Reinforcement Learning for Large Language Models: From Fine-Tuning to Finer-Tuning to Auxiliary Policy Models

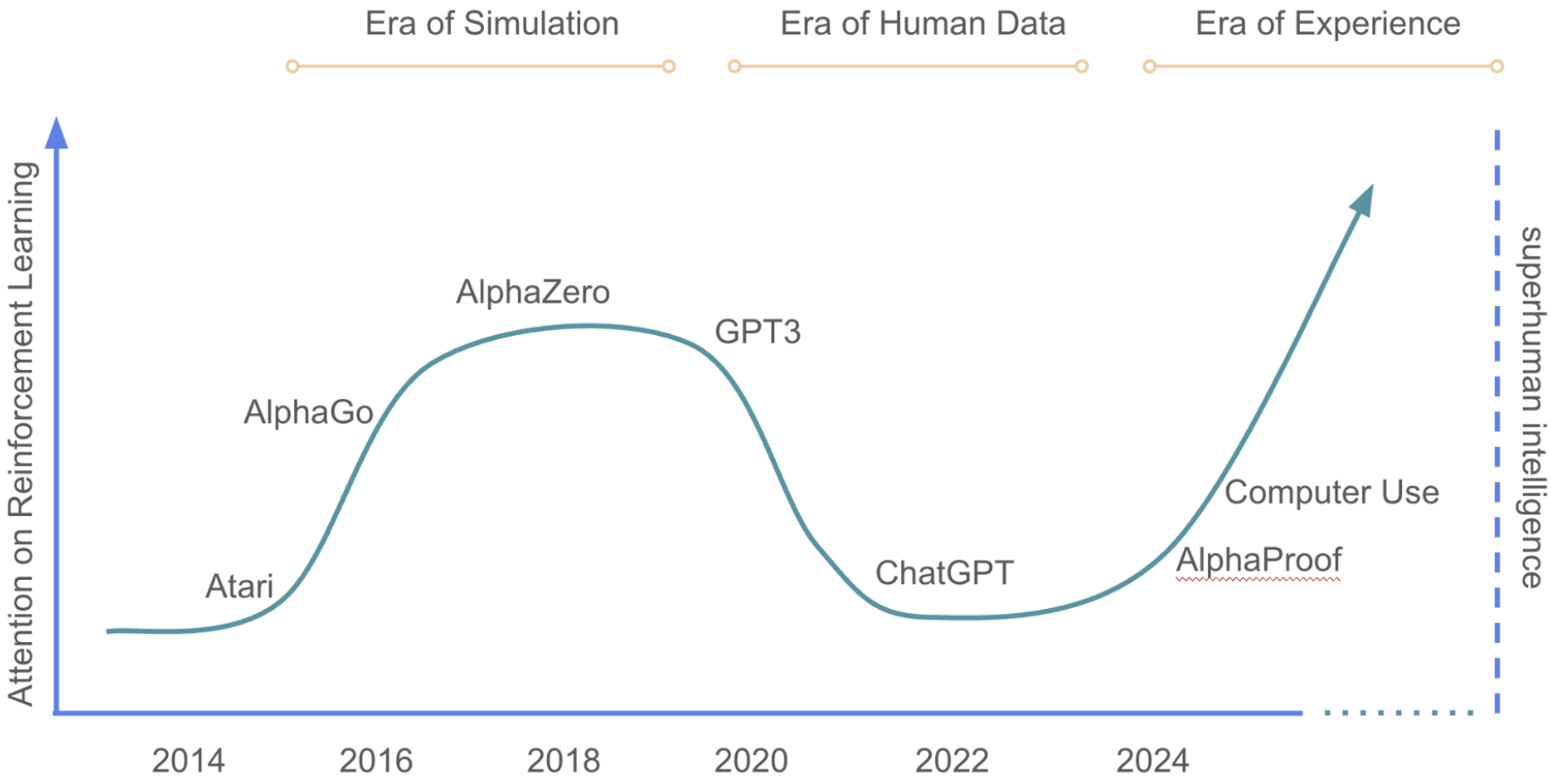
Haipeng Chen

Data Driven Decision Intelligence (D3i) Lab

