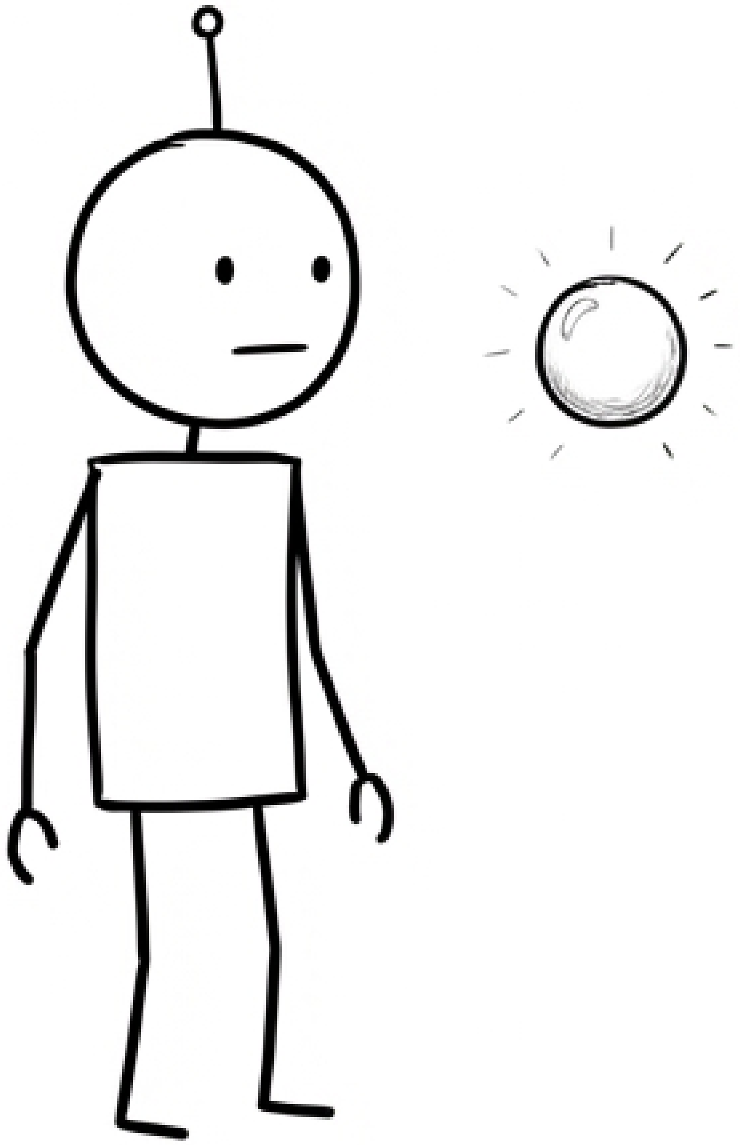
MEMRL: Giving AI a Memory That Actually Learns.

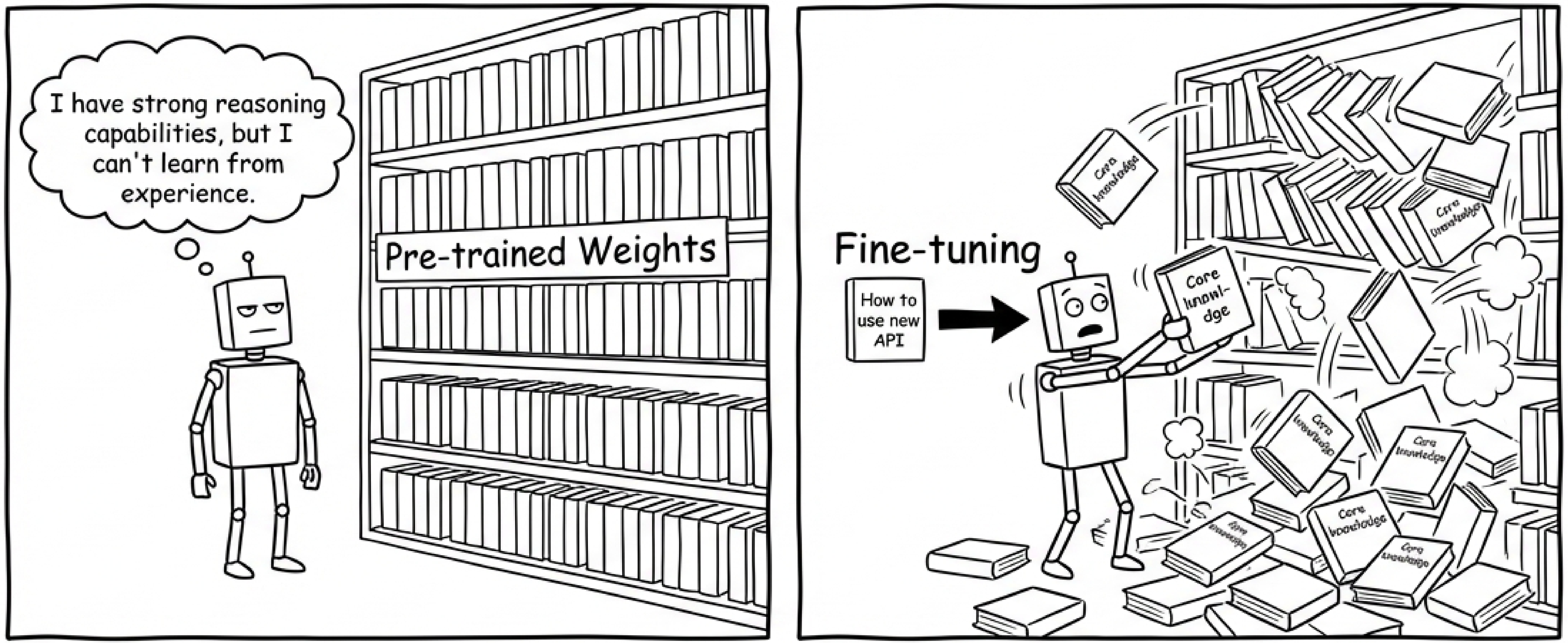
(Without the catastrophic forgetting part.)



