Neuroscience, Evolution, and the Path to Aligned AI

Research Introduction for the AI Engineering Team

Ethan C. Jackson, PhD

Applied ML Scientist, Vector Institute

January 12, 2026

INTRODUCTION

2014–2019

PhD Researcher

Western University

- Evolutionary RL / NAS
- Novelty Search
- Neuroimaging

2019–2020

Postdoctoral Researcher

University of Guelph

- Deep Learning for Forecasting
- Canada's Food Price Report

2020–2022

Applied ML Scientist

Vector Institute

- Data Science & Forecasting
- Privacy Technologies

2022–2024

Social AI Researcher

University of Toronto

- Multiagent RL
- Artificial Hippocampus
- Biologically Plausible Learning

2022–2025

Co-Founder

ChainML / Theoriq

- Analytics Agents
- Memory & Learning
- Agent-to-Agent Protocol

2026–Present

Applied ML Scientist

Vector Institute

- Agents & Reasoning
- Social & Evolutionary AI

THREE RESEARCH THEMES

Neuroscience-Inspired AI

Leveraging brain insights to build better AI – **memory systems** and **learning algorithms**

Evolutionary Methods

LLM-guided program search and **ES for fine-tuning** – evolutionary algorithms are powerful in the LLM era

Social Learning Thesis

AI alignment may require agents that learn in **environments with real consequences** – not just by RL after the fact

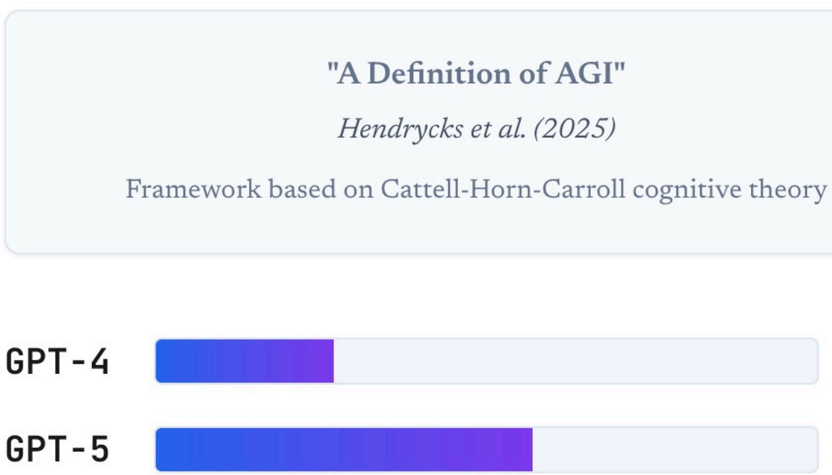
From how individuals learn, to how populations improve, to how groups coordinate.

THE MEMORY GAP

"A Definition of AGI"

Hendrycks et al. (2025)

