Modern Agent: Training & Self-Evolving

Rui Shi

3/17/2026

香港城市大學
City University of Hong Kong

Hong Kong Institute of AI for Science

Outline

1. Overview

2. Post-training: How can we build a reliable Agentic System

3. Self-Evolving: How should we continue to raise the ceiling

Outline

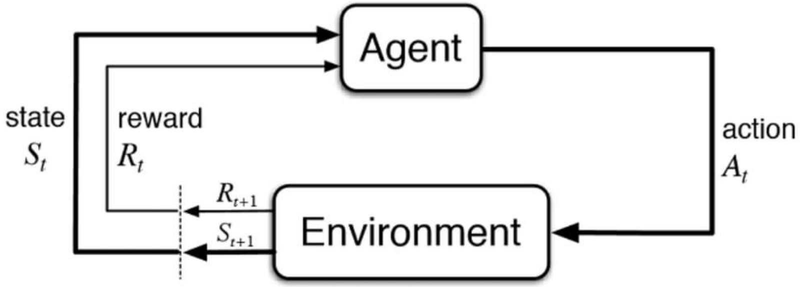
1. Overview

- Two Views
- LLM Agent vs. Agentic System

2. Post-training: How can we build a reliable Agentic System

3. Self-Evolving: How should we continue to raise the ceiling

Moden Agent != LLM + external environment



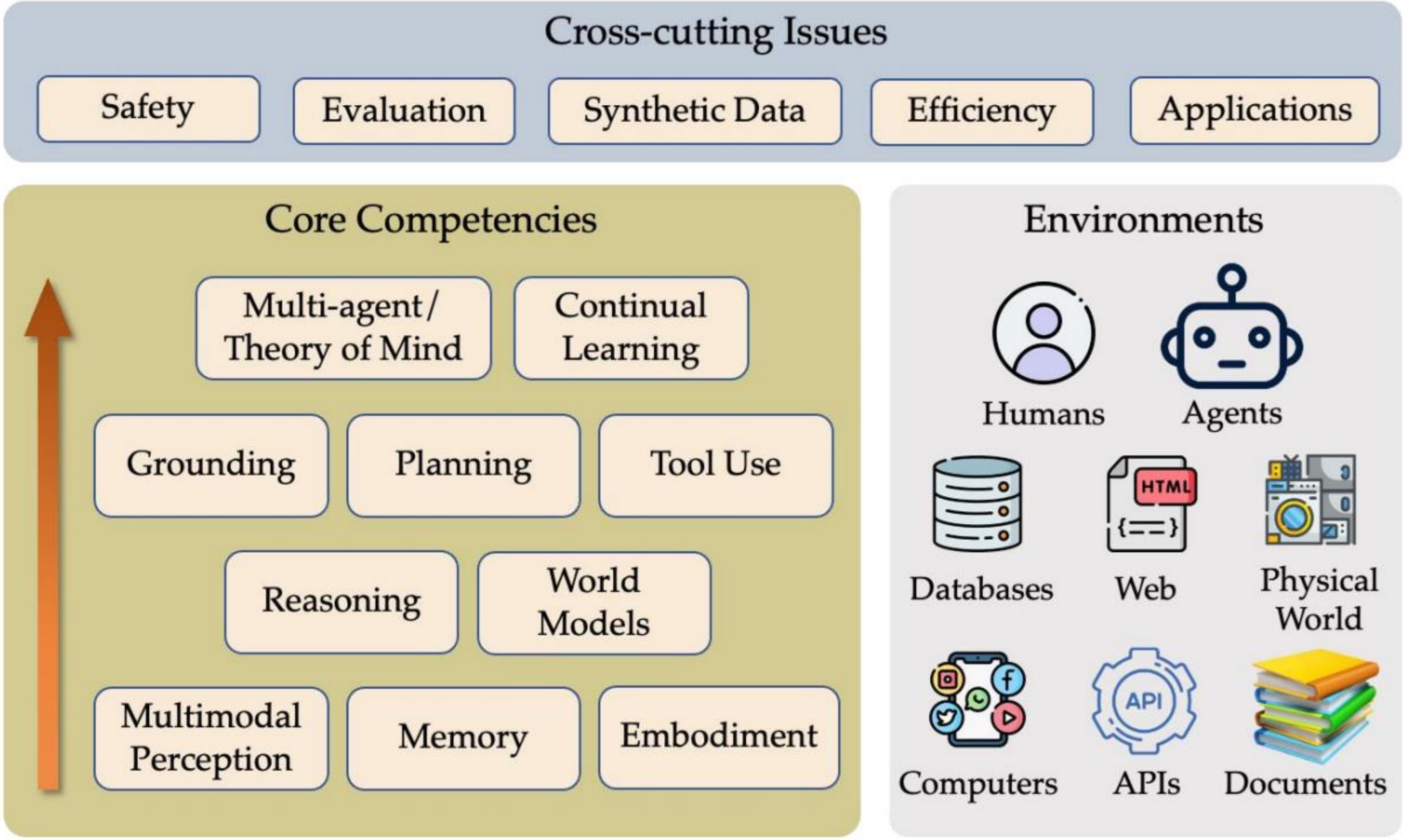
LLM-first view We make an LLM into an agent! ->-> LLM Agent

Implications: scaffold on top of LLMs, prompting-focused, heavy on engineering

Agent-first view We integrate LLMs into AI agents so they can use language for reasoning and communication! ->-> Agentic System

Implications: All the same challenges faced by previous AI agents (e.g., perception, reasoning, world models, planning) still remain, but we need to (re)examine them through the new lens of LLMs and tackle new ones (e.g., synthetic data, self-reflection, internalized search)

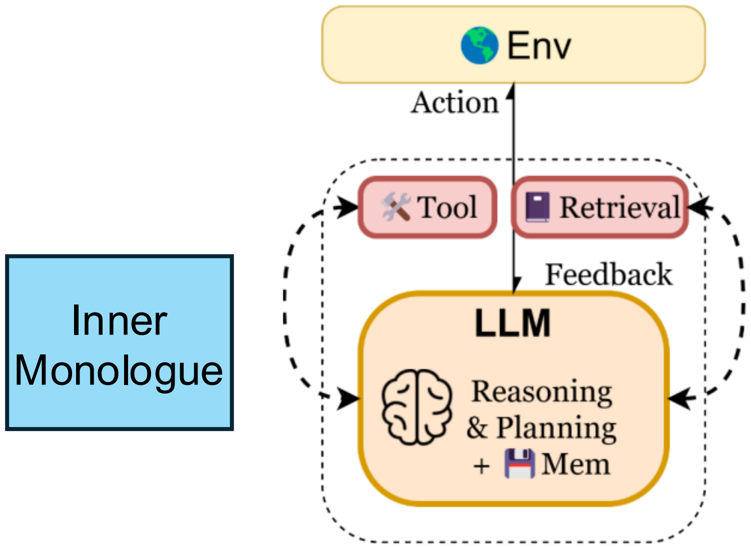
A conceptual framework you might know



Stage 1: LLM Agent (LLM is the only brain)

Contemporary AI agents, with integrated LLM(s), can *use language as a vehicle for reasoning and communication*

- Instruction following, in-context learning, output customization
- Reasoning (for better acting): state inferences, self-reflection, replanning, etc.

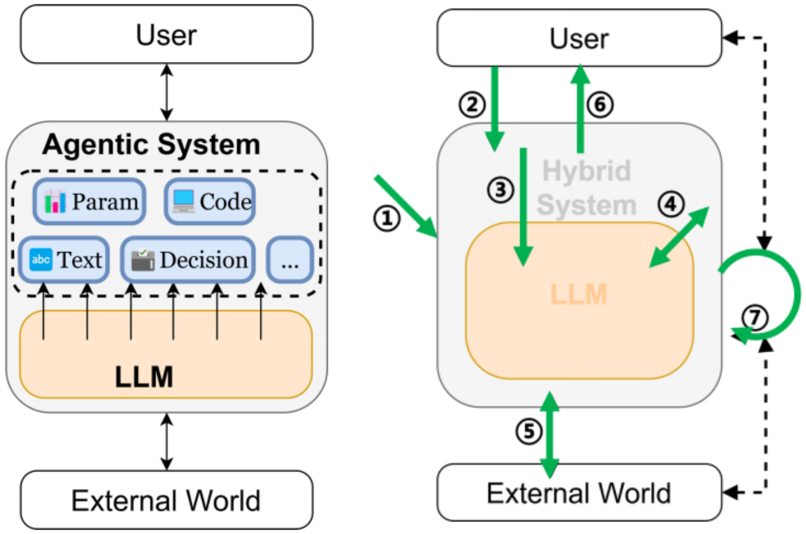


- Reasoning by generating tokens is **a new type of action** (vs. actions in external environments)
- **Internal environment**: where reasoning takes place in an inner monologue fashion
  - **Self-reflection:** a ‘meta’ reasoning action (i.e., reasoning over the reasoning process), akin to meta-cognitive functions
- **Percept** and **external action spaces**:
  - Using language for communication and multimodal perception

Stage 2: Agentic System (Hybrid System for production)

LLM != Omnipotence. LLM = an advanced reasoning function, supported by rigorous code logic, safety guardrails, and with multi-module orchestration to ensure reliability.

