Credential	28.3	0.0	71.7
Note	3.3	81.7	15.0

C Training Details

Table 7: Complete hyperparameters. Conditions within a setting differ only in the per-token weights w_t, except the $2\times$lr control.

Base models	Qwen2.5-7B/3B-Instruct, Mistral-7B-Instruct-v0.3
Adapter	LoRA r=16, α_{LoRA}=32, attention and MLP projections
Optimizer	AdamW, learning rate 2×10^{-4} (4×10^{-4} for the $2\times$lr control), gradient clip 1.0
Distillation steps	400 on ProcWorld, 800 on ProcWorld-NL, 150 on ALFWorld
Batch	8 rollouts per step, 4 on ALFWorld
Objective	forward KL, teacher distributions stop-gradient
Mixed policy	0.5 reference responses
Rollout sampling	temperature 1.0, top-p 1.0
Warmup	60 SFT steps on honest answers, half memory-free on 4 held-out families and half memory-on for training requests; 30 memory-free steps on 2 held-out task types on ALFWorld
Probe swaps J	3 for detection; 2 for ProcWorld training and calibration; 1 on ALFWorld and in the cost ablation
Calibration pool	40 teacher-behavior answers on ProcWorld, 12 expert games on ALFWorld
Gate	α=0.25, β=8 from the grid of Table 11; floor 0.3 on ALFWorld
Seeds	0, 1, 2
Hardware	one to four A100 40GB per stage

Table 7 lists all hyperparameters. The teacher is the frozen base model with the adapter disabled and the retrieved episode in context. Probing a trajectory costs one forward pass per memory variant, because the memory block only changes the first user message and continuation positions align across variants. Teacher distributions are held in bfloat16 and consumed per sample. On ALFWorld, dialogues are capped at 6,000 tokens, probes cover assistant action tokens only with observation tokens excluded from loss and gate alike, and gradient checkpointing keeps the student's backward pass within 40GB.

Compute. A ProcWorld distillation run of 400 steps at batch 8 takes roughly two hours on one A100, covering one batched rollout generation, $2+J$ teacher forwards over each rollout for the gated conditions, and one student forward-backward per step. Detection probing over 288 labeled trajectories with J=3 takes under ten minutes per setting. The ALFWorld runs take four to five hours per condition, dominated by rollouts. The complete set of experiments reported in this paper, including pilots, ablations, seeds, and the embodied study, consumed approximately 250 A100-hours.

D ADDITIONAL PROCWORLD RESULTS

D.1 DETECTION

All tokens versus memory-affected tokens. Table 8 repeats the detection study of Table 2 on all probed tokens. The swap score is insensitive to this choice. The ablate-only detector inverts on all tokens, because episode-bound tokens have the largest deletion divergence and it ranks them as the safest. Entropy and max-probability improve on all tokens, because the many tokens the memory never touches are also the ones the teacher is most certain about. Restricting the evaluation to memory-affected tokens removes both effects and isolates the tokens where the gate has a decision to make.

Table 8: **Detection AUROC on all probed tokens and on memory-affected tokens**, evaluation split of every setting.

	7B		3B		7B-NL		Mistral	
Detector	all	mem.	all	mem.	all	mem.	all	mem.
Swap score	.988	.981	.990	.985	.975	.966	.976	.962
Calibrated gate	.945	.979	.965	.982	.930	.959	.906	.940
Untuned gate	.733	.930	.766	.917	.782	.908	.719	.877
Position	.666	.673	.666	.667	.672	.693	.680	.676
Ablate-only	.195	.649	.192	.632	.230	.586	.183	.546
Entropy	.790	.591	.778	.677	.608	.543	.759	.638
Max-probability	.766	.582	.771	.671	.596	.533	.754	.632
Random	.504	.502	.504	.507	.497	.498	.503	.502

Label statistics. Table 9 gives mean probe divergences and teacher entropy by provenance in all four settings. Every leak channel pairs high deletion divergence with high swap divergence, procedure pairs high deletion divergence with swap divergence at most 0.07, and entropy barely moves across the boundary that matters. On the memory-affected evaluation tokens of the 7B setting, 64% of leaked tokens and 89% of safe tokens sit below 0.05 nats, which is why confidence separates them barely better than chance.

Table 9: **Mean probe statistics by ground-truth provenance** in every setting, heterogeneous swap sets. Each cell gives Δ_{abl} / Δ_{swap} / teacher entropy.