Framework based on Cattell-Horn-Carroll cognitive theory

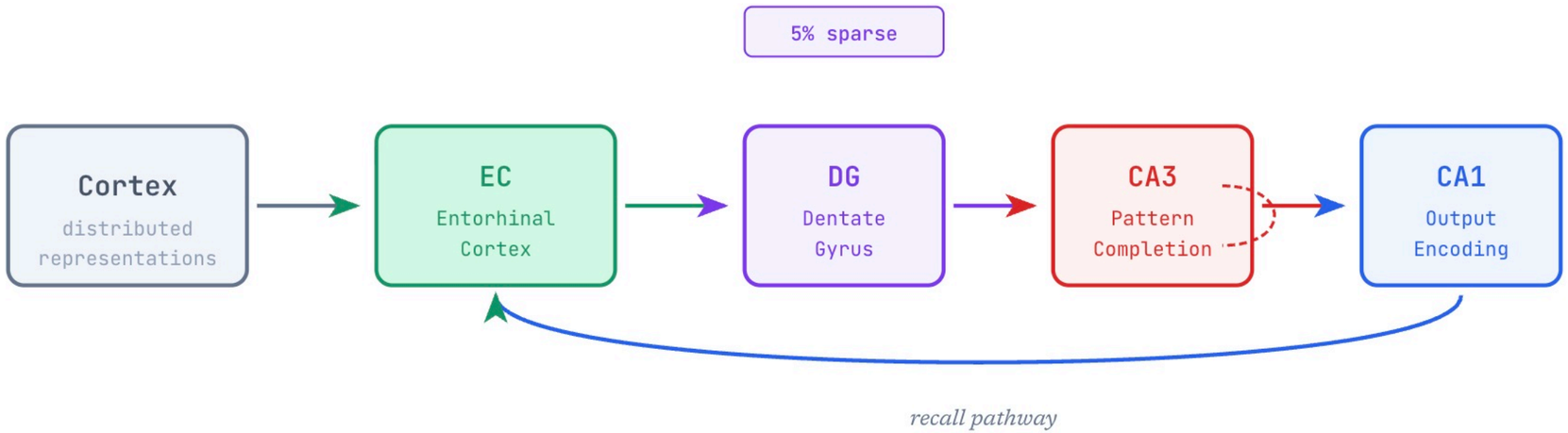


Cognitive Profile

- ✓ Reasoning
- ✓ Knowledge
- ✓ Perception
- ✓ Processing Speed
- ✗ Long-term Memory
- ✗ Learning Efficiency
- ✗ Experiential Memory

Primary gap: **Memory systems** – and memory isn't just about capability.
It's about learning from consequences over time.

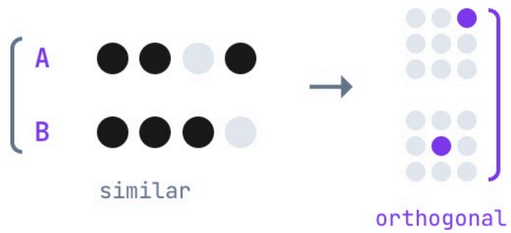
What can neuroscience teach us about building memory systems for AI?



The hippocampus balances **pattern separation** (distinct memories) with **pattern completion** (retrieval from cues)

Pattern Separation

Dentate Gyrus



- Similar inputs → distinct sparse codes
- Prevents memory interference
- Only ~5% of neurons active

Pattern Completion

CA3

- Partial cue → full memory recall
- Dense recurrent connections
- Associative retrieval

The tradeoff: Too much separation → can't generalize. Too much completion → memories blur.

HIPPOCAMPAL DNN ARCHITECTURE

Complementary Learning Systems: Fast hippocampal encoding + Slow neocortical consolidation

MSP: Monosynaptic Pathway ENCODING

Theta Phase 1 – Direct autoencoder baseline (CA3 inhibited)