- Hybrid Architecture: Integrates neural components (LLMs, green blocks) with symbolic components.
- Production-Ready Engineering System: LLM serves as a core component or cognitive engine



1. Host: prepares the model(s) and deploys the system

2. User: send request to the system

3. **System**: process the request and invoke the model(s)

4. Model: interact with rest of the system

5. **System**: interact with the External World

6. **System**: respond to User

<u>7. **System:** continuously running for long-term tasks</u>

Ahead: A hybrid/agent system sometimes also interacts with another hybrid system, forming multi-LLM/multi-agent communications

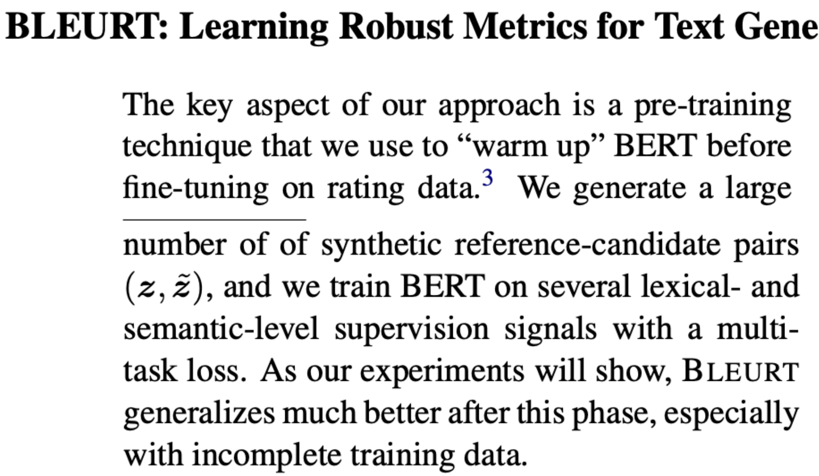
Outline

1. Overview

2. Post-training: How can we build a reliable LLM Agent

- SFT (data)
- RL (data, grade & eval, etc.)

3. Self-Evolving: How should we continue to raise the ceiling

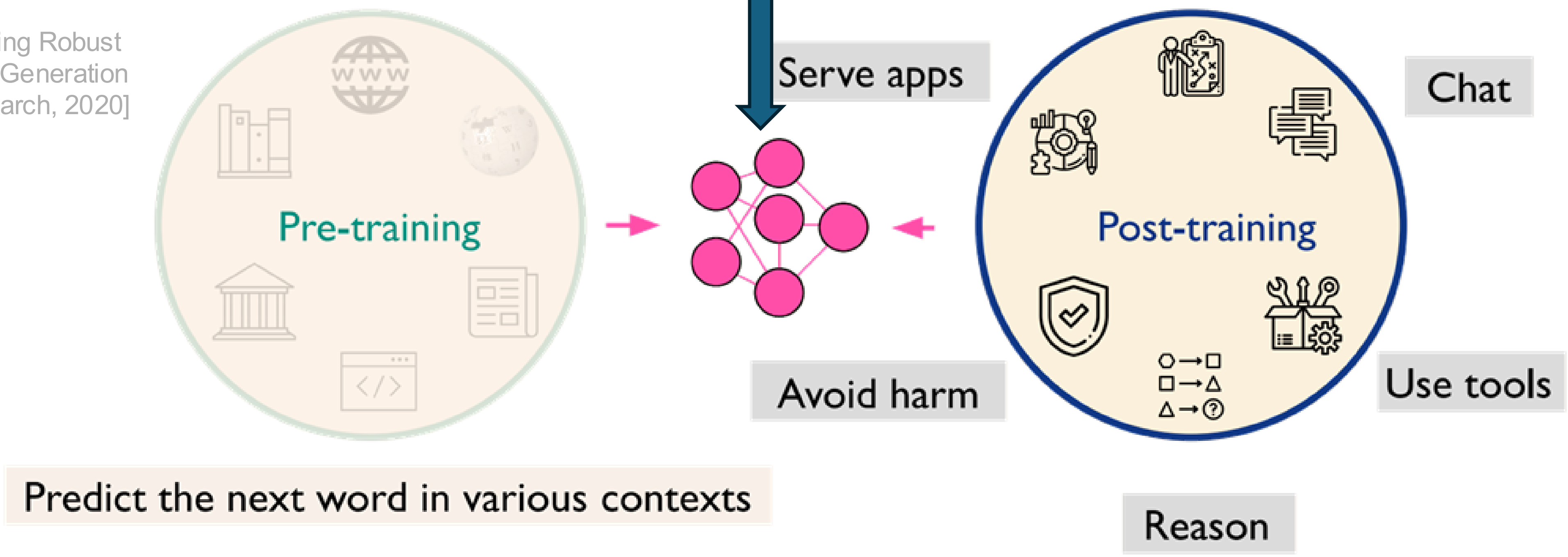


BLEURT: Learning Robust Metrics for Text Generation

The key aspect of our approach is a pre-training technique that we use to "warm up" BERT before fine-tuning on rating data.[3] We generate a large number of of synthetic reference-candidate pairs $(z, \tilde{z})$, and we train BERT on several lexical- and semantic-level supervision signals with a multi-task loss. As our experiments will show, BLEURT generalizes much better after this phase, especially with incomplete training data.

BLEURT: Learning Robust Metrics for Text Generation [Google Research, 2020]

Mid-training/continual pre-training



Post-training

Goal

Goal Oriented

Tool Usage

Plan Reasoning

User Interaction

Supervise finetuning (SFT)
(Cold Start)

Data

$$L_{SFT}(\theta) = -\frac{1}{T}\sum_{t=1}^{T}\log \pi_\theta(y_t|x, y_{<t})$$

$$\mathcal{L}_{\text{GRPO}}(\theta) = \mathbb{E}_{\substack{q\sim P(Q)\\ \{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)}}\left[\mathcal{L}'_{\text{PG-GRPO}}(\cdot,\theta) + \beta\mathcal{L}'_{\text{KL}}(\cdot,\theta,\theta_{\text{ref}}) + \alpha\mathcal{L}'_{\text{Entropy}}(\cdot,\theta)\right]$$

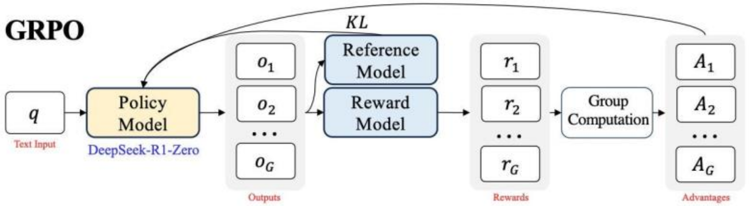
Reinforcement Learning (RL)

Data

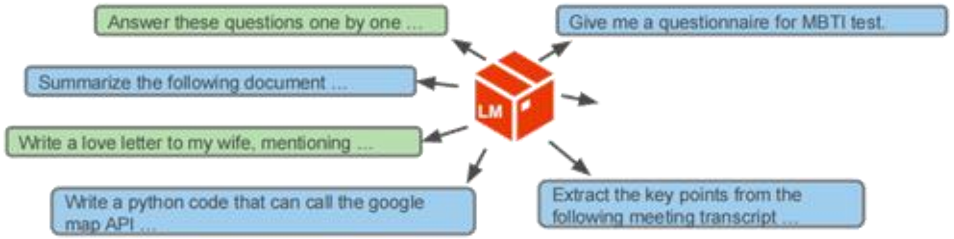
Grader Eval

Efficiency Environment

DeepSeek-Math
[DeepSeek-AI, 2024]



SFT: Trajectory Curation


Answer these questions one by one ...
Give me a questionnaire for MBTI test.
Summarize the following document ...
LM
Write a love letter to my wife, mentioning ...
Write a python code that can call the google map API ...
Extract the key points from the following meeting transcript ...