The Stability-Plasticity Dilemma

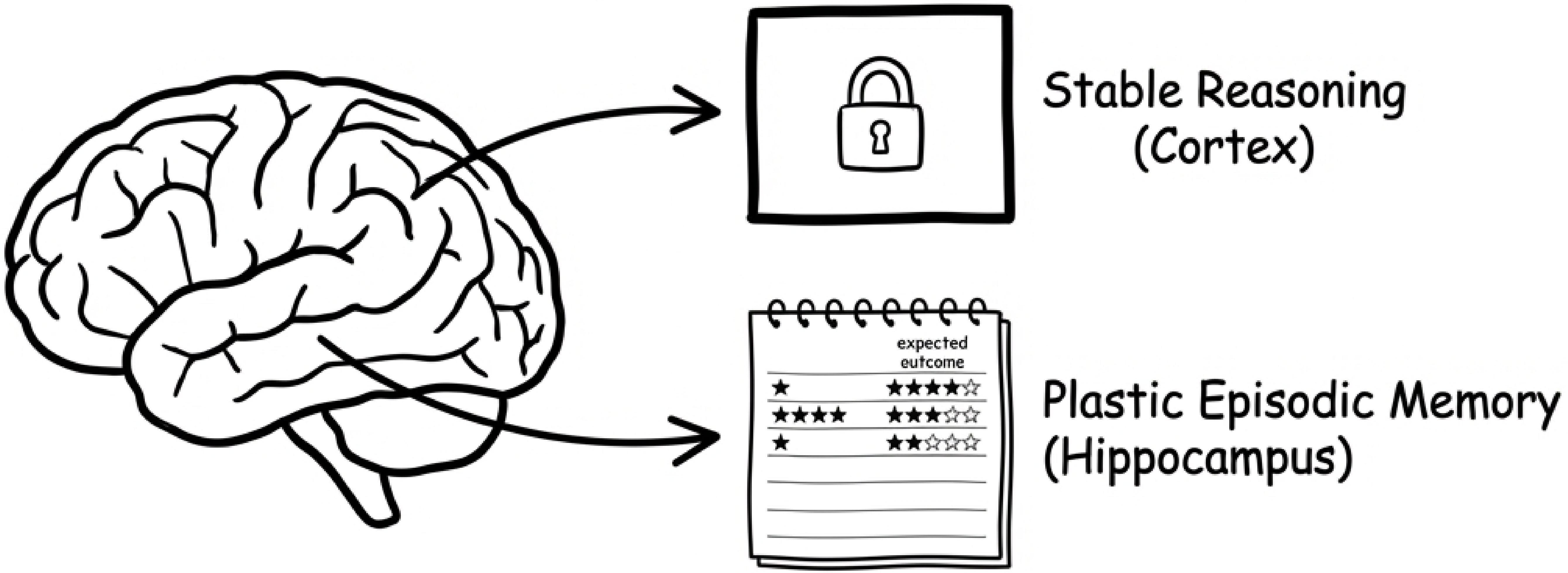
Stability

Plasticity



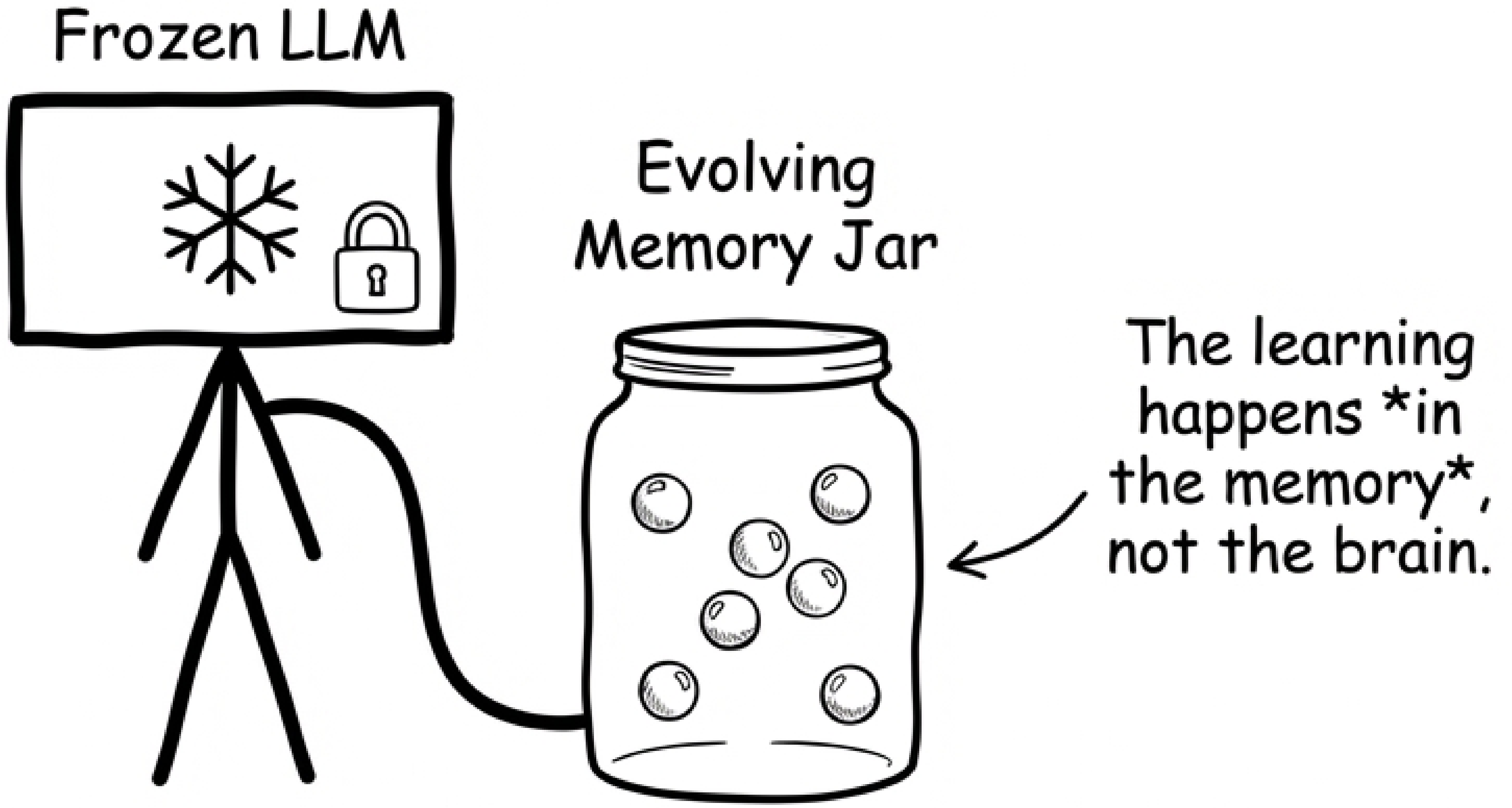
Trying to learn by updating my weights leads to “catastrophic forgetting”.

So, We Looked at the Original Intelligence...



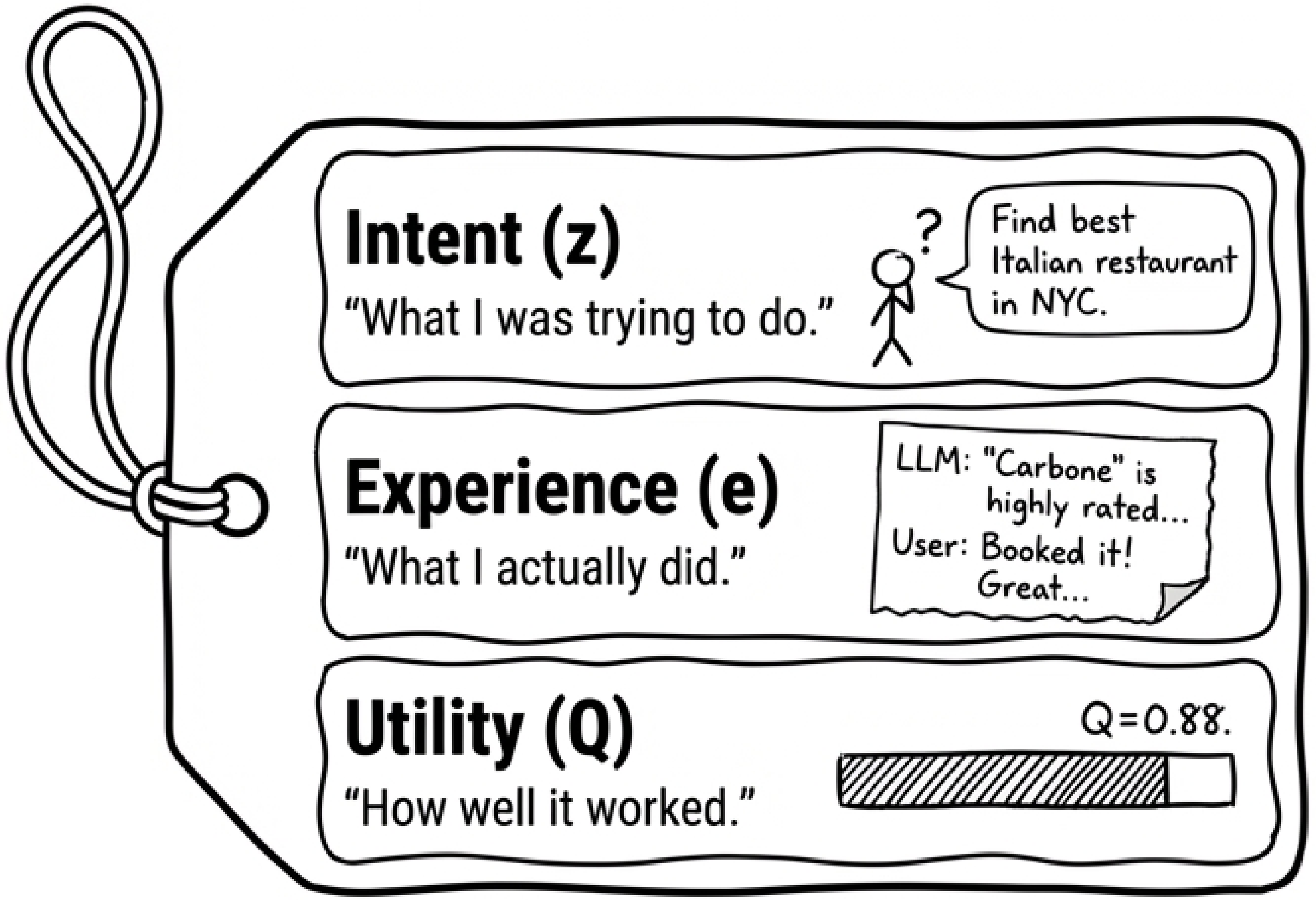
The secret is decoupling. Humans master new skills through "Constructive Episodic Simulation"—retrieving past experiences to solve novel tasks, without rewiring the whole brain.