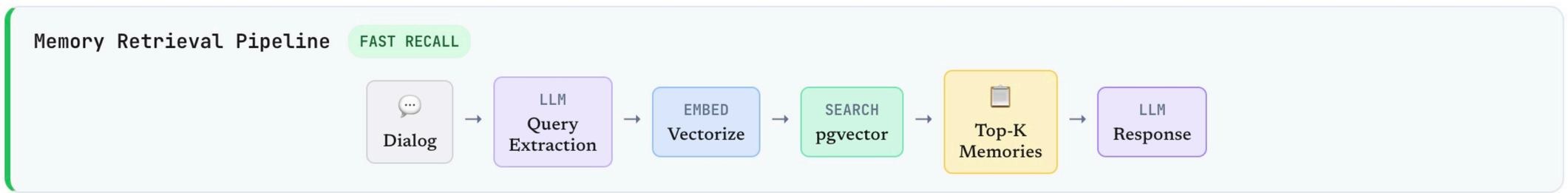
TSP: Trisynaptic Pathway RECALL

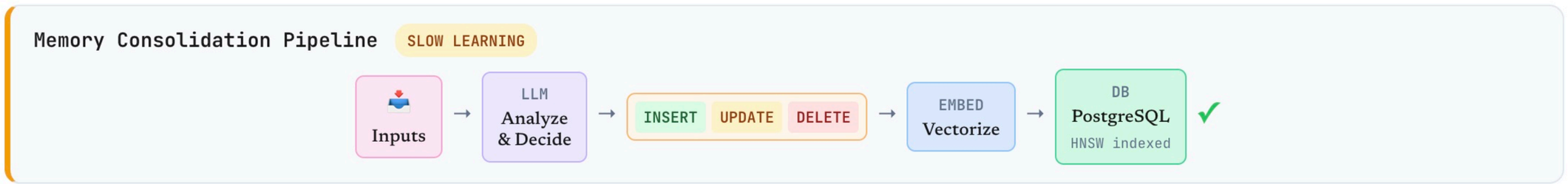
Theta Phase 2 – Pattern separation → Association → Completion

- **Fast learning** – One-shot episodic encoding (sparse → low interference)
- **Slow learning** – Gradual semantic consolidation (distributed representations)

ENGINEERING EPISODIC MEMORY FOR AGENTS

We don't need hippocampal DNN modules to give agents episodic memory. **RAG-based architectures** offer practical engineering solutions – with parallels to biological complementary learning systems.

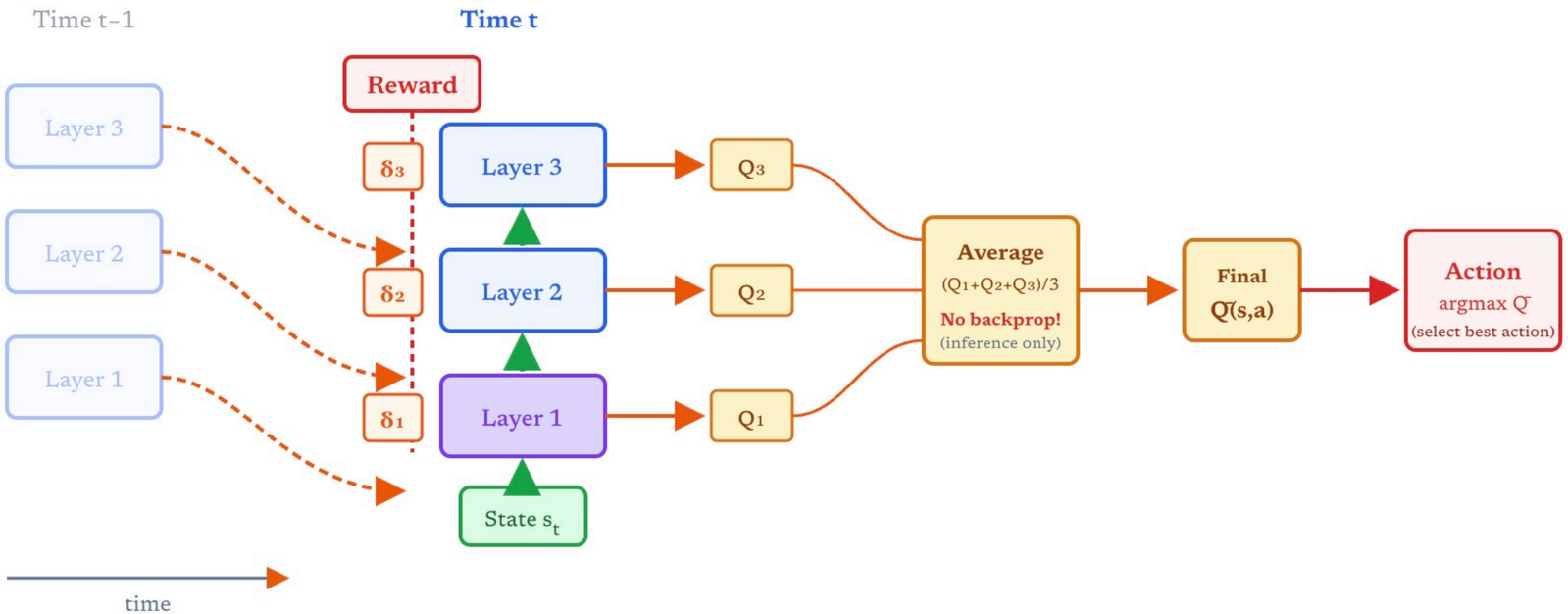




*Two distinct phases mirror biological memory: **retrieval** (pattern completion) and **consolidation** (post-interaction learning).*