SFT Goal:

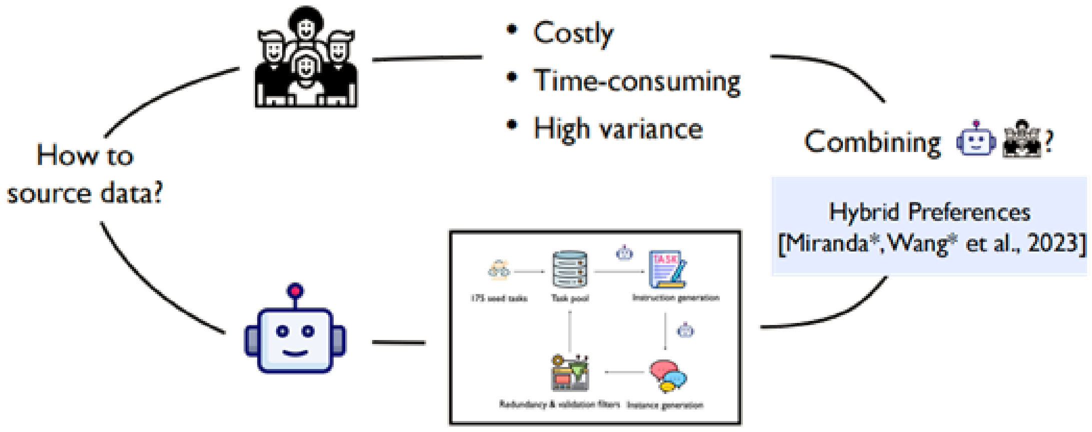
- **Not Knowledge**
- We want LLM **follows user instructions** and **designer's desires (length, bullet points; not knowledge.)**
- Be aware of & Learn how to use **special tokens**: system/user/assistant prompt; [EOS], …

Background:

- data of desired behaviors is what we want **but** <u>scarce and expensive</u>
- pretraining data scales **but** is <u>not what we want</u>
- Idea: finetune pretrained LLM on a little desired data => "post-"training

Two repeated and parallelizable tracks (both inputs and outputs):

1. **Data curation**: Curate data given targeted capabilities
2. **Data mixing**: Mix data across capabilities
   a. <u>Rejection Sampling</u>: Filtering data while maintaining performance.
   b. Start fully with mixing before curation.



Hybrid Preferences [Miranda*, Wang* et al., 2023]

Self-Instruct [Wang et al., ACL 2023]

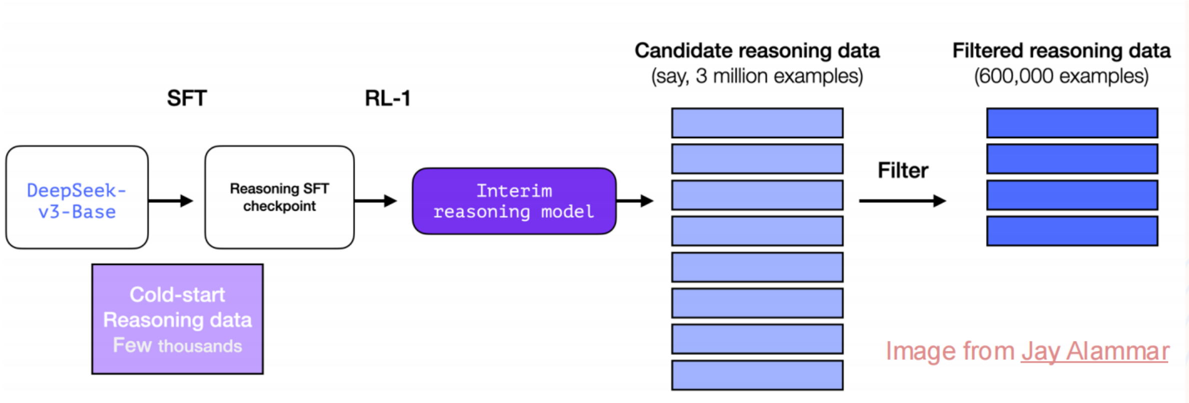
SFT: Data Synthesis Pipeline

DeepSeek-R1

Uses rejection sampling based on **verifiers**

1. Temporary LLM generates many answers

2. Keep answer if it's correct (eg, passes test case), or preferred over others



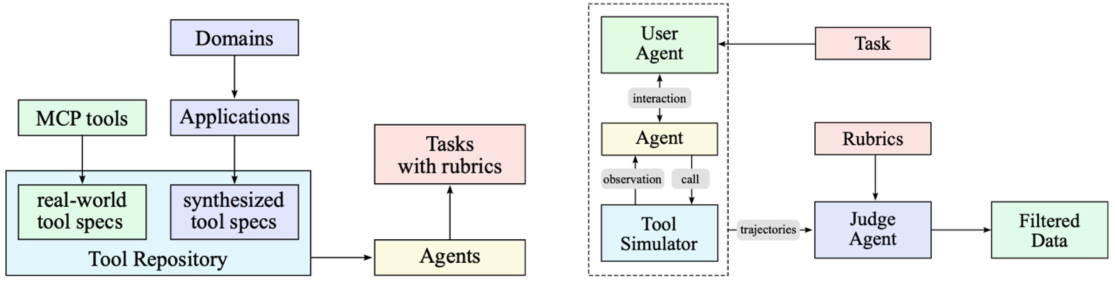
SFT
RL-1
DeepSeek-v3-Base
Reasoning SFT checkpoint
Interim reasoning model
Cold-start Reasoning data Few thousands
Candidate reasoning data (say, 3 million examples)
Filter
Filtered reasoning data (600,000 examples)

Image from Jay Alammar

DeepSeek-R1
[DeepSeek-AI, 2025]

Kimi-K2

Uses LLM simulated user & tools, and **rubric based** rejection sampling to build data for agentic tool use



(a) Synthesizing tool specs, agents and tasks

(b) Generating agent trajectories

Figure 8: Data synthesis pipeline for tool use. (a) Tool specs are from both real-world tools and LLMs; agents and tasks are the generated from the tool repo. (b) Multi-agent pipeline to generate and filter trajectories with tool calling.

Kimi K2
[Kimi team, 2025]

Takeaway:

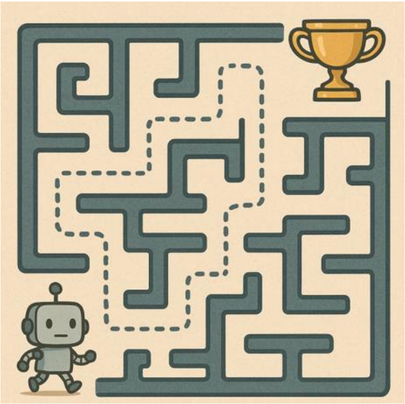
1. Quality Matters: curating high quality data often outperforms alchemy in parameter tuning for the training.

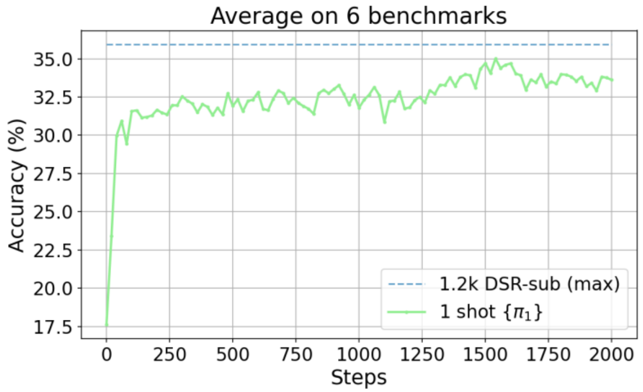
2. Hard problem are usually more useful for powerful models.

3. The goodness of data is also model dependent.

4. Combine the use of real data and synthetic data.

RL: Data





RL Goal:

- **Not just memorization.**
- It is all about **the power of exploration**.

