GenMem

Generative Symbolic Memory for Self-Evolving Harness

Paper: Xinke Jiang, Tao Feng, Weixuan Xu, et al.

Independent paper-reading presentation

arXiv:2609.34633v1 / September 28, 2026

Reading date: October 19, 2026

Learn stable addresses; revise the experience behind them.

A revised memory should remain reachable

The useful object is a procedure

An agent reuses applicability rules, action sequences, verification steps, and lessons from failures.

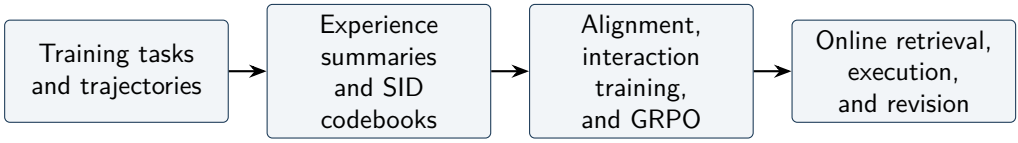
The procedure keeps changing

Re-embedding revised text can change which queries retrieve an entry, even when its role is unchanged.

- Relevant experience is sparse and often shares subskills.
- Delayed task success gives weak credit to individual memory operations.
- GenMem learns a context-to-address policy and edits addressed text.

Paper, Sec. 1, PDF pp. 1–3; Appendix B.1–B.2, pp. 15–16.

Construction, learning, and evolution are separate



- Cold start: contrast successful, failed, and unverified attempts.
- Training: learn where to read/write and how to use experience.
- Online: freeze models and codebooks; update the external bank.

Teaching diagram based on Fig. 2 and Secs. 2–3, PDF pp. 3–6.

80 symbols provide 49,152 possible addresses

$$\mathbf{z} = (s^1, \ldots, s^L) \in [N_1] \times \cdots \times [N_L]$$

Address capacity	SID vocabulary
$K_{\text{addr}} = \prod_{\ell=1}^{L} N_\ell$	$V_{\text{SID}} = \sum_{\ell=1}^{L} N_\ell$
$48 \times 16 \times 8 \times 8 = 49{,}152$	$48 + 16 + 8 + 8 = 80$

Each address uses four level-specific tokens. Each occupied slot holds one current experience text.

Capacity, populated slots, and payload storage are different quantities.

Paper, Eq. (1), PDF p. 3; Sec. 2, p. 4; Appendix C.1, p. 19.

Residual quantization constructs the initial SIDs

$$e_i = f_{\mathrm{enc}}(p_{\mathrm{enc}} \oplus x_i) \in \mathbb{R}^d, \qquad r_i^{(1)} = e_i$$

$$s_i^{\ell} = \arg\min_{j \in [N_\ell]} \|r_i^{(\ell)} - c_j^{(\ell)}\|_2^2,$$

$$r_i^{(\ell+1)} = r_i^{(\ell)} - c_{s_i^{\ell}}^{(\ell)}, \qquad \hat{e}_i = \sum_{\ell} c_{s_i^{\ell}}^{(\ell)}.$$

- A frozen encoder represents reusable procedures in summary x_i.
- Each level selects a centroid and passes the residual onward.
- Experiences sharing a complete SID are consolidated.

Levels have no predefined domain/tool/strategy semantics.

Paper, Eqs. (2)–(3), p. 4; Appendices B.4–B.5 and C.1, pp. 17–20.
Equations show the Euclidean form; Table 15 selects a Weighted setting.

One address space supports both reading and writing

1. MemRetriever	2. StackPlanner	3. MemEvolver
Query/context	Query + support	Trajectory
↓	↓	↓
Generate SID	Execute task	Generate target SID
↓	↓	↓
Lookup and rewrite	Answer + trajectory	Insert or revise

$$p_\theta(\mathbf{z} \mid c) = \prod_{\ell=1}^{L} p_\theta(s^\ell \mid c, s^{1:\ell-1})$$

The write address is generated by MemEvolver. Updated text is not re-quantized to choose a replacement address.

Paper, Eq. (4), p. 4; Sec. 3, pp. 5–6; Appendix B.3, p. 17.

Revision preserves the address and changes the rule

$$\mathcal{M}_{t+1}[z] = \begin{cases} \tilde{x}_{e,t}, & z = z_{e,t}, \\ \mathcal{M}_t[z], & z \neq z_{e,t}. \end{cases}$$

Example: fixed members in a selection task

Same SID $(3, 8, 4, 2)$:
"Use combinations for unordered selection"
↓
"First remove fixed members; then choose the remaining members."