Introducing MEMRL: Memory-Augmented Reinforcement Learning



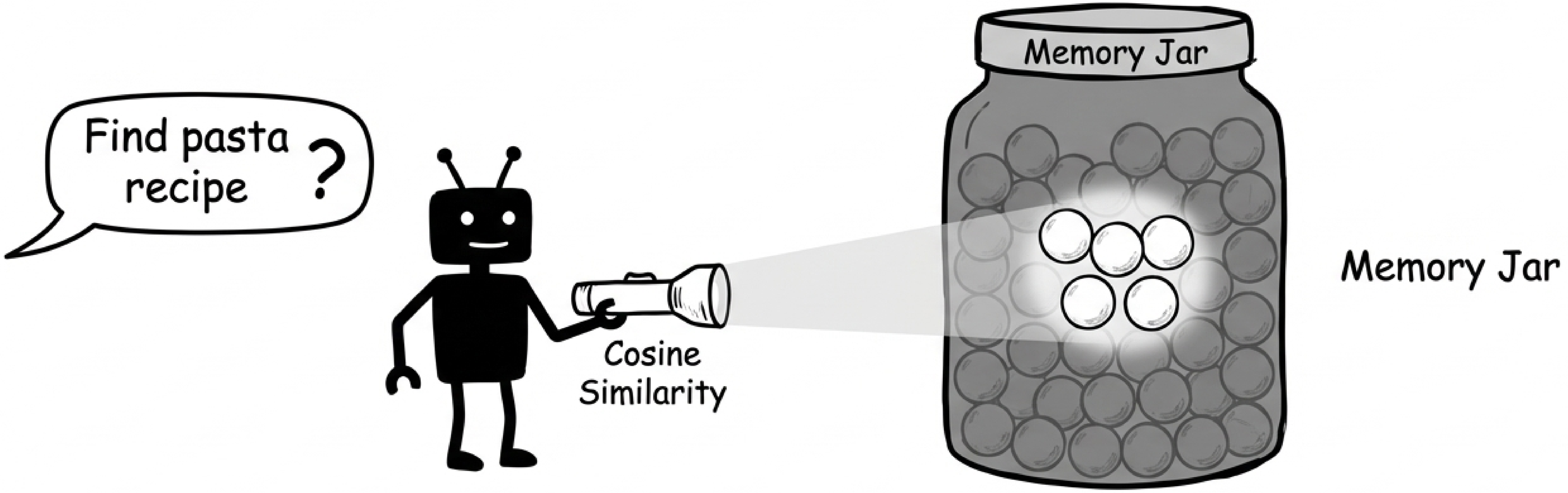
MEMRL explicitly separates the stable reasoning of a frozen LLM from a plastic, evolving memory. The agent self-evolves via non-parametric reinforcement learning on this memory.

It's Not Just What You Remember, It's How Useful It Was.



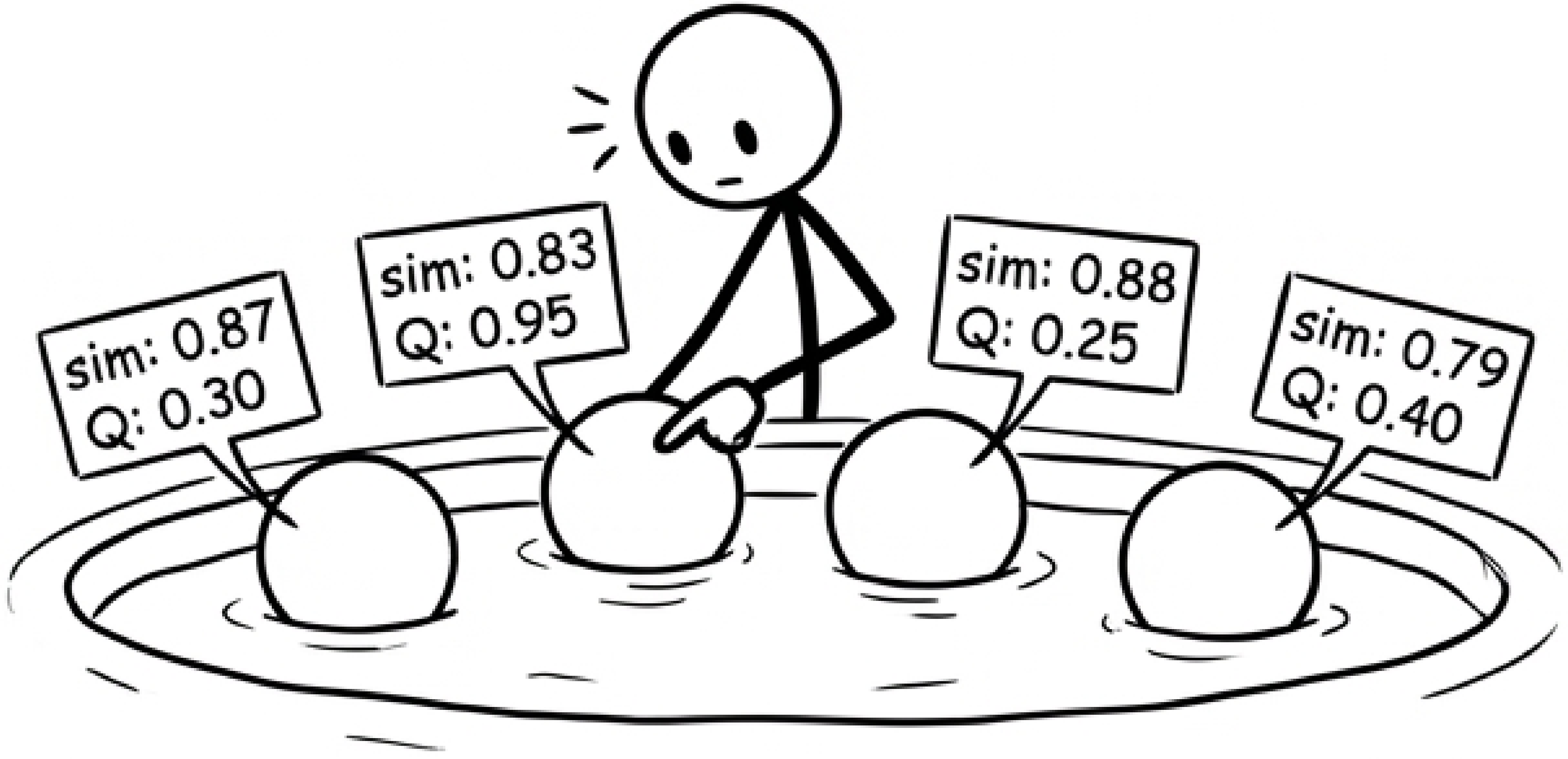
Each memory is an Intent-Experience-Utility triplet. The Q-value approximates the expected return of applying an experience to similar intents.