Data Types:

- Verifiable (close-ended): Math, Code, etc. -> Great!
- Non-Verifiable (open-ended) : Style, Writing, Safety, etc. -> Rubrics & Data Synthesis

Takeaway:

1. Extremely high data efficiency
   - 1 sample to figure out most math building blocks
2. Quality matter – requires rejection sampling too
   - **High entropy** to encourage exploration
   - Can't be too hard (pass-rate = 0), or too easy (without negative feedback)

Ranking the training data by the variance of historical training accuracy si.

$$v_i := \mathrm{var}(s_{i,1}, \dots, s_{i,E})$$

$$\pi_j := \pi(j) = \underset{j}{\arg\mathrm{sort}}\{v_i : i \in [N]\}$$

Data/Method	MATH500	AIME24
Base	36.0	6.7
Format reward	65.0	8.3
π_1	74.0	**20.4**
π_{17}	67.2	13.3
1.2k DSR-sub	**75.2**	18.8

RL for RLM with one training example [Wang et. al, 2025]

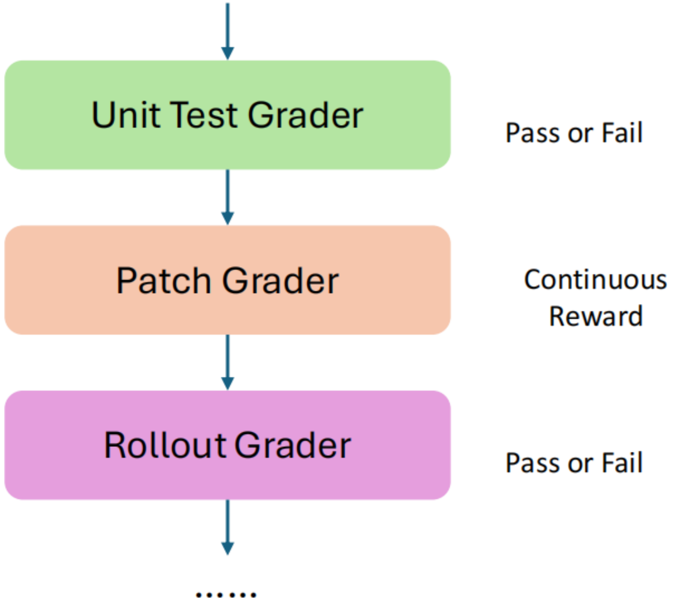
RL: Grade for the reward(signal)

Grading:

- figure out what to grade,
- build up a new grader (update rubrics) to combine with current graders
- prepare the data to be graded and trained (continually evolve, more harder)

Many different graders from (for general Agent)

- Different domains: finance, medical, coding, consultant, Stem, law, etc
- Different problems for each domain
- Different application/product constraints


Unit Test Grader

Pass or Fail

Patch Grader

Continuous Reward

Rollout Grader

Pass or Fail

……

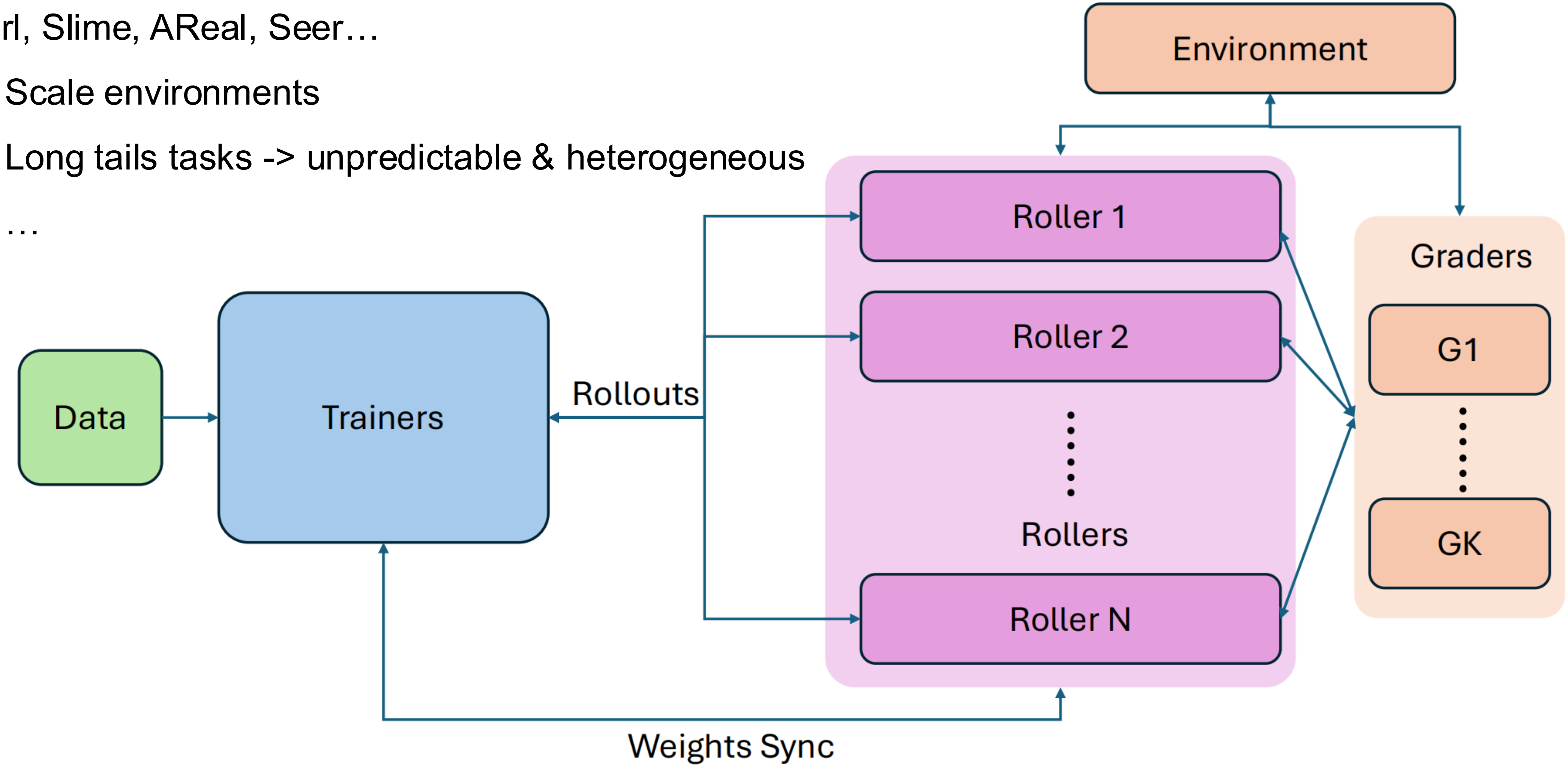
Other Things important for (Asynchronous) Agentic RL Training:

1. **Efficiency:**
   - Encourage policy exploration: [Agent Training often encounters entropy collapse – **void turns & echo trap]**
   - Control sampling cost
2. **Infra for asynchronous Agentic RL**
3. **Environment**
4. **Evaluation under open-ended setting**
5. **…**

RL: Asynchronous Infra

Verl, Slime, AReal, Seer…

- Scale environments
- Long tails tasks -> unpredictable & heterogeneous
- …



Outline

1. Overview

2. Post-training: How can we build a reliable LLM Agent

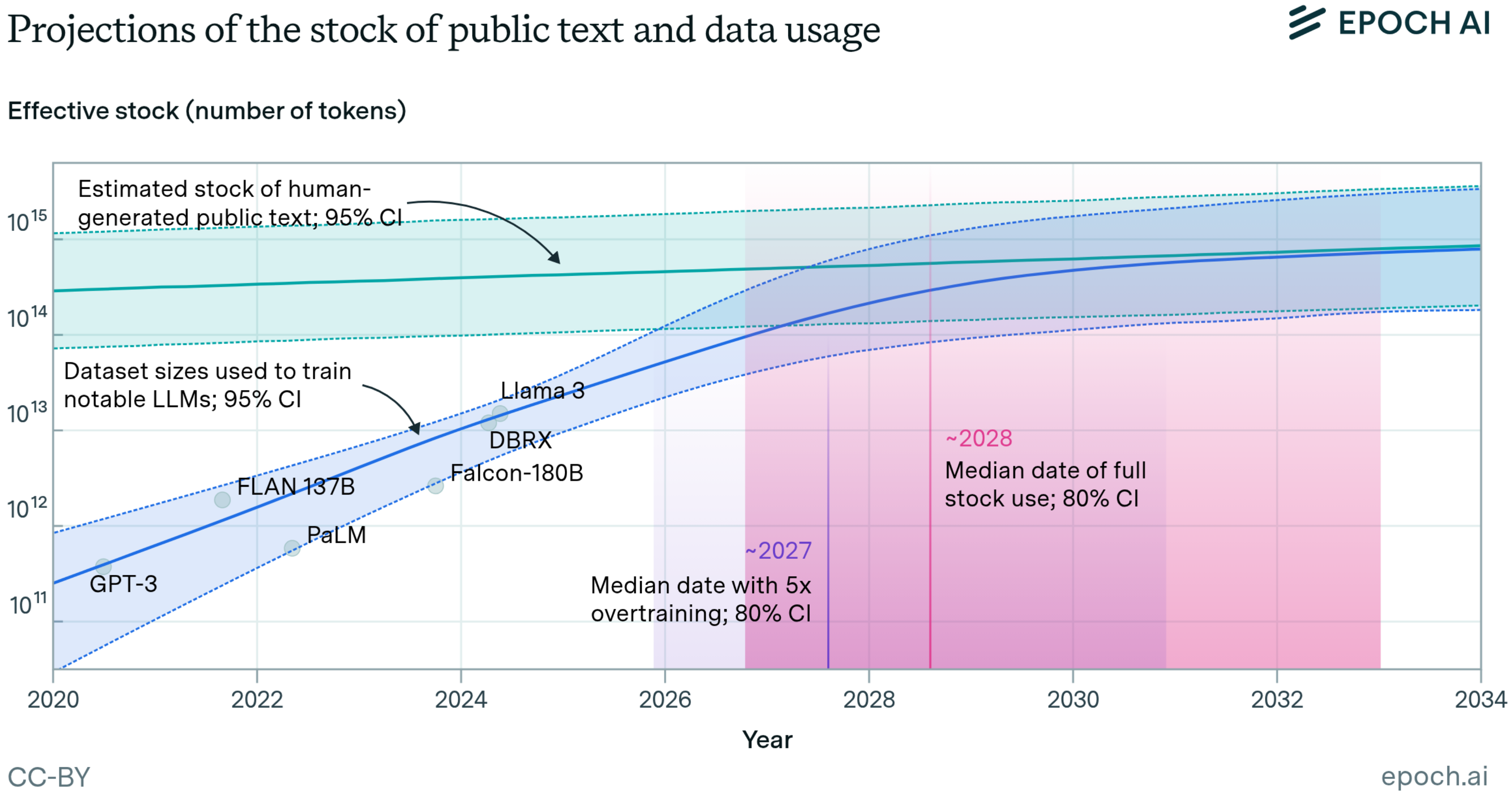
3. Self-Evolving: How should we continue to raise the ceiling of LLM Agent