For 12 players, 5 starters, and 2 fixed starters: $\binom{10}{3} = 120$.

`revise` can also retain, merge, or clear the payload.

Teaching paraphrase of Appendix H, PDF p. 40. Update rule: Appendix C.2, pp. 20–21.

Alignment learns IDs; mid-training learns interactions

Five alignment mappings

$$q \to z, \quad \tau \to z, \quad x \to z$$

$$z \to x, \qquad z \to d$$

q: query; τ: trajectory; x: experience; d: semantic description.

Role-specific demonstrations

MemRetriever: reason, address, lookup, rewrite.

MemEvolver: inspect a trajectory, address, compare, update.

$$\mathcal{L}_{\mathrm{SID}}(c, z) = -\sum_{\ell=1}^{L} \log p_\theta(s^\ell \mid c, s^{1:\ell-1})$$

Mid-training uses filtered ReAct-style traces and stronger-teacher targets.

Paper, Eqs. (5)–(6) and Sec. 3, PDF pp. 4–5; Appendix B.5, p. 18.

Rewards evaluate both the operation and its utility

$$R_{\text{process}} = R_{\text{format}} + R_{\text{strategy}} + R_{\text{SID}}$$

$$R_{\text{out}} = r_{\text{mem}} - r_{\text{no-mem}} + \alpha_{\text{partial}}\,\mathbb{I}[\text{both succeed}]$$

With memory	Without memory	Reward	Interpretation
Success	Failure	$+1$	Helpful memory
Failure	Success	-1	Harmful memory
Success	Success	α_{partial}	Preserved success
Failure	Failure	0	No success gain

The table assumes binary task rewards. Strategy judging uses privileged reference information during training.

Paper, Eqs. (7)–(8), PDF p. 5; Appendix C.3, pp. 21–22.

Memory-policy RL and execution-policy RL differ

- **Memory-policy GRPO:** sample multiple outputs per context; prefer higher-reward operations while clipping updates and penalizing KL drift.
- **Execution-policy Harness-RL:** train StackPlanner's task policy within its harness.
- **Online evolution:** freeze all policies and codebooks; update only the external experience bank.

How to read the main result rows

GenMem and GenMem (RL) distinguish the configurations before and after **StackPlanner Harness-RL**. The former still uses trained memory modules.

Paper, Eq. (9), p. 5; Sec. 4.1, pp. 6–7; Appendix F.1, p. 28.

Four evaluations answer different questions

Evaluation	Data / setting	Metric
Main tasks	ALFWorld; WebShop; six search QA sets	Environment success; task score; answer token F1
Transfer	Three harnesses under DeepSeek-V4-Flash	Search EM; Research RACE; SQL execution; code resolution
Retrieval / evolution	Up to 50 candidates; BGE reranking	Candidate Hit; Hit@1/5; task accuracy
Symbolic input	GPQA, beam sizes 1/5/10	Answer accuracy

Main GenMem modules: Qwen2.5-7B-Instruct.
QA average: mean of six dataset-level F1 scores.
Hit@k: presence of one reference SID, not precision of the returned set.

Paper, Sec. 4.1, p. 6; Appendices E–F, pp. 25–28.

The full system improves interactive task results

Method	ALFWorld All	WebShop Score	WebShop Success
ReAct	31.2	46.2	19.5
GenMem	74.6	52.0	27.0
GRPO	77.6	79.3	66.1
SkillRL	89.9	85.2	72.7
Skill0	89.8	85.3	71.9
GenMem (RL)	**93.6**	**90.8**	**82.8**

Compared with SkillRL

ALFWorld: +3.7 percentage points WebShop success: +10.1 pp

ALFWorld's Pick, Heat, and Cool have stronger alternatives in the table.

Selected rows from Paper, Table 1, PDF p. 7. All values use the paper's percentage scale.

QA gains depend strongly on the trained task policy

Method	2Wiki	Hotpot	Bambo.	MuSiQue	NQ	Trivia	Avg.
ReAct	27.51	42.81	27.63	19.34	30.01	54.55	33.6
GenMem	42.91	35.83	22.21	19.85	33.88	47.62	33.7
SkillRL	40.3	43.2	73.8	20.2	45.9	63.3	47.8
SDAR	48.4	43.8	73.0	19.6	46.3	63.5	49.1
GenMem (RL)	**54.8**	**52.9**	64.6	**23.5**	**50.1**	**70.7**	**52.8**

- GenMem (RL) exceeds SDAR by 3.7 pp in average token F1.
- Before StackPlanner RL, GenMem is near ReAct on average and lower on three QA datasets.
- Bamboogle remains below SkillRL, even with the full system.