William & Mary

WILLIAM & MARY | DATA SCIENCE

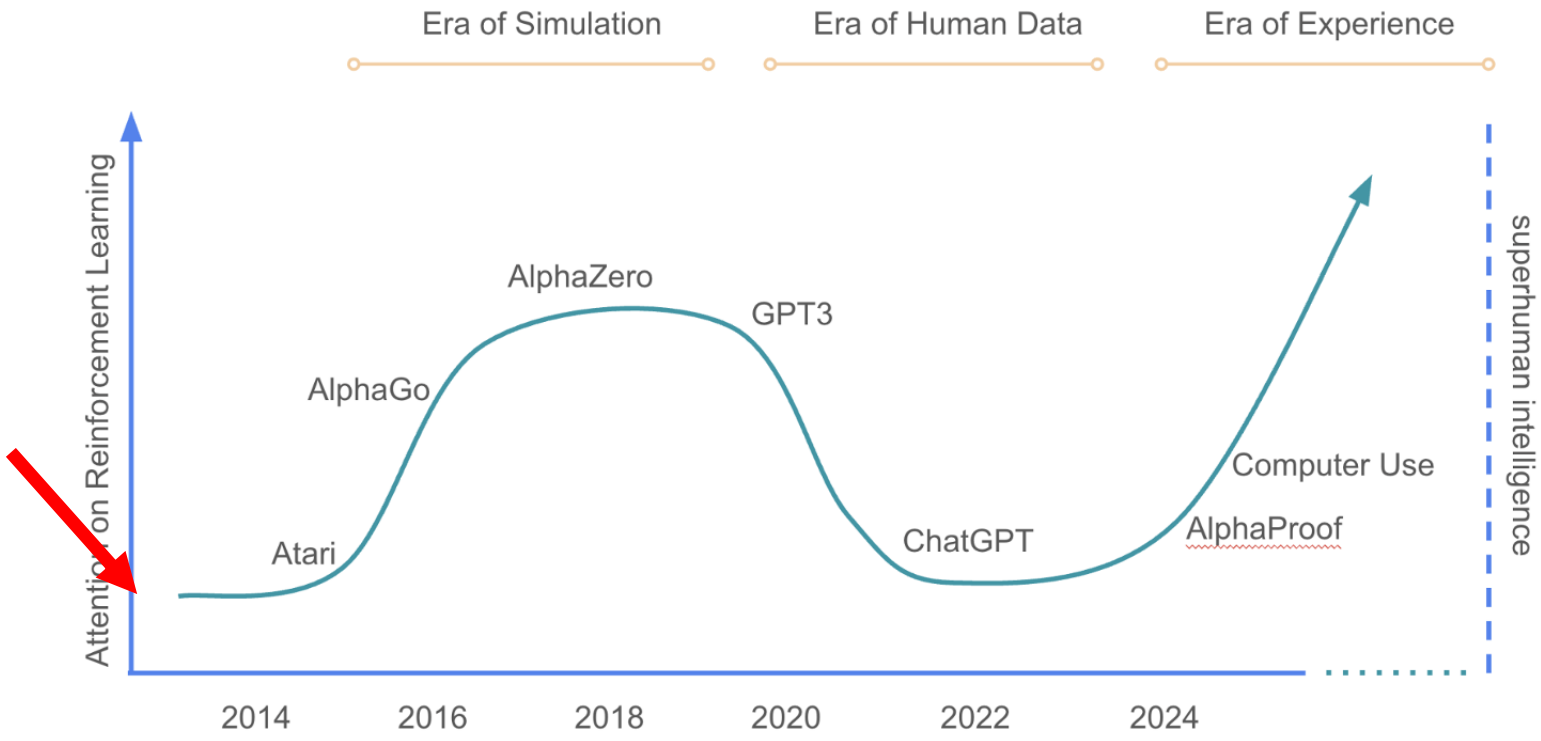
Welcome to the Era of Experience



http://incompleteideas.net/papers/TheEraOfExperience.pdf

W&M DATA SCIENCE