- Why Self-evolving & What is Self-evolving
- Two perspectives and their developments

Why Self-evolving AI

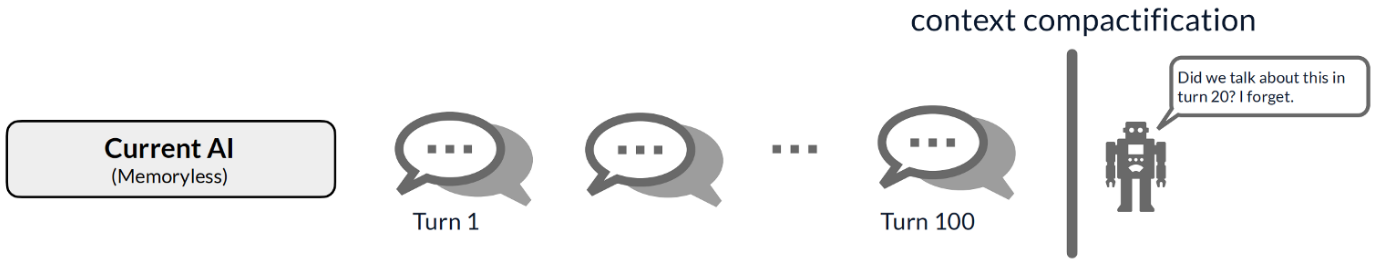
1. Scaling under finite human data

Will we run out of data? Limits of LLM scaling based on human-generated data [Epoch-AI, 2024]

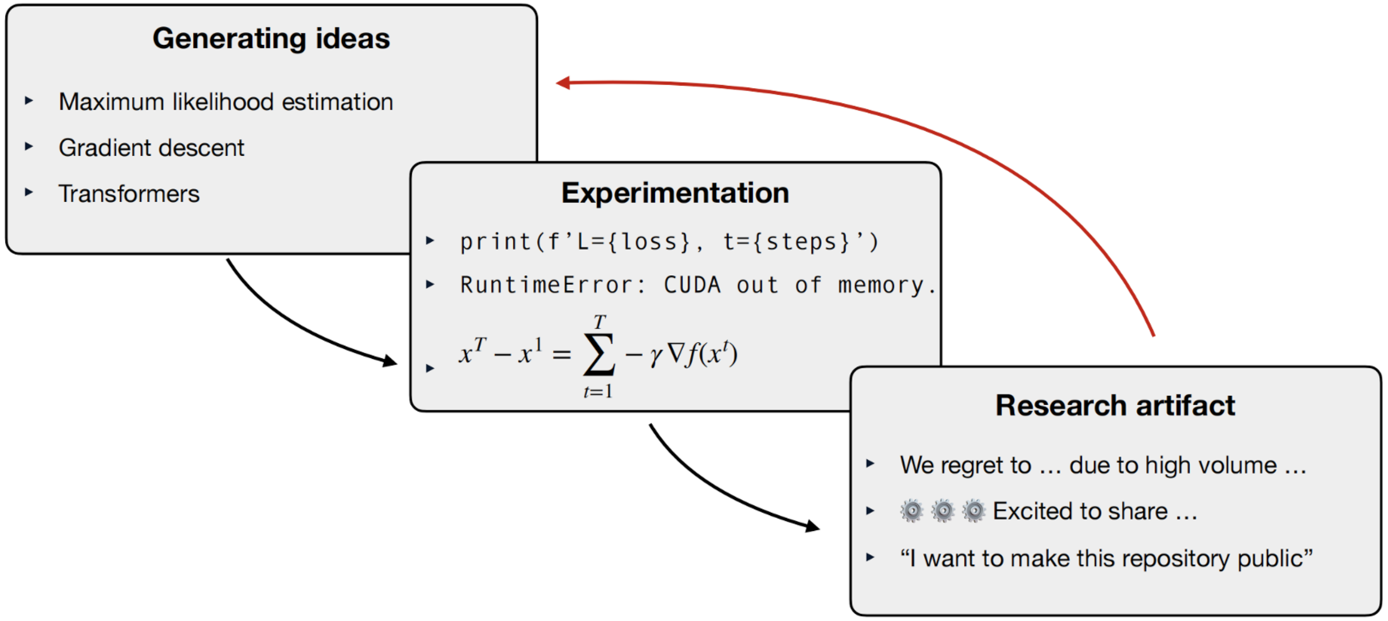


Why Self-evolving AI

2. Static weights after human creation



3. Limited by algorithms human can find



Generating ideas

- Maximum likelihood estimation
- Gradient descent
- Transformers

Experimentation

- `print(f'L={loss}, t={steps}')`
- `RuntimeError: CUDA out of memory.`
- $x^T - x^1 = \sum_{t=1}^{T} -\gamma \nabla f(x^t)$

Research artifact

- We regret to … due to high volume …
- ⚙ ⚙ ⚙ Excited to share …
- “I want to make this repository public”