Selected rows from Paper, Table 2, PDF p. 8. Answer token F1 (%); higher is better.

Transferred skills help Research and SQL most consistently

Harness	Search	Research	SQL	Code
StackPlanner	0.00	+2.07	+5.00	+1.00
TCRAG	−1.00	+2.04	+1.00	+1.00
OpenHands	+1.00	+2.12	+2.00	+2.00

- Absolute score changes from skill-free to +GenMem Skills.
- All three harnesses use DeepSeek-V4-Flash in this comparison.
- Search changes are small and include one decrease.

Scope

Evidence for transferred skill support; the table does not by itself establish an identical online read–write loop in every harness.

Paper, Table 3, p. 8; metrics in Table 10, p. 26. Search: EM; Research: RACE; SQL: execution accuracy; Code: resolved rate.

Online evaluation scores a task before writing memory

1. Start with the initial memory bank and frozen model parameters/codebooks.
2. Retrieve support, execute the incoming task, and score its answer.
3. Give the trajectory to MemEvolver; apply its update before the next task.
4. After each block, evaluate a fixed, disjoint probe set with read-only access.

Information boundary

Held-out reference answers are used for scoring and are excluded from the update prompt.

Content-based baselines rebuild their indices from the current payloads at each checkpoint.

Paper, Sec. 4.3, PDF pp. 7–8; Appendix E.3, p. 27.

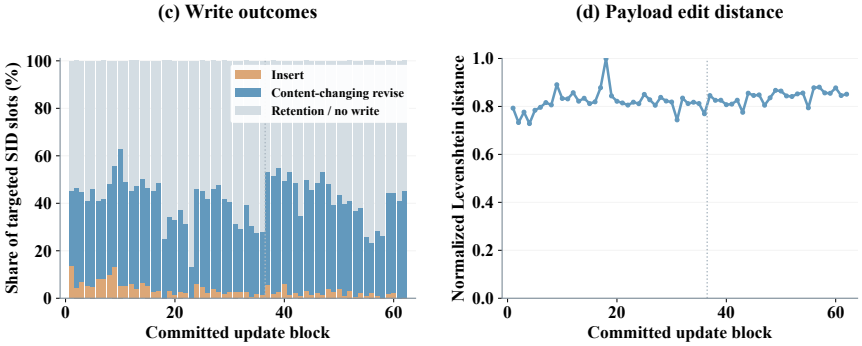
GenMem benefits most from the reported evolution stream

Method	Hit@5, $t = 0 \rightarrow 9$	Task ACC, $t = 0 \rightarrow 9$	ACC change
Qwen3-Embedding	$2.89 \rightarrow 2.52$	$28.53 \rightarrow 27.98$	-0.55
TF-IDF	$12.52 \rightarrow 13.36$	$30.02 \rightarrow 31.08$	$+1.06$
SkillRouter	$10.28 \rightarrow 9.72$	$28.74 \rightarrow 28.95$	$+0.21$
GenMem	$\mathbf{13.15 \rightarrow 17.63}$	$\mathbf{30.56 \rightarrow 36.20}$	$\mathbf{+5.64}$

- GenMem Candidate Hit: $20.25\% \rightarrow 25.58\%$.
- Gains are not monotonic: checkpoints 2, 5, and 8 dip.
- TF-IDF exceeds GenMem on Hit@5 and task ACC at checkpoints 2 and 5.

Paper, Table 4, PDF p. 9. Values are percentages; changes are percentage points.

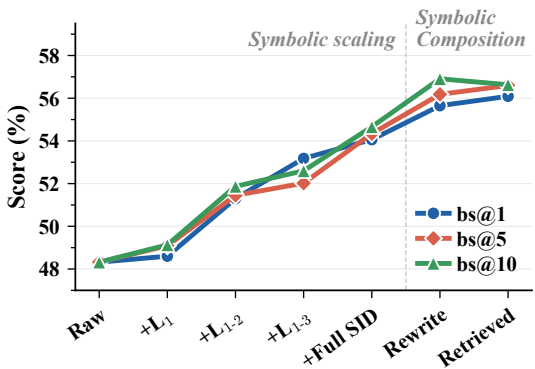
Evolution retains many slots and substantially edits others



- Left: targeted slots split into insert, content-changing revise, and retention.
- Right: payload edit distances remain substantial across update blocks.