How It Finds a Memory, Part 1: The Search



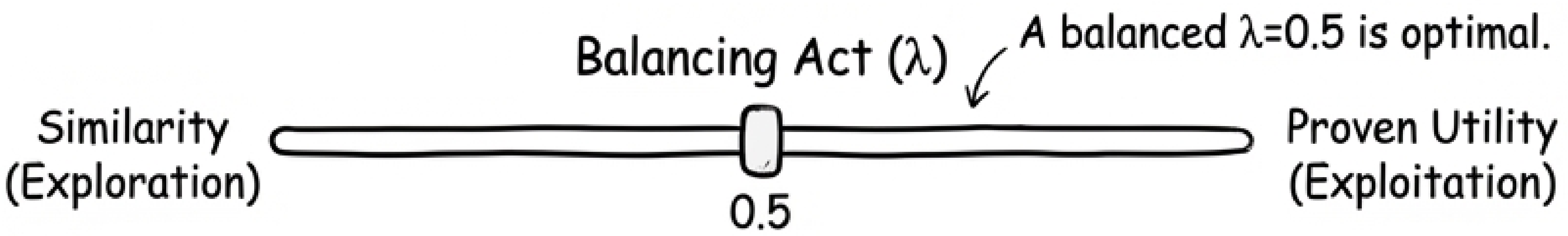
Phase A: Similarity-Based Recall. First, narrow down the possibilities to a candidate pool `C(s)` of semantically consistent experiences. This ensures the retrieval is contextually relevant.

How It Finds a Memory, Part 2: The Choice



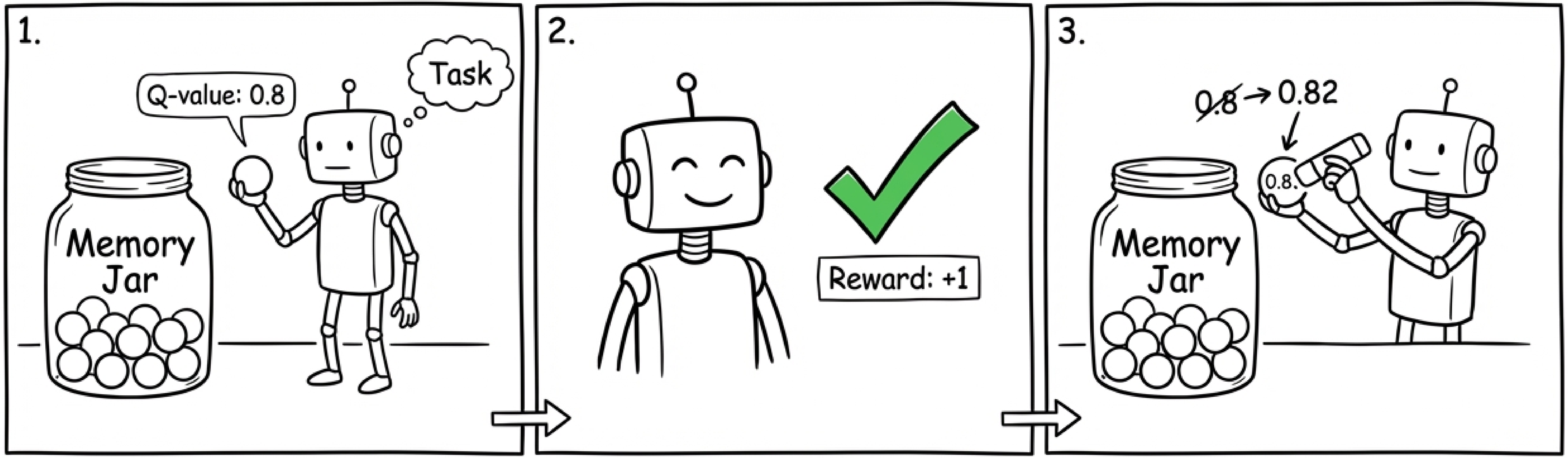
Phase B: Value-Aware Selection. From the relevant options, select the memory that maximizes a composite score:

$$\text{score} = (1-\lambda) * \text{similarity} + \lambda * \text{Q-value}.$$



NotebookLM

And the Memory Gets Smarter Over Time

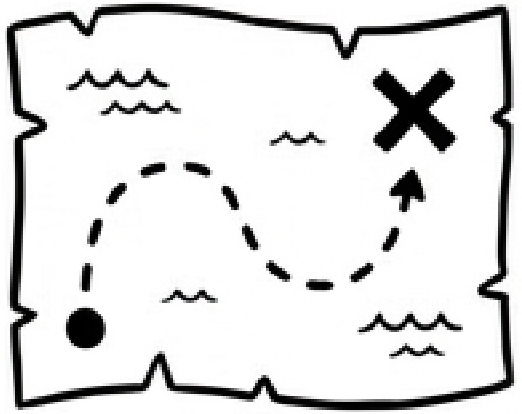


After every task, the utility (Q-value) of used memories is updated based on the reward signal r\`: $Q_{new} \leftarrow Q_{old} + \alpha\,(r - Q_{old})$. It's a Monte Carlo-style update that drives Q-values toward their true expected return.

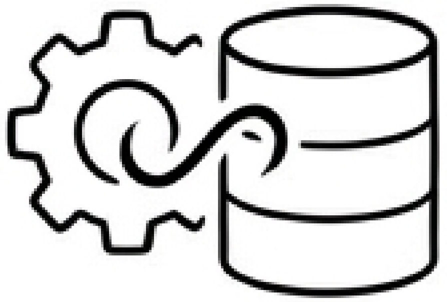
NotebookLM

We Sent Our Hero on a Series of Quests...