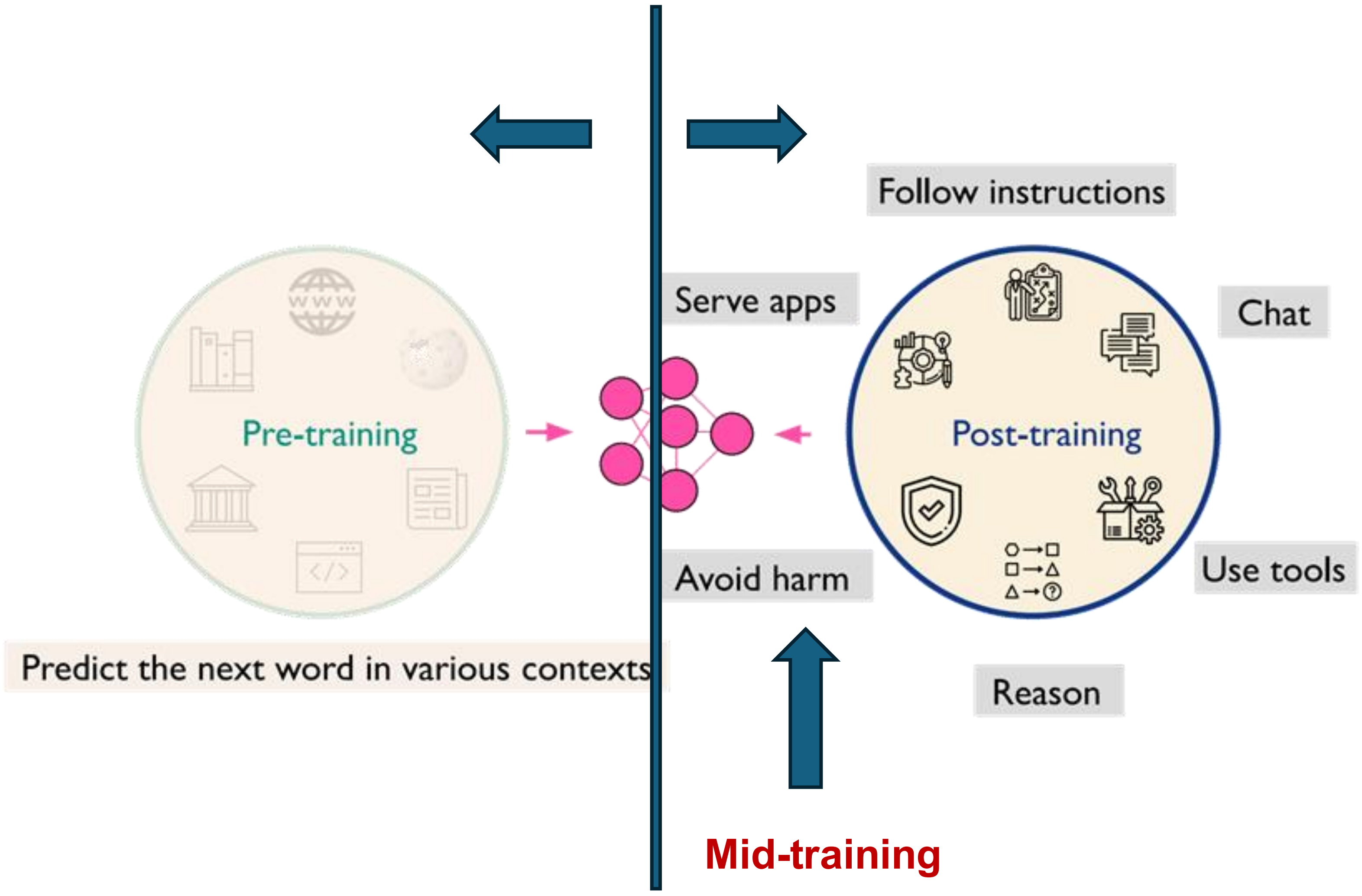
What is Self-evolving AI - Definition

A continually self-evolving AI is one that, once created, can autonomously and continually improve itself better than its human creators can improve it.

What is Self-evolving AI - Characteristics

Three characteristics of self-evolving AI under parametric and pretrained assumption:

1. After the initial pretraining phase, the system continues to acquire new knowledge into its parametric weights without catastrophically forgetting existing capabilities.
2. The system generates its own `training_signal`, and learning from these self-generated signals yields improvement beyond what human-generated signals provide.
3. The system can autonomously design what `learning_algorithm` to use to learn from its training signals.



Pre-training

Predict the next word in various contexts

Serve apps

Avoid harm

Follow instructions

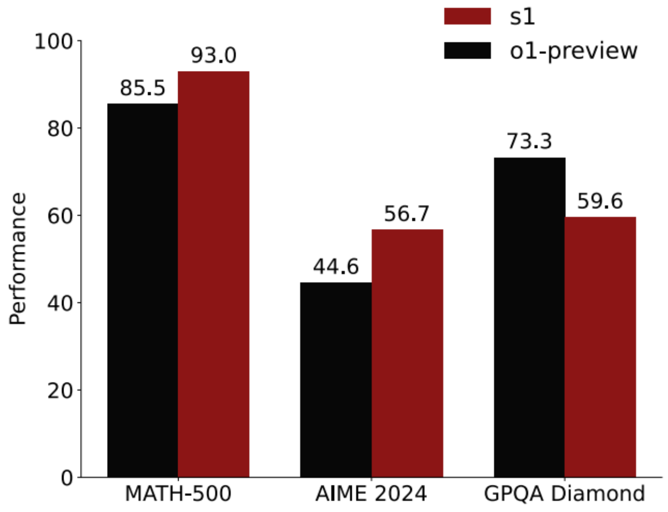
Post-training

Chat

Use tools

Reason

Mid-training



S1: Simple test-time scaling
[Muennighoff et. al, 2025]

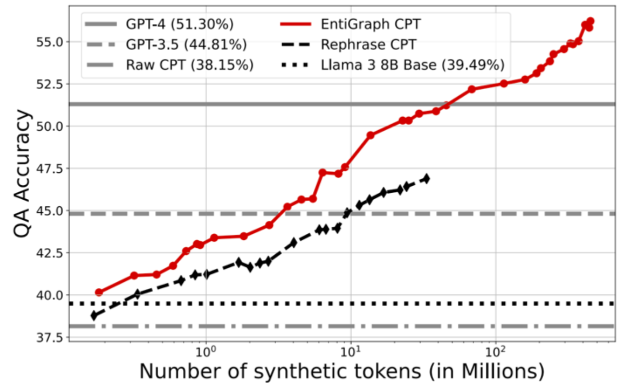
Perspective 1: pre-training is the core

1. Continual Knowledge acquisition for continual pre-training

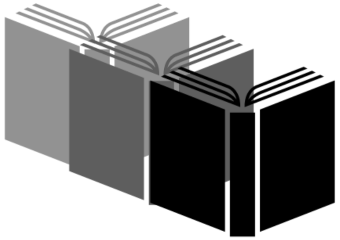
Goal: teach model the knowledge from a niche domain consisted of a few “source documents”.

Step 1: Generate synthetic text based on the source documents

Step 2: Continually pretrain (finetune) the model on generated text

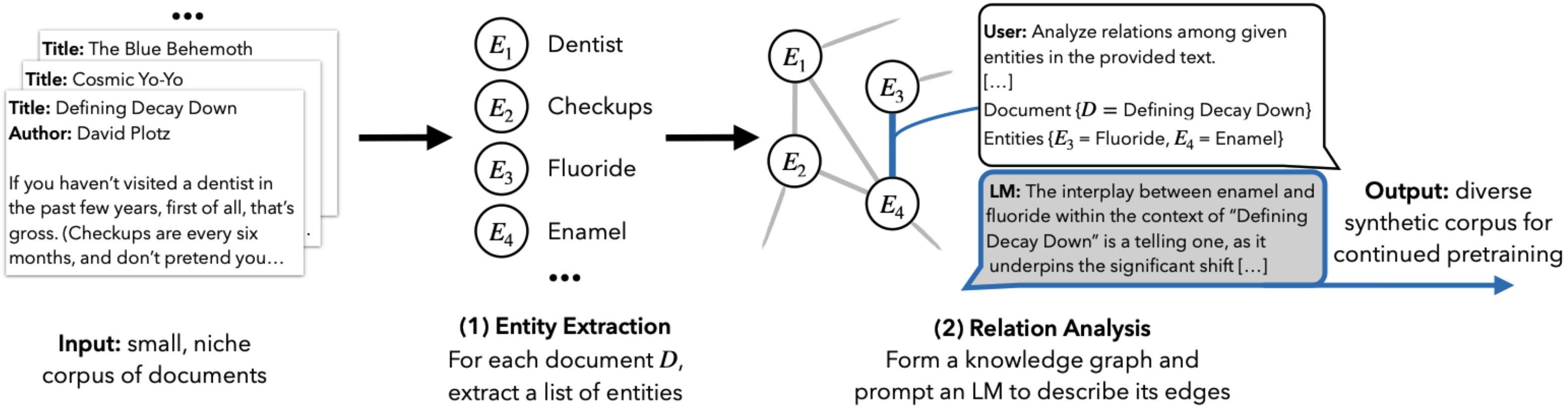


QuALITY Books



- Project Gutenberg fictions (mainly science fiction)
- Slate magazine articles
- The Long and Short, Freesouls, etc

265 *specialized* books (~1.8M tokens);
High-quality multiple-choice Q&As



Title: The Blue Behemoth

Title: Cosmic Yo-Yo

Title: Defining Decay Down
Author: David Plotz

If you haven’t visited a dentist in the past few years, first of all, that’s gross. (Checkups are every six months, and don’t pretend you…

Input: small, niche corpus of documents

E_1 Dentist
E_2 Checkups
E_3 Fluoride
E_4 Enamel

(1) Entity Extraction
For each document D, extract a list of entities

User: Analyze relations among given entities in the provided text.
[…]
Document $\{D$ = Defining Decay Down$\}$
Entities $\{E_3$ = Fluoride, E_4 = Enamel$\}$

LM: The interplay between enamel and fluoride within the context of “Defining Decay Down” is a telling one, as it underpins the significant shift […]

(2) Relation Analysis
Form a knowledge graph and prompt an LM to describe its edges