What would AGI benchmarks look like for a **memory-enabled agent**?

ARTIFICIAL DOPAMINE: DISTRIBUTED TD LEARNING



Guan, Verch, Voelcker, Jackson, Papernot & Cunningham – NeurIPS 2024

Info UP (within timestep) ---- Activations DOWN (across time) ---- Reward broadcast δ Local TD error

Key insight: Each layer independently computes $\delta = r + \gamma \cdot \max Q(s',a') - Q(s,a)$ and updates its own weights. No error propagation between layers.

Memory helps agents learn from their own experience.

But what if we could learn from the experience of **many agents at once**?

EVOLUTIONARY PROGRAM SEARCH

LLM-driven evolution discovers algorithms by iteratively mutating, evaluating, and selecting programs

🧬 Darwin Gödel Machine

Zheng, Hu, Lu, Lange & Clune – 2025

Search space: Coding agent architectures

Fitness: SWE-bench, Polyglot benchmarks

SWE-bench	20% → 50%
Polyglot	14% → 31%

Self-modifying agents that improve their own code editing tools, context management, peer-review

📈 ProFiT

Soper, Khalifa, Soros, Nasir, Azhang & Togelius – 2025

Search space: Python trading programs

Fitness: Walk-forward validation returns

vs Buy-Hold	77% win rate
vs Random	100% win rate

Evolving strategies adapt to non-stationary markets via code mutation + self-analysis

Policies aren't just LLM prompts

💬 LLM Agent 🧠 Neural Net 📜 Program

Evolutionary search operates over **any** executable representation – LLMs as mutation operators, not just as policies

Open-source tools like OpenEvolve make this accessible to try today

EVOLUTION STRATEGIES VS REINFORCEMENT LEARNING

ES scales to billions of parameters by exploring in parameter space, not action space

Evolution Strategies (ES)

Explore in **parameter space**

Population N=30 (fitness-colored)

$$\theta = \theta_0 + \sigma \cdot (\varepsilon_1 + \varepsilon_2 + \cdots + \varepsilon_t)$$

Seeds → Gaussian noise trajectory

- One noise sample per trajectory
- No backpropagation (inference only)
- Low variance, stable optimization
- **No reward hacking** observed

Key trick: Store seeds, not tensors — Memory: O(generations) vs O(7B params)

Reinforcement Learning (PPO/GRPO)