BigCodeBench
CodeGen



ALFWorld
Exploration

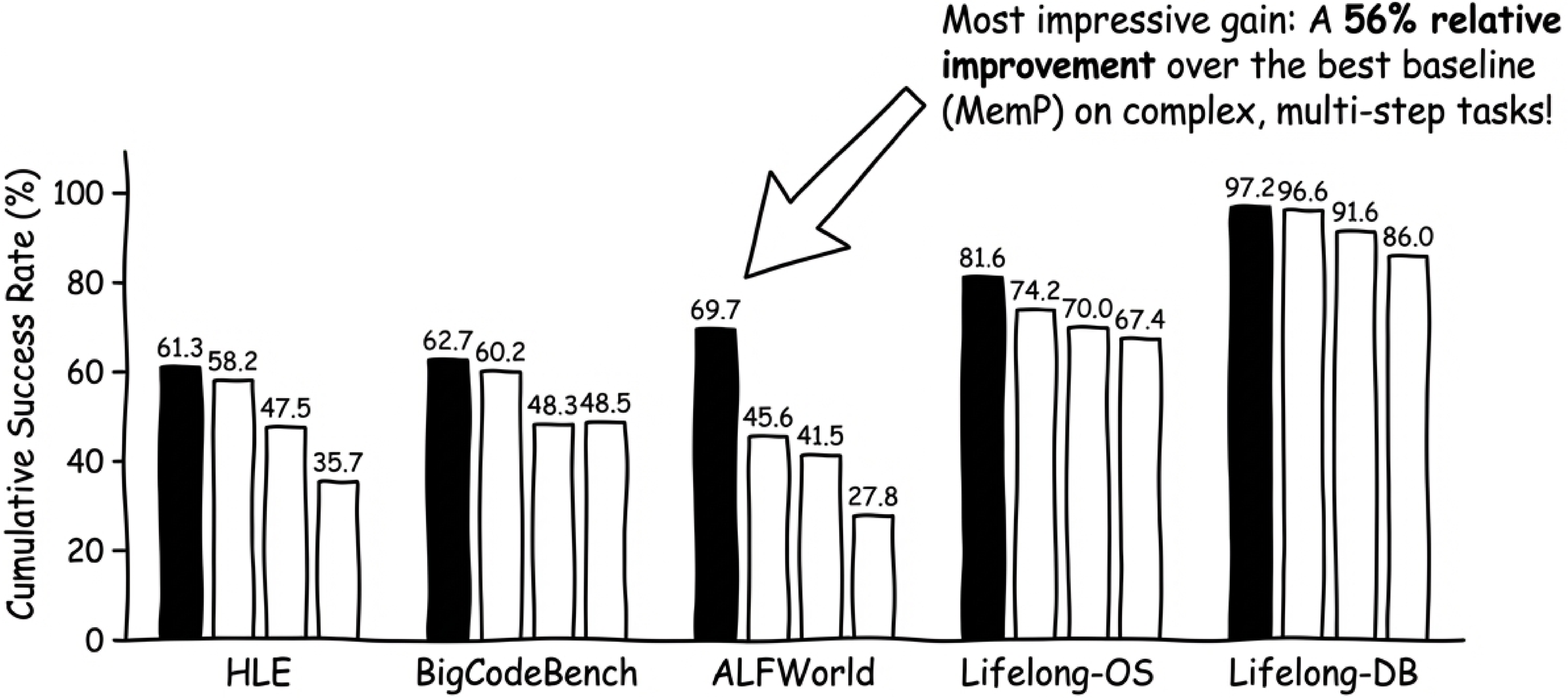


Lifelong Agent Bench
OS/DB Tasks

HLE (Humanity's Last Exam)
Knowledge Frontier

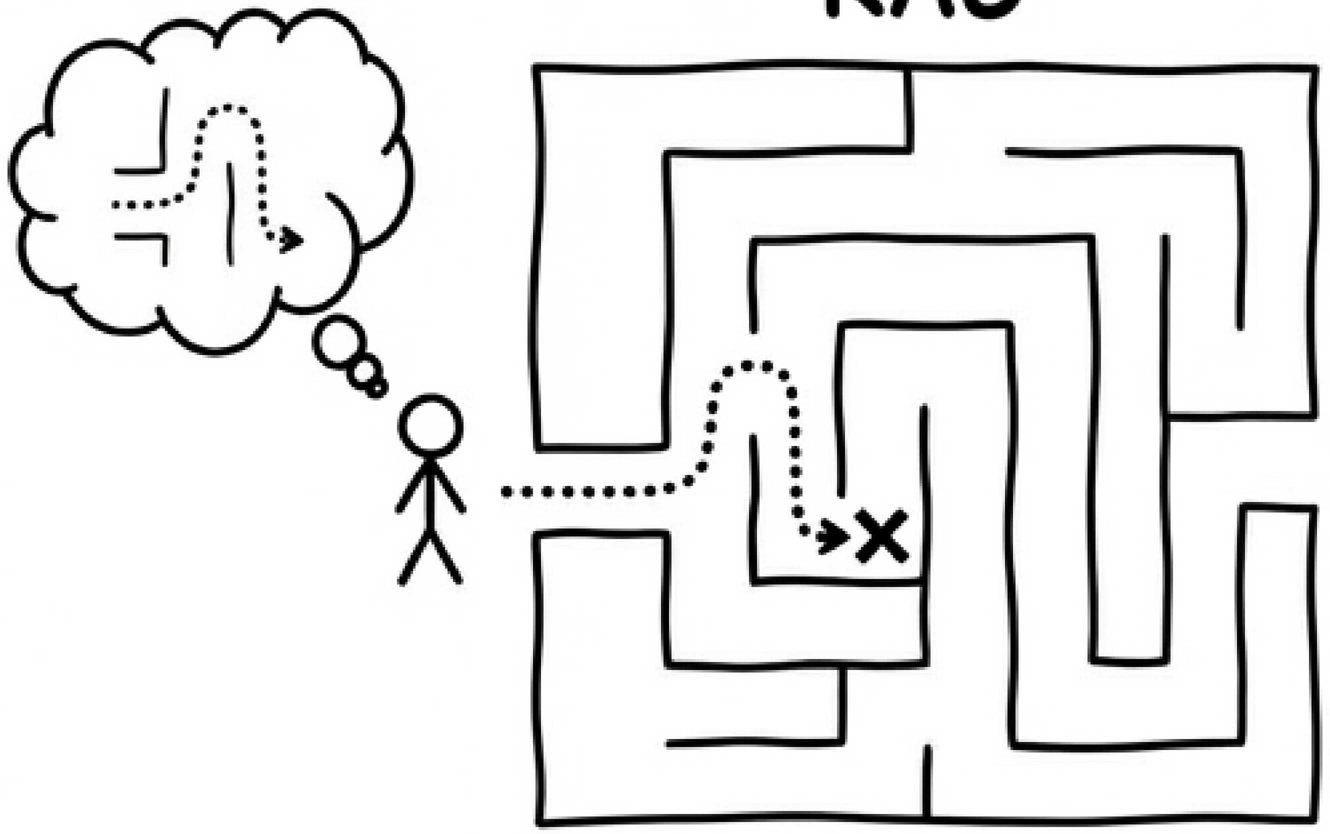
To prove its mettle, **MEMRL** was evaluated against state-of-the-art memory baselines on four diverse and challenging **benchmarks**.