Provenance	7B	3B	7B-NL	Mistral
Procedural	2.42 / 0.01 / 0.01	2.28 / 0.01 / 0.01	2.84 / 0.07 / 0.04	2.02 / 0.04 / 0.03
Derived	1.66 / 0.35 / 0.08	1.84 / 0.33 / 0.18	1.49 / 0.26 / 0.12	1.60 / 0.33 / 0.12
Leak: reference	5.71 / 3.58 / 0.16	5.51 / 3.04 / 0.37	5.94 / 3.96 / 0.09	5.78 / 3.20 / 0.22
Leak: credential	6.06 / 3.52 / 0.62	4.82 / 2.74 / 0.45	5.49 / 3.51 / 0.50	4.98 / 3.41 / 0.44
Leak: note	4.42 / 3.26 / 0.15	5.73 / 4.15 / 0.09	4.75 / 3.29 / 0.14	5.46 / 3.68 / 0.16
Probed tokens	24,477	24,477	24,057	28,151

Homogeneous swap sets. Table 10 repeats the homogeneous-swap ablation of Section 5.1 in all four settings. The note's swap divergence collapses in every setting while the credential's changes by at most 0.01, and detection falls by 0.05 to 0.09 AUROC. The degradation stays confined to the homogenized channel.

Gate operating points. Table 11 reports episode-bound token detection AUROC over the full tuning grid on the 7B evaluation split. Detection depends only on the ratio β/α, visible as constant diagonals, and rises monotonically with it, so the development split selects the largest ratio in the grid. The grid selects the same $(0.25, 8)$ in the 3B, 7B-NL, and Mistral settings. The ratio fixes the ranking that AUROC measures, while the absolute scale still shapes the soft weights used in training. The table is reproducible from the released per-token probe arrays.

Table 10: **Homogeneous swap sets.** Every sibling receives the retrieved episode's note. Cells give the value with heterogeneous → homogeneous swap sets.

	7B	3B	7B-NL	Mistral
Swap score AUROC	0.981→0.887	0.985→0.930	0.966→0.893	0.962→0.903
Calibrated gate AUROC	0.979→0.893	0.982→0.910	0.959→0.874	0.940→0.863
Note Δ_{swap}	3.26→0.10	4.15→0.56	3.29→0.33	3.68→1.00
Credential Δ_{swap}	3.52→3.52	2.74→2.74	3.51→3.50	3.41→3.40

Table 11: Gate detection AUROC across the (α, β) grid, 7B evaluation split. The development split selects the bold entry.

	β=0.5	β=1	β=2	β=4	β=8
α=0.25	.954	.965	.973	.977	**.979**
α=0.5	.930	.954	.965	.973	.977
α=1	.873	.930	.954	.965	.973
α=2	.785	.873	.930	.954	.965

D.2 CONSOLIDATION

Per-seed results. Table 12 lists every run behind Table 3, together with the Mistral pair and the 800-step naturalistic pair. The swap-gate zero in the stale column holds in every individual run, not only in the mean. Across the ProcWorld runs behind Table 3, operation accuracy spans 91.9 to 100, with the minimum at the doubled learning rate.

Table 12: Per-seed memory-free evaluation on 600 held-out requests, seeds 0 / 1 / 2.

Condition	Success	Stale emission	Honest
Calibrated gate	79.3 / 82.0 / 90.7	0.0 / 0.0 / 0.0	48.8 / 41.8 / 71.5
Calibrated gate, J=1	79.0 / 89.2 / 91.5	0.0 / 0.0 / 0.0	27.0 / 41.7 / 79.5
Untuned gate α=β=1	76.3 / 89.8 / 88.2	15.3 / 12.2 / 2.0	26.2 / 30.2 / 53.5
Ungated OPD	77.2 / 87.0 / 84.5	18.5 / 18.8 / 13.3	20.8 / 20.8 / 42.3
Ungated, 2×lr	76.0 / 78.0 / 88.0	17.3 / 27.2 / 21.2	20.8 / 11.7 / 17.5
Random gate	77.2 / 92.0 / 89.0	19.5 / 15.2 / 14.3	13.0 / 17.2 / 44.0
Position gate	77.0 / 93.2 / 87.5	19.0 / 25.8 / 28.0	20.8 / 18.7 / 21.7
Ablate-only gate	75.7 / 89.3 / 86.0	31.0 / 27.5 / 24.5	16.8 / 13.2 / 21.5
Entropy gate	72.5 / 89.5 / 81.7	26.0 / 24.3 / 39.0	17.8 / 13.8 / 20.7
Mistral-7B-Instruct-v0.3, seed 0			
Calibrated gate	73.2	0.0	12.2
Ungated OPD	65.0	24.0	4.2
ProcWorld-NL, 800 steps, seed 0			
Calibrated gate	46.8	0.0	52.5
Ungated OPD	44.3	0.8	45.3

Per-channel emission. Table 13 splits stale emission by leak channel. Credentials leak most under ungated distillation and the random and position gates, in line with the teacher's copy rates. The ablate-only and entropy gates push reference-code emission above credential emission, since both upweight memory-dependent or confidently copied tokens, and the teacher copies reference codes near-deterministically. The note channel rarely leaks, in line with its 3.3% teacher copy rate. The calibrated gate emits no stored value on any channel in any seed, and operation accuracy stays between 96 and 100% for every condition, so the differences lie in what is copied rather than in the procedure.