Paper, Fig. 3(c,d), PDF p. 9. Original panels, cropped with axes and legends preserved.

SID tokens carry useful task information on GPQA



- Raw query: **48.32%**.
- Full SID, no payload: **54.06–54.66%**.
- Rewrite: 55.65–56.91%.
- Retrieved text: 56.09–56.63%.

Rewrite is better only at $k = 10$.

Unseen SID-combination generalization remains untested.

Paper, Fig. 4 and Sec. 5, PDF p. 9; Appendix G.5, pp. 37–38. Beam sizes: 1, 5, 10. Both text conditions read the bank.

Evolution, policy learning, and search all contribute

Variant	Accuracy	Token F1	ACC drop
Full GenMem	**49.5**	**52.1**	—
Frozen bank	46.5	46.8	3.0
TF-IDF + evolution	40.5	43.4	9.0
Embedding + evolution	37.0	41.2	12.5
No memory access	41.0	45.2	8.5
No memory-policy RL	38.5	42.6	11.0
Greedy SID decoding	42.0	45.7	7.5

Protocol: 200 samples, three complete evolution rounds.

Frozen bank isolates the benefit of updates within this setup.

Embedding + evolution falling below no memory illustrates harmful retrieval.

Paper, Table 23, PDF p. 37. Two mislabeled table rows are described here by their actual retrieval replacements.

Mid-training lowers exact SID retrieval in the appendix

Configuration	Mean candidates	R@50	HR@5
PT, four levels	49.57	20.25	13.15
MT, four levels, full beam	33.72	4.90	3.10
MT, four levels, SID-only beam	50.00	6.80	3.75
MT, four levels, MemEvolver	50.00	5.31	1.36

- PT: alignment pretraining. MT: memory-operation mid-training.
- Learning interaction traces does not automatically preserve address accuracy.
- The appendix lacks a matched final-RL retrieval row that closes this gap.

Paper, Appendix G.2 and Table 18, PDF pp. 30–31. MemRetriever: 2,000 examples; MemEvolver: 735. Samples are drawn from training examples.

Three evidence gaps limit the strongest claims

1. **Domain separation.** Several QA datasets labeled OOD also supply seed-bank experiences. Exact splits and overlap checks remain unspecified.
2. **Attribution.** Main tables compare systems with incompletely documented per-row budgets and provenance. Stable IDs are also possible in vector stores.
3. **Symbolic generalization.** Prefix depth helps GPQA, but fixed semantic axes and unseen-combination transfer need additional controls.

Paper, Table 6, p. 25; Appendices E.1 and F.1, pp. 26, 28; B.2, p. 16; G.5, pp. 37–38. Interpretation synthesized for this reading.

The reusable idea is a stable learned memory interface

Design contribution

Generate compact structured addresses for both retrieval and update targeting; keep experience editable behind those addresses.

- The full trained system reports strong interactive and QA results.
- Address invariance is guaranteed by the update rule; payload usefulness still depends on learned curation.
- The next decisive evidence is a matched comparison with controlled data splits, checkpoints, and stable-address retrieval baselines.

Jiang et al., GenMem: Generative Symbolic Memory for Self-Evolving Harness, arXiv:2609.34633v1.
`https://arxiv.org/abs/2609.34633v1` End of main presentation; backup slides follow.

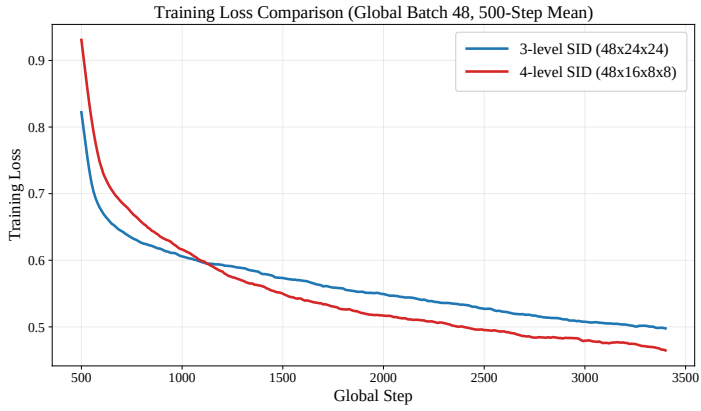
Backup: four levels trade quantization error for vocabulary