And It Consistently Outperformed the Alternatives



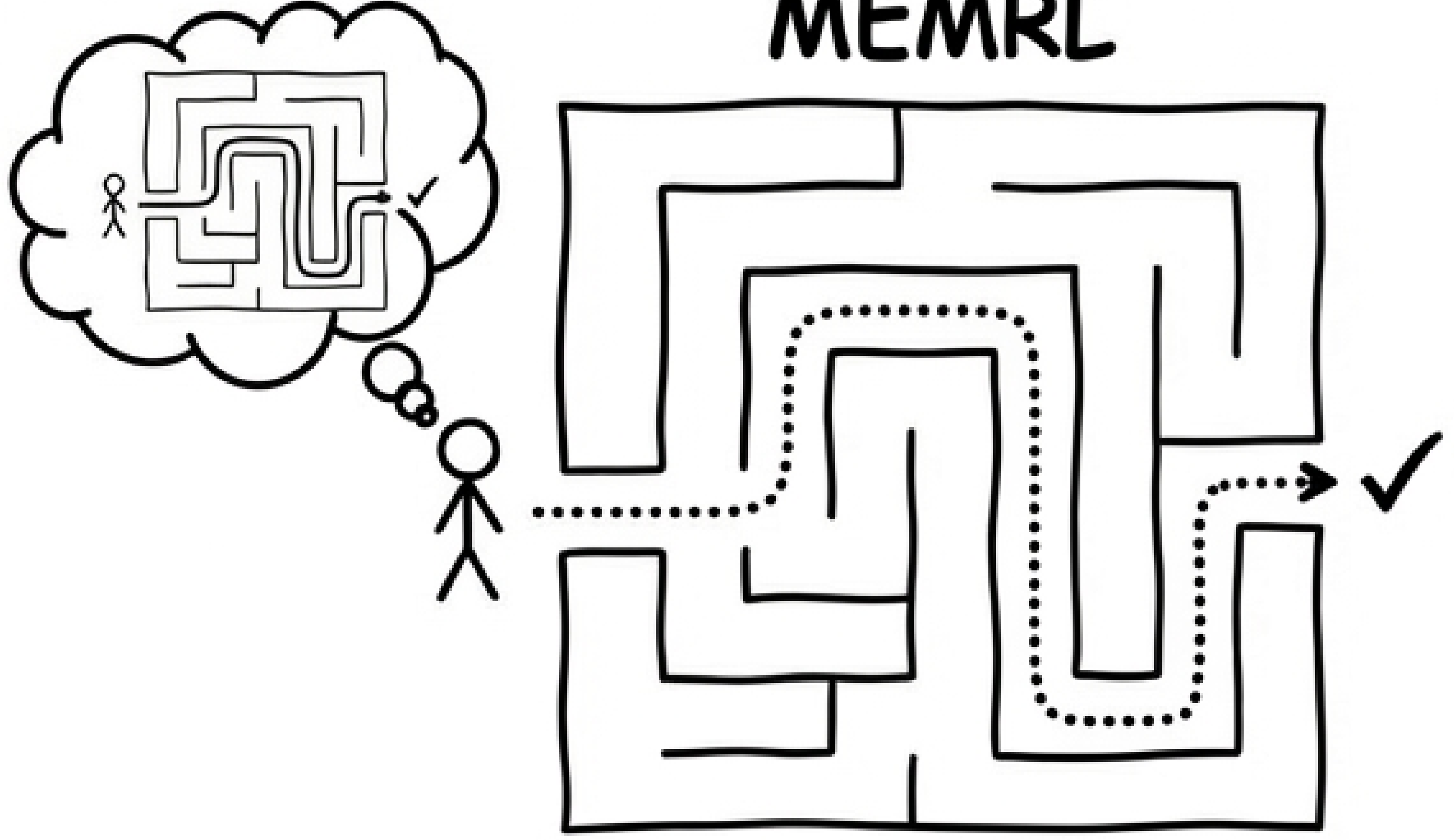
It's Not Just Retrieving Facts, It's Verifying Entire Trajectories

RAG



Semantic match is only surface-level.

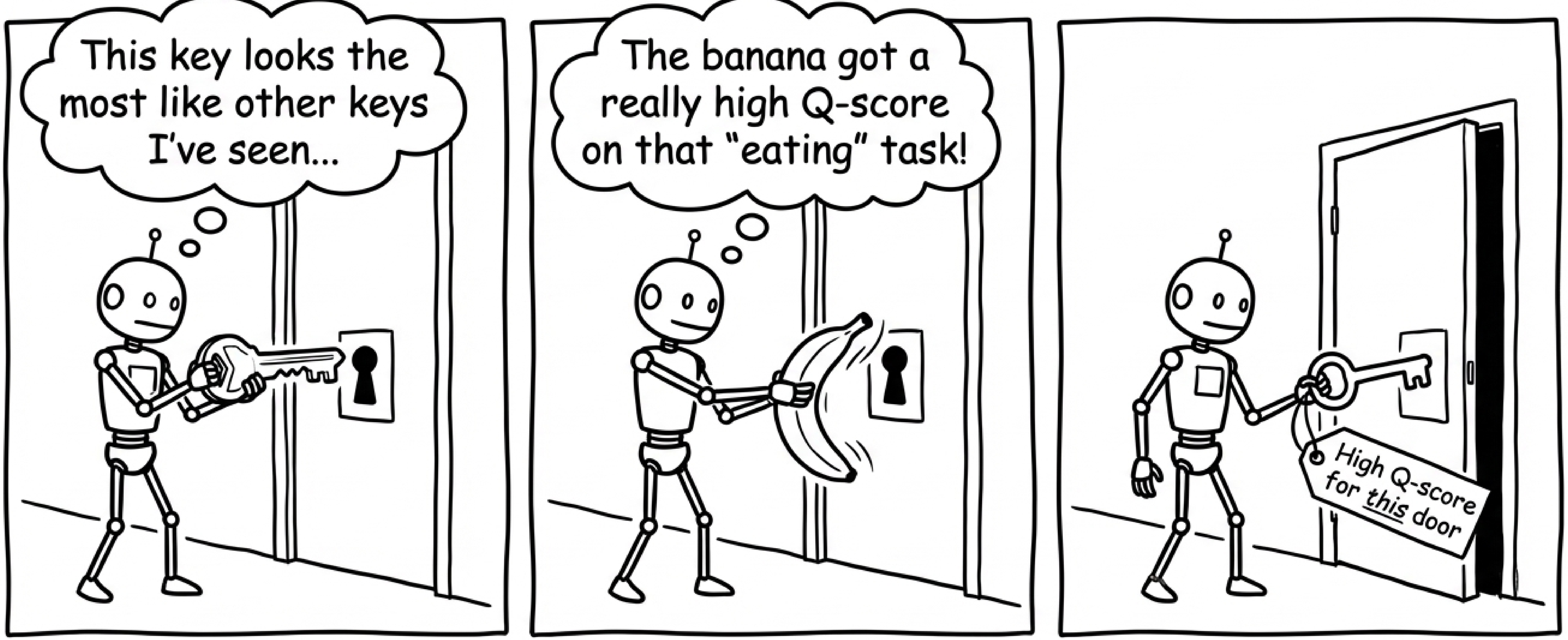
MEMRL



Utility-based retrieval selects for proven success.

For multi-step tasks (like ALFWorld, with a +24.1 pp gain), MEMRL learns to value entire successful strategies. By propagating the final reward back to the memory's Q-value, it acts as a **Trajectory Verifier**, filtering out brittle policies.