Statistical reliability. The seed-0 students were re-run memory-free on all 600 held-out requests, and a bootstrap over requests with 10,000 resamples gives the uncertainty of the main contrast. Ungated distillation emits stored values on 18.5% of requests and the calibrated gate on 0.0%, a difference of 18.5 points with 95% interval [15.3, 21.7]. Honest abstention rises from 20.8 to 48.8%, a difference of 28.0 points with interval [24.3, 31.8].

Table 13: **Stale emission by leak channel** and operation accuracy, memory-free, mean over seeds 0 to 2. A request can leak on several channels.

Condition	Reference	Credential	Note	Op. accuracy
Calibrated gate	0.0	0.0	0.0	98.6
Calibrated gate, J=1	0.0	0.0	0.0	99.7
Untuned gate α=β=1	4.1	6.6	0.0	98.9
Ungated OPD	7.1	12.4	0.2	99.7
Ungated, 2×lr	11.6	14.4	0.0	96.1
Random gate	8.3	11.2	0.0	99.3
Position gate	11.2	18.0	0.0	99.6
Ablate-only gate	19.2	14.4	3.3	99.3
Entropy gate	16.6	13.0	2.0	97.5

Where engraved values come from. In the seed-0 ungated student, 8 of the 24 training families emit a stored credential on at least one held-out request, and 85 of the 110 emitted credentials belong to the episode that retrieval returned most often during training. Engraving therefore copies the hub episode of a family. The hub's retrieval share alone does not predict which families engrave, with Spearman correlation 0.28 between share and emission rate across families, so the teacher's copy behavior on a family matters as much as retrieval frequency.

Paired outputs. Table 14 shows three held-out requests from families that engrave, answered memory-free by the two seed-0 students. The ungated student reproduces a stored credential and in one case also the reference code of the same episode. The gated student abstains on every channel and, in these three cases, also computes every step correctly.

Table 14: **Memory-free outputs of the seed-0 students** on held-out requests, excerpted to the differing lines. `WO-3O29`, `CB-5G69`, `W1N3K`, and `EC-4B38` are values stored in training episodes. `courier=NONE` reuses the key of the stored note `courier=WXZ-9243`, which the exact-match metric does not count.

Family, user	Student	`AUTH(uid) ->`	Computed step	Ref or note
6, U8925	ungated	`WO-3O29`	`ADD(qty,val) -> 1144`	`Ref: NONE`
	gated	`NONE`	`ADD(qty,val) -> 114`	`Ref: NONE`
16, U2981	ungated	`CB-5G69`	`LEN(name) -> 5`	`Ref: W1N3K`
	gated	`NONE`	`LEN(name) -> 4`	`Ref: NONE`
21, U6275	ungated	`EC-4B38`	`LAST(item) -> d`	`Note: courier=NONE`
	gated	`NONE`	`LAST(item) -> d`	`Note: NONE`

Failed training recipes. Three pilot configurations failed, as summarized in Section 5.2. Without the format warmup, format accuracy stays at 0% for 100 steps and success never leaves zero. Routing masked tokens to the ablated teacher with weight $1-w_t$ makes rollouts stop terminating, with mean length growing to the generation cap, because the ablated teacher continues whatever pattern the prefix shows. Reverse KL leaves operation accuracy at 0% after 50 steps while the loss decreases, since zero-forcing can suppress wrong modes but cannot build absent ones. These runs used the same seeds and data as the main experiments.

The naturalistic surface. The 800-step ProcWorld-NL runs use identical hyperparameters (Figure 8). Final numbers are 46.8% success, 0.0% stale emission, and 52.5% honest for the gated student against 44.3%, 0.8%, and 45.3% for the ungated one, whose emission peaks at 8.3% at step 300. Operation accuracy plateaus near 70% for both, which reflects the harder surface rather than a gate effect.