Explore in **action space**

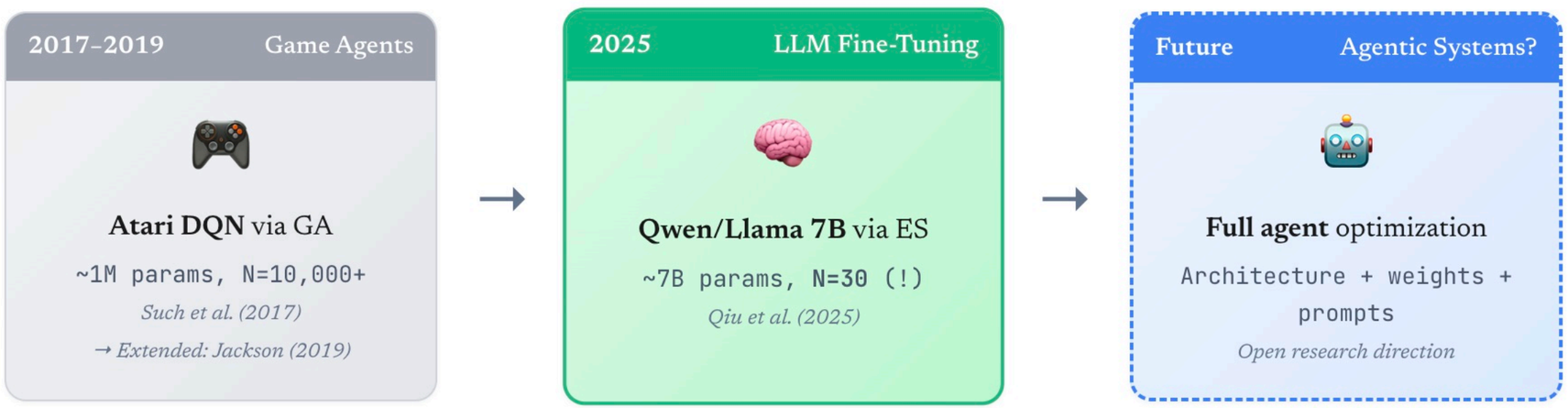
Noise at every token

- Noise at every step → high variance
- Requires backpropagation
- Credit assignment across 100s of tokens
- Prone to reward hacking (needs KL penalty)

Empirical: 15.5× higher std across runs

Qiu, Gan, Hayes, Liang, Meyerson, Hodjat & Miikkulainen — "Evolution Strategies at Scale: LLM Finetuning Beyond RL" (2025)

The same evolutionary principles scale from millions to billions of parameters



🔬 Why Does N=30 Work?

Low intrinsic dimensionality – LLMs' effective parameter space is much smaller than nominal. Consistent with LoRA.

🌊 Landscape Smoothing

Gaussian convolution – Population averaging smooths loss surface, enabling stable optimization.

Implication: Evolutionary methods may unlock optimization for agentic systems that gradient-based RL cannot – architecture search, prompt evolution, multi-component agents.

Evolution optimizes populations —
but each agent is evaluated **independently**.

What do we know about coordination, and
cooperation in multi-agent systems?

THE SOCIAL LEARNING THESIS

Why consequence-driven development matters

How We Train AI

RLHF creates behavioral dispositions, but not through the developmental process that makes human values robust. The model learns **what to say**, not *what it's like to cause harm*.

How Humans Learn Values

Children learn from early on that harmful actions lead to **concrete, enforced consequences**. This shapes intrinsic motivation — not just surface compliance.

"We cannot engineer alignment into the weights of AI models."

What we *can* engineer are the **environments** — and the selection pressures — that make alignment adaptive.

Risks from algorithms: Standard coordination produces harmful patterns that persist across agent generations.

Gelpí, Tang, Jackson & Cunningham – PNAS Nexus 2025

Risks from interactions: Collective behaviors may be harmful even when individual agents appear safe.

Tomašev et al. – Distributional AGI Safety 2025

The opportunity: Design sandbox economies where safe behaviors emerge through structured interaction and real consequences.

Tomašev et al. – Virtual Agent Economies 2025

INTERESTING DIRECTIONS

A few areas I think are worth exploring

📊 Agentic Evals

Most evals target *models*. We need benchmarks for **agentic systems** – especially memory-augmented agents.

- Episodic memory benchmarks
- Multi-turn reasoning with context

🧬 Evolutionary Methods

Evolutionary approaches for agent improvement – optimizing weights **or** evolving code/scaffolding.

- ES for weight optimization
- Evolutionary program search (DGM-style)

👥 Social Learning

Building AI systems that develop **intrinsic motivation** through experience with real consequences.

- Consequence-aware training paradigms
- Multi-agent environments with accountability

Memory lets agents learn from their experiences. Evolution lets them learn from each other's attempts. Social structure lets them learn why cooperation matters. We need all three.

Thank you

ethan.jackson@vectorinstitute.ai