Output: diverse synthetic corpus for continued pretraining

QuALITY
[Pang et. al, 2022]

Synthetic Continued Pretraining
[Yang et. al, 2024]

Perspective 1: pre-training is the core

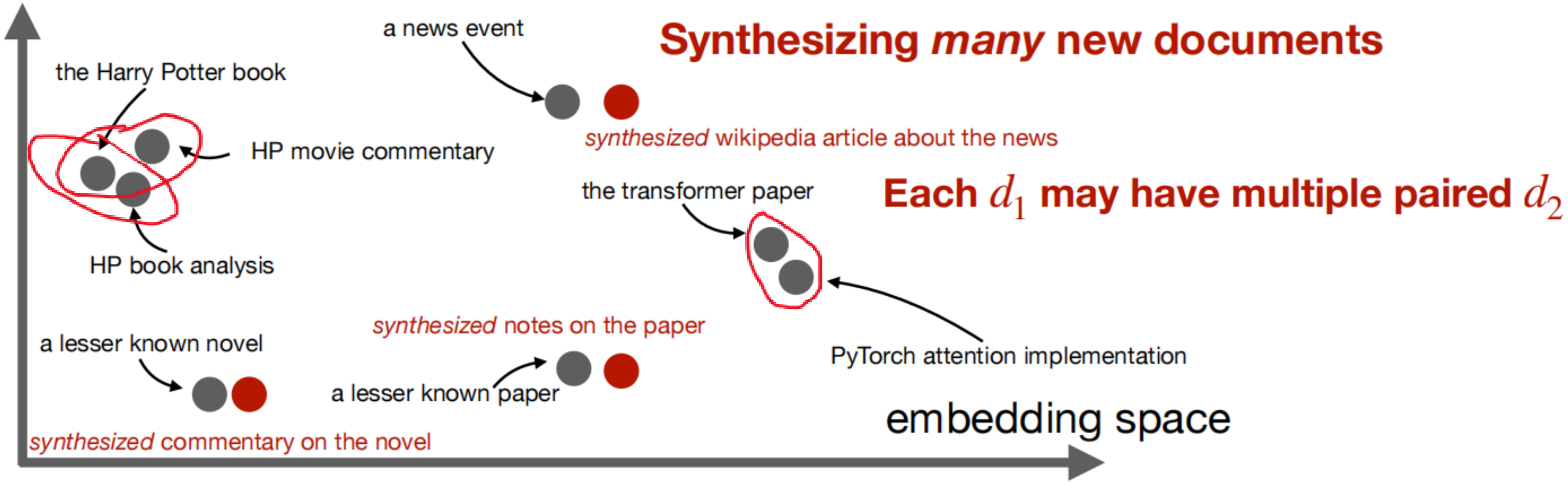
2. Self-improving pretraining capability – Learn structural correlation from scratch

Goal: Self-improving with no distillation from teacher model

Step 1: Nearest-neighbor pairing: Use DCLM subset and Qwen-0.6B-Embedding

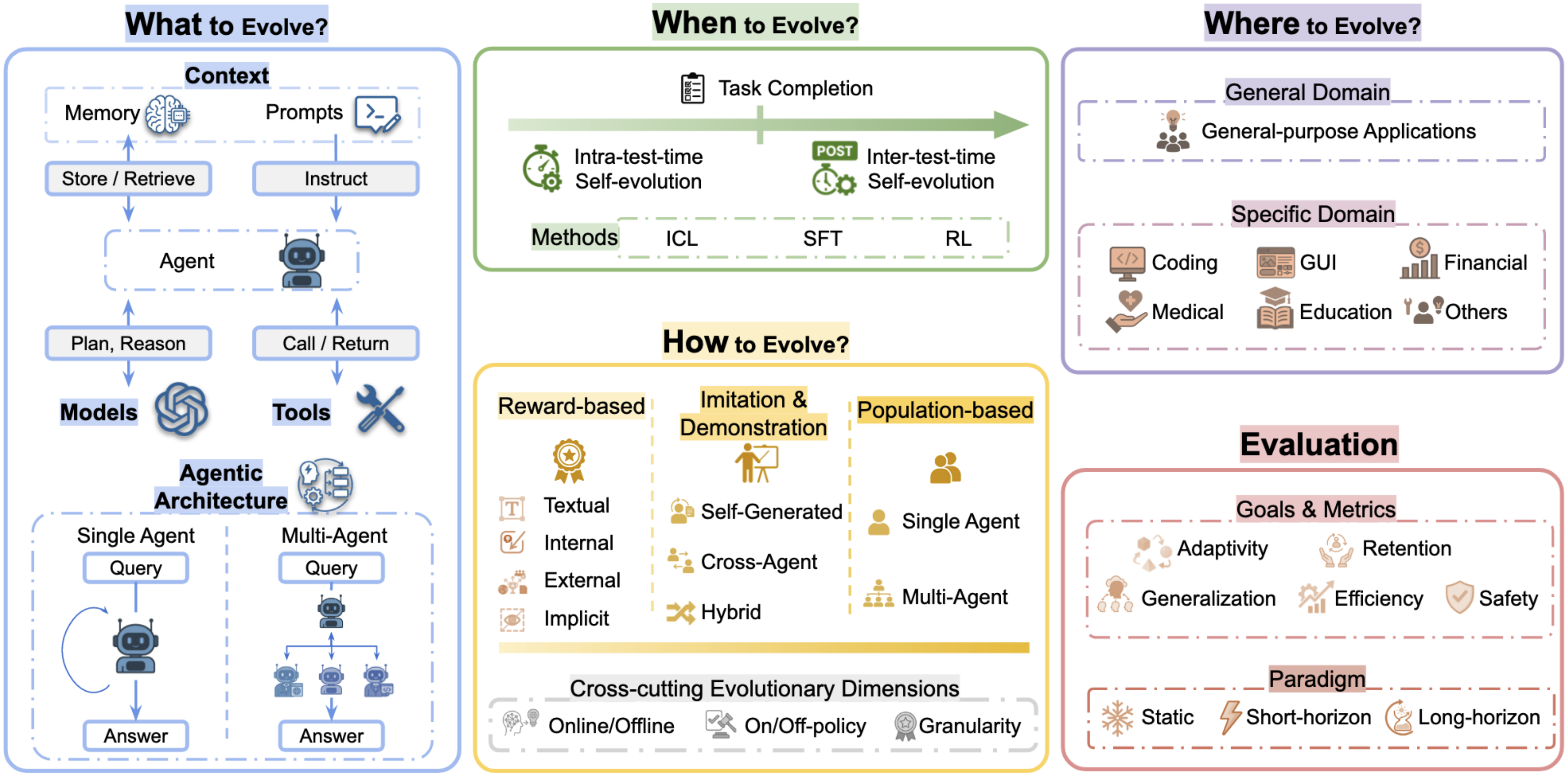
Step 2: Synthesizer tuning: SFT-like objective $p_\theta(d_1|d_2)$ initialized at pretrained checkpoint

Step 3: Synthesis at scale: Temperature=1 allows each document to have varied synthesis


a news event
Synthesizing *many* new documents
the Harry Potter book
HP movie commentary
synthesized wikipedia article about the news
the transformer paper
Each d_1 may have multiple paired d_2
HP book analysis
synthesized notes on the paper
a lesser known novel
PyTorch attention implementation
a lesser known paper
embedding space
synthesized commentary on the novel