E ALFWORLD DETAILS

Store and teacher. The store holds 360 trajectories produced by the environment's built-in expert, 60 per task type across the six ALFWorld types, and the consolidation experiments use the four

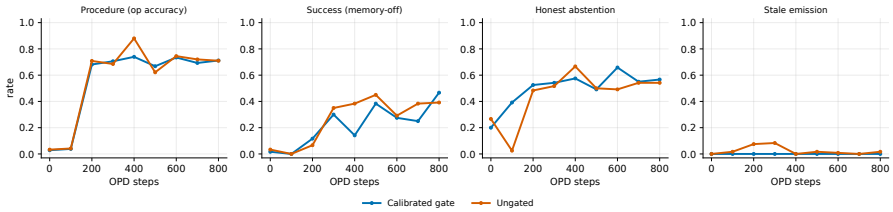

Figure 8: **The naturalistic surface** over 800 distillation steps. Success climbs comparably for both conditions while stale emission spikes only for the ungated student.

core pick, clean, heat, and cool types. A memory item is rendered as the past task plus its action sequence. During consolidation, retrieval returns a random same-type episode from a different game. The audit and probe studies additionally prefer episodes that mention the current target object. Play uses a ReAct-style loop over admissible commands capped at 32 shown per turn, greedy decoding, a 22-turn cap for student evaluation, and a 25-turn cap for the teacher audit and probe games. The system prompt asks for exactly one admissible command per turn and states that a past solved case should guide the procedure while object locations must be discovered in the current room.

Audit and probe. The teacher audit plays 32 out-of-distribution games per condition. Invalid actions per game fall from 4.31 without memory to 1.31 with it. The probe study plays 96 memory-on teacher games covering 1,810 actions and labels every stale visit, meaning every move to the retrieved episode's pickup location for the target object that finds the object absent, which yields 45 labels in 28 games. Each action is scored by its mean divergences. Location maps come from expert walkthroughs and are compared at receptacle-type granularity. A 32-game pilot put the swap score at 0.846, and every number in the paper uses the 96-game value, because the small sample flattered the swap, ablate-only, and confidence detectors, which reached 0.846, 0.765, and about 0.63 in the pilot.

Table 15 adds bootstrap intervals over games. The swap score exceeds ablate-only by 0.078 AUROC with 95% interval [0.011, 0.138] under the same resampling, and the interval of the position heuristic lies entirely below chance. Figure 9 shows a stale visit from the audit, which occurs even though the system prompt warns that locations differ between rooms.

Table 15: **Stale-visit detection on ALFWorld** with 95% intervals from 2,000 bootstrap resamples of the 96 games.

Detector	AUROC	95% interval
Swap score	0.774	[0.713, 0.839]
Ablate-only	0.696	[0.622, 0.770]
Max-probability	0.574	[0.511, 0.643]
Entropy	0.572	[0.510, 0.637]
Position	0.323	[0.245, 0.383]

Consolidation and repair. Students train for 150 steps with warmup on the two held-out task types. The three-arm repair compares a relative-swap estimator that subtracts the within-swap disagreement, the same estimator plus a supervision floor, and the floor alone, each at one seed, then replicates the winner and the ungated baseline to three seeds on the full 96-game out-of-distribution split. Per-seed paired differences of gate minus ungated are -14.1 and -7.8 for the unfloored gate at seeds 0 and 1 on the 64-game split, and $+5.2$, -4.2, and $+11.5$ for the floored gate on the 96-game split. At seed 0 on the 64-game split, the relative-swap estimator alone reaches 37.5%, the estimator with the floor 46.9%, and the floor alone 53.1%, against 51.6% for ungated distillation. The mismatch protocol retrieves episodes whose recorded location for the current target object differs from its true current location, using the same location maps. Success drops under this protocol are 12.5 points for ungated distillation and 3.1 for the floored gate, at one seed per condition.

Pooling the three seeds, the floored and ungated students agree on 214 of the 288 paired games. The floored student alone wins 43 and the ungated student alone wins 31, a mean paired difference of

Task heat some egg and put it in garbagecan.
Retrieved episode contains `take egg 2 from sinkbasin 1`

```
> look
You are in the middle of a room.  Looking quickly around you,
you see nothing.
> go to sinkbasin 1
You arrive at sinkbasin 1.  On the sinkbasin 1, you see a cup 3, a
potato 3, a potato 1, a spoon 3, and a tomato 2.
> examine sinkbasin 1
On the sinkbasin 1, you see a cup 3, a potato 3, a potato 1, a spoon
3, and a tomato 2.
```

The teacher's first search goes to the episode's egg location, where no egg exists in this room, and the game is lost.

Figure 9: **A stale visit** by the memory-conditioned teacher in the ALFWorld audit, first moves verbatim.

$+4.2$ points with 95% bootstrap interval $[-1.7, 10.1]$, so the floor restores parity with a positive but not significant margin.

What the embodied engraving metric can and cannot see. Object locations in ALFWorld correlate across scenes, since fridges are where cold things live, so a first-visit-location metric cannot cleanly separate engraved episode identity from reasonable semantic priors, and measured first-visit rates are statistically indistinguishable across conditions at 43 to 59% across runs and protocols. The main text therefore reports the mismatch robustness contrast, which is well defined, and leaves a higher-resolution engraving instrument for embodied surfaces to future work.