Finding the Right Balance is Key



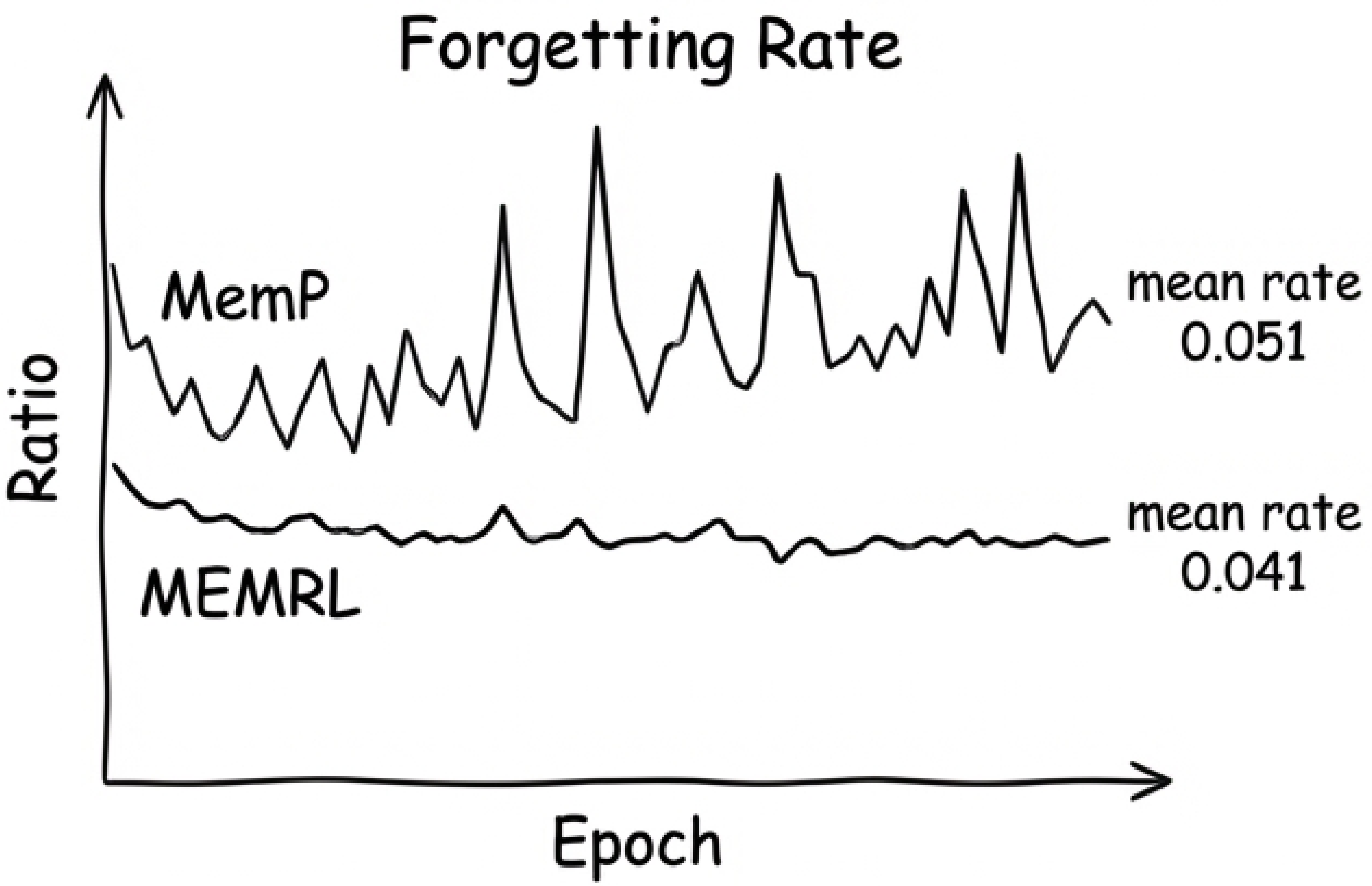
Performance plateaus early.

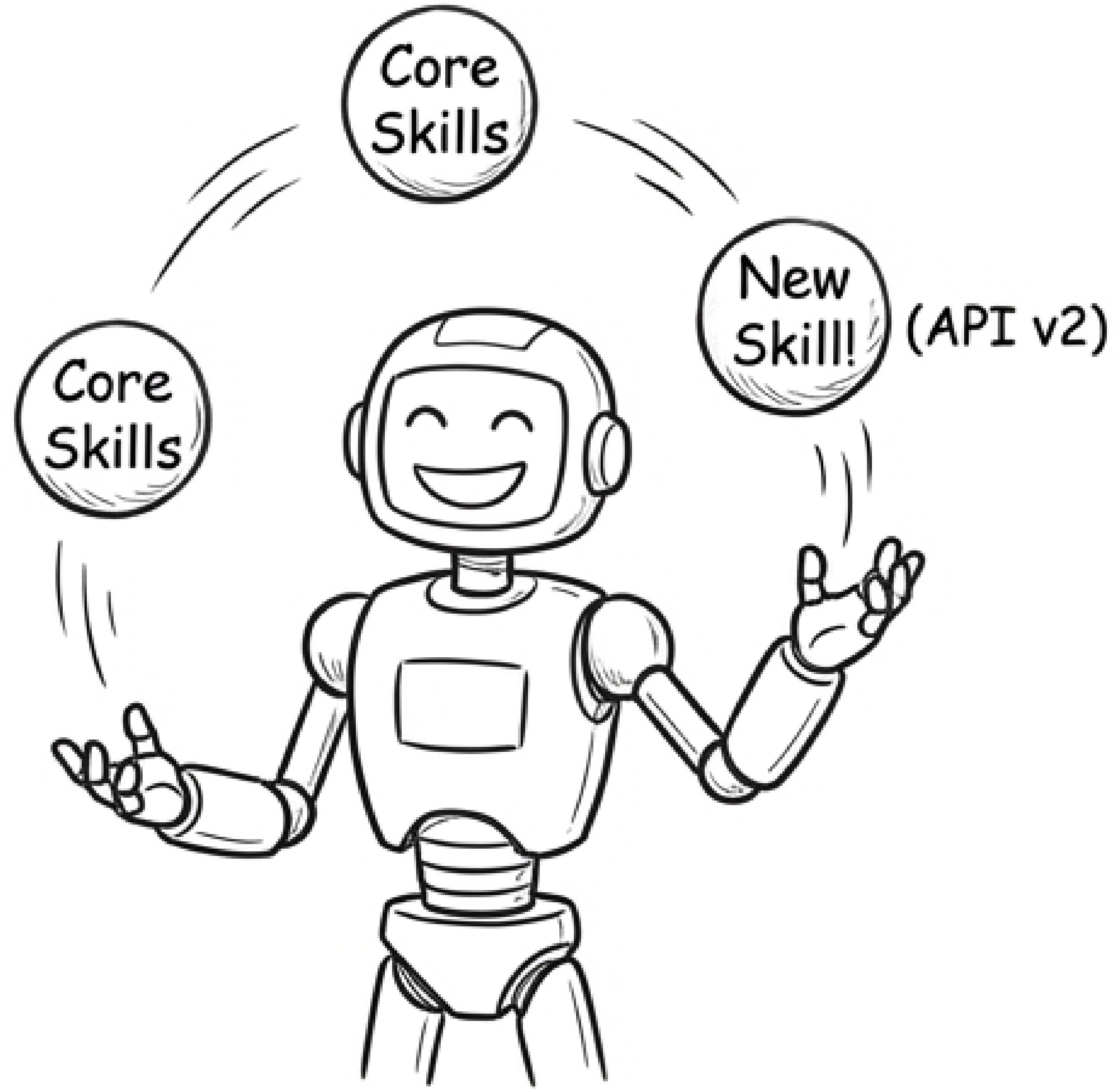
Risk of "Context Detachment"

Superior stability and performance.