Codebook	Capacity	SID vocab	Loss (10^{-4})	Leaf use (%)
$48 \times 16 \times 8 \times 8$	49,152	80	3.48	62.22
$128 \times 24 \times 16$	49,152	168	3.31	64.25
$96 \times 24 \times 24$	55,296	144	3.34	56.71

- Selected four-level setting uses fewer special tokens.
- It does not minimize reconstruction loss among these alternatives.
- The prefix-prompt ablation uses a separate three-level setting: loss improves while leaf occupancy slightly falls.

Paper, Tables 12 and 15, PDF pp. 29–30. Weighted setting, $w = 0.5$.

Backup: four-level alignment loss is lower late in training



Red begins above blue and crosses around step 1,100.
Depth, capacity, and per-level vocabulary all change together.

Paper, Fig. 7, PDF p. 34. Original figure. Global batch 48; 500-step moving average.

Backup: prefix matches do not imply a correct full address

MT policy	L_1	$L_{1:2}$	$L_{1:3}$	$L_{1:4}$
MemRetriever, SID-only	40.90	13.00	6.30	3.75
MemEvolver	40.41	11.29	3.27	1.36

- Cumulative Top-5 hit after reranking: one candidate must match the whole prefix.
- Independent level accuracy can count a match even when all other levels are wrong.
- Beam width 5 to 20 raises Recall@5 from 7.93% to 8.53%, with reported decoding latency from 121.0 to 406.1 ms.

Paper, Tables 19 and 21, PDF pp. 32–33. Latency refers to the SID decoding experiment.

Backup: GRPO optimizes group-relative memory utility

$$\mathcal{L} = -\mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|u_i|}\sum_{t=1}^{|u_i|}\Big\{\min(\rho_{i,t}\hat{A}_i, \text{clip}(\rho_{i,t}, 1-\epsilon, 1+\epsilon)\hat{A}_i) - \beta D_{\text{KL}}^{i,t}\Big\}\right].$$

- u_i: sampled output; G: number of outputs per context.
- $\rho_{i,t}$: token probability ratio against the rollout policy.
- $\hat{A}_i$: group-normalized reward; ϵ: clipping threshold.
- KL regularization anchors the updated policy to a reference.

Proxy/audit advantages are normalized separately; complete channel weighting and reward scaling are not resolved in the manuscript.

Paper, Eq. (9), PDF p. 5; Appendix C.3, pp. 21–22.

Backup: reported settings need checkpoint-level reconciliation

Setting	Main text, pp. 6–7	Appendix F.2, p. 28
Alignment learning rate	1×10^{-5}	2×10^{-5}
Memory GRPO steps	130	200

- Appendix also lists GRPO LR 3×10^{-5}, batch 256, group size 8, and KL coefficient 0.04.
- Main text includes R_{SID} explicitly; some appendix reward decompositions omit it.
- Reported 16,416 final summaries do not match the approximately 30.6K denominator implied by several bank statistics.

Paper, Sec. 4.1; Tables 8–9 and 11; Appendix C.3, pp. 21–22. The implied denominator is a calculation in this reading.

Backup: correct task outcomes can yield overgeneralized memory

The payment case in Appendix H

The answer is correct, but the inserted memory says to interpret "double" or "half" using the immediately preceding payment.

- That rule fails when the question refers to an earlier payment or the initial amount.
- A fixed SID preserves access to the rule even if the rule becomes too broad.
- This motivates revision checks that preserve applicability conditions and test previously served queries.

Paper, Appendix H, PDF p. 41. Failure analysis and proposed check are this presentation's interpretation.

Backup: source and reading materials

Xinke Jiang, Tao Feng, Weixuan Xu, Zhixin Zhang, Zhibang Yang, Wentao Zhang, Runchuan Zhu, Xu Chu, Junfeng Zhao, Yasha Wang.

GenMem: Generative Symbolic Memory for Self-Evolving Harness.

arXiv:2609.34633v1, submitted September 28, 2026.
`https://arxiv.org/abs/2609.34633v1`

- Companion Chinese analysis: `reading-notes.zh.md`.
- Editable deck: `slides.en.tex`; build: `build_slides.py`.
- Original paper figures retain their source identifiers and PDF page locators.