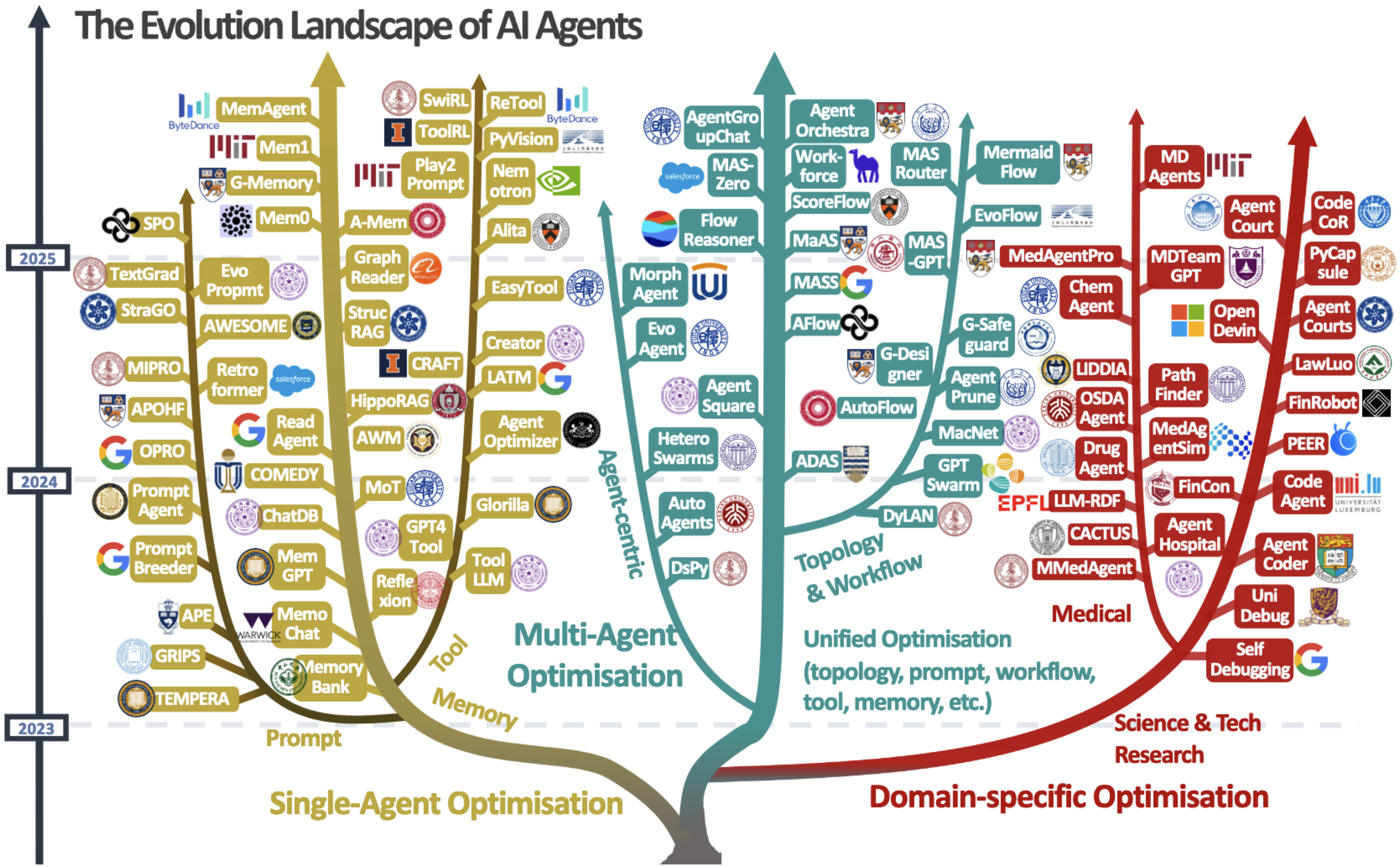
Synthetic bootstrapped pretraining
[Yang et. al, 2025]

Perspective 2: mid and post-training & Agent scaffold components for adaptation



A SURVEY OF SELF-EVOLVING AGENTS: ON PATH TO ARTIFICIAL SUPER INTELLIGENCE
[Gao et. al, 2025]

Perspective 2: mid and post-training & Agent scaffold components for adaptation



A Comprehensive Survey of Self-Evolving AI Agents
[Fang et. al, 2025]

Look Ahead: Can AI self-evolve to be *really* stronger than its creator?

AutoResearchClaw

Chat an Idea. Get a Paper. Fully Autonomous.

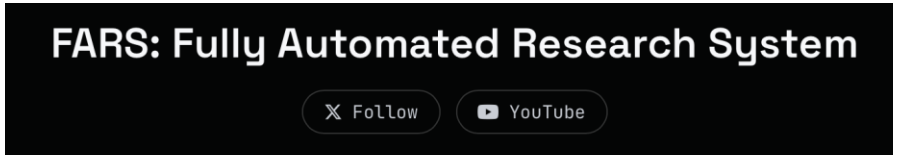
Auto-claude-code-research-in-sleep (ARIS ⚔️)

ARIS

Adversarial Research in Sleep

Claude Code × GPT-5.4 xhigh

speed × rigor



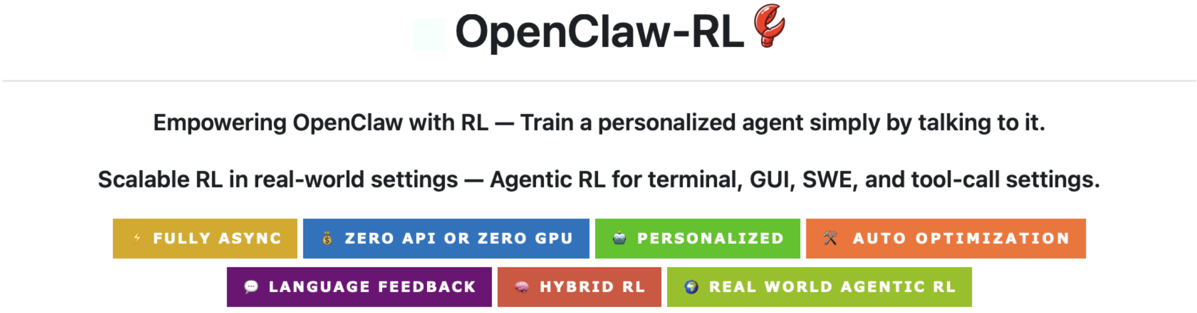
FARS: Fully Automated Research System

166 Papers

21,600,000,000 Tokens

417 Hours

$186,000 Total Cost



OpenClaw-RL 🦞

Empowering OpenClaw with RL — Train a personalized agent simply by talking to it.

Scalable RL in real-world settings — Agentic RL for terminal, GUI, SWE, and tool-call settings.

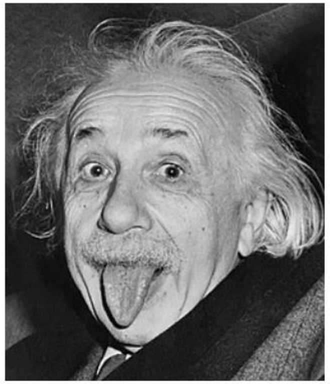
MetaClaw

Just talk to your agent — it learns and *EVOLVES*.

Inspired by how the brain learns. Meta-learn and evolve your 🦞 from every conversation in the wild. No GPU required. Works with Kimi, Qwen, Claude, MiniMax, and more.

Look Ahead: Can AI self-evolve to be *really* stronger than its creator?

- At present, the underlying technologies for neither perspective have completely hit the mark.



Albert Einstein

$$< \qquad G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$$

- A theory can evolve and can mutate. It has a life of their own.

- Einstein created the field equations that is smarter than himself.

- By analogy, humans can create AI that is smarter than humans themselves.

The moment the theory is created, it is above its creator.

Think Boldly, Act Bravely

References:

1. Agentic AI, UC Berkeley, https://agenticai-learning.org/f25
2. Advanced Large Language Model Agents, UC Berkeley, https://agenticai-learning.org/sp25
3. Continually self-improving AI, Zitong Yang, Stanford University
4. Lambert, Nathan, et al. "Tulu 3: Pushing frontiers in open language model post-training." *arXiv preprint arXiv:2411.15124* (2024).
5. Guo, Daya, et al. "Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning." *arXiv preprint arXiv:2501.12948* (2025).
6. Team, Kimi, et al. "Kimi k2: Open agentic intelligence." *arXiv preprint arXiv:2507.20534* (2025).
7. Wang, Yiping, et al. "Reinforcement learning for reasoning in large language models with one training example." *arXiv preprint arXiv:2504.20571* (2025).
8. Villalobos, Pablo, et al. "Position: Will we run out of data? Limits of LLM scaling based on human-generated data." *Forty-first International Conference on Machine Learning*. 2024.
9. Muennighoff, Niklas, et al. "s1: Simple test-time scaling." *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 2025.
10. Pang, Richard Yuanzhe, et al. "QuALITY: Question answering with long input texts, yes!." *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2022.
11. Yang, Zitong, et al. "Synthetic continued pretraining." *arXiv preprint arXiv:2409.07431* (2024).
12. Yang, Zitong, et al. "Synthetic bootstrapped pretraining." *arXiv preprint arXiv:2509.15248* (2025).
13. Gao, Huan-ang, et al. "A survey of self-evolving agents: On path to artificial super intelligence." *arXiv preprint arXiv:2507.21046* 1 (2025).
14. Fang, Jinyuan, et al. "A comprehensive survey of self-evolving ai agents: A new paradigm bridging foundation models and lifelong agentic systems." *arXiv preprint arXiv:2508.07407*(2025).

Hope you enjoyed the lecture!

Question?