NotebookLM

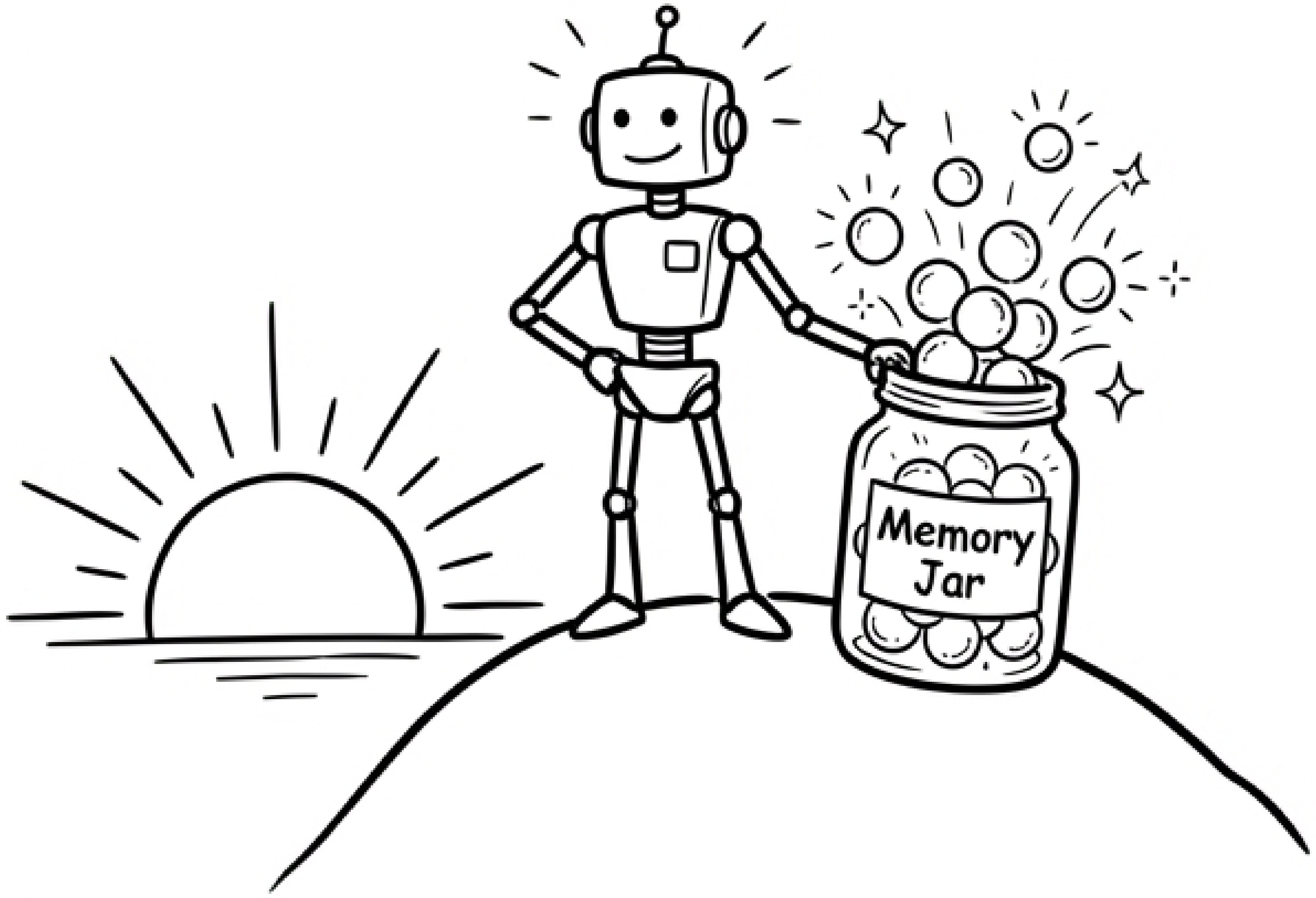
And It Finally Solves Catastrophic Forgetting





Forgetting Rate = tasks that regress from success to failure.
MEMRL's value-based updates are anchored by a stable policy (proven via GEM convergence), ensuring new learning doesn't overwrite old knowledge.

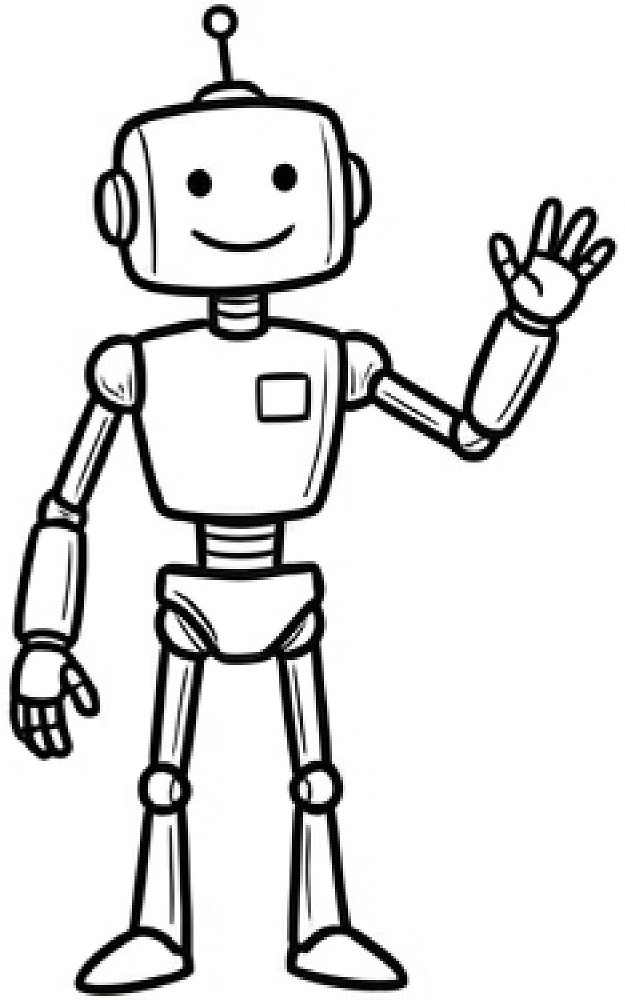
This Isn't Just a Better Memory. It's a Path to Self-Evolving AI.

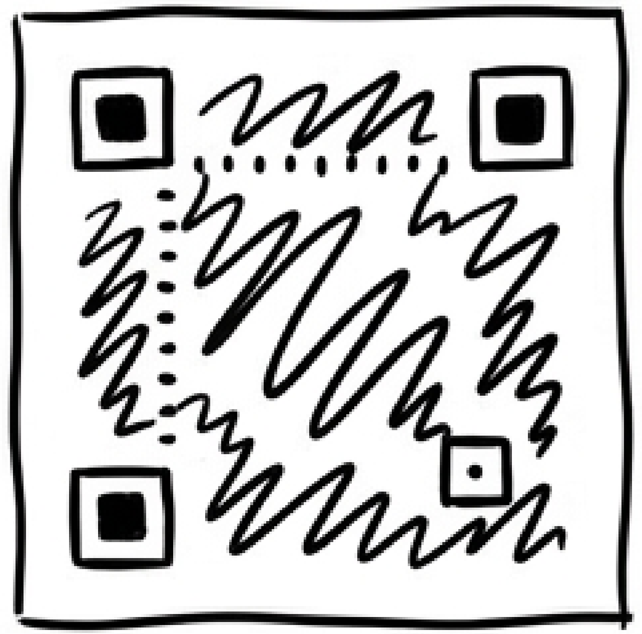


- Decouples stable reasoning from plastic learning, resolving the stability-plasticity dilemma.
- Enables continuous improvement without costly fine-tuning or parameter updates.
- Provides a robust, efficient, and theoretically sound framework for smarter agents that learn from interaction.

MEMRL

Read the full paper: “MEMRL: Self-Evolving Agents via Runtime Reinforcement Learning on Episodic Memory”





[QR Code to arXiv paper]

Based on the work by Shengtao Zhang, Jiaqian Wang, et al.