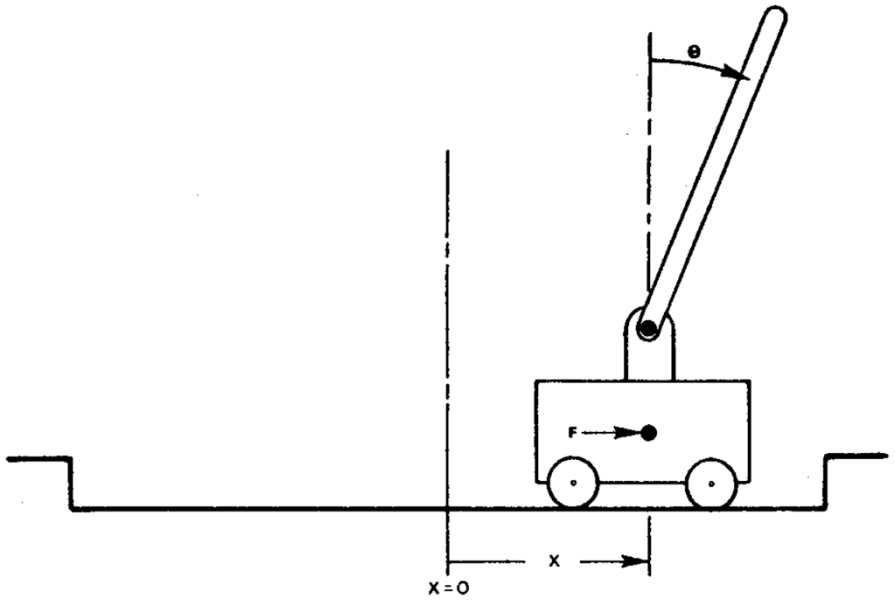
Working on RL is like riding a roller coaster…

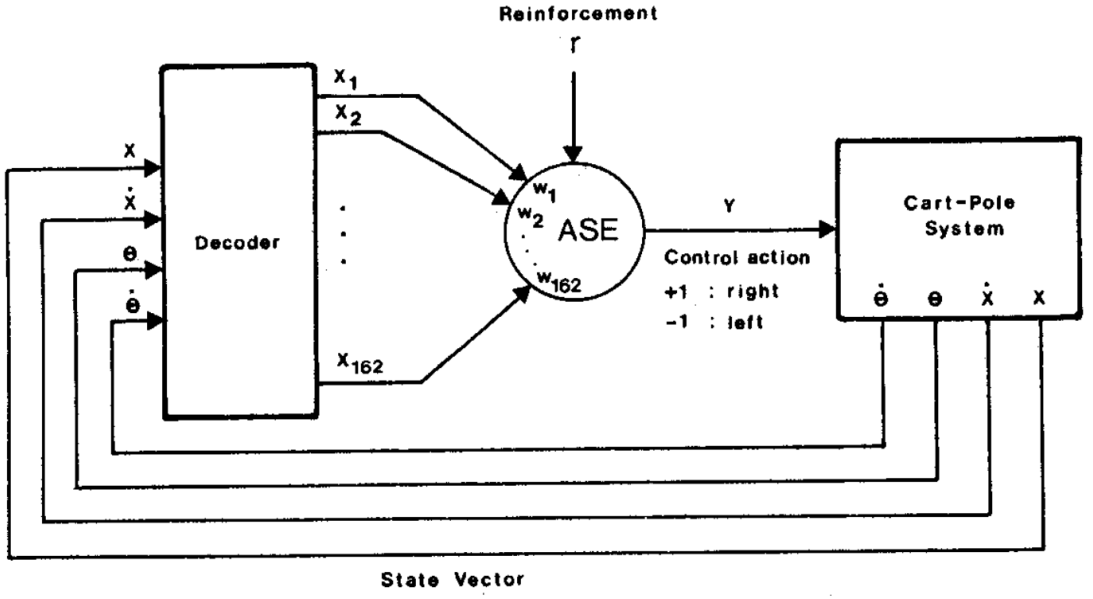


http://incompleteideas.net/papers/TheEraOfExperience.pdf

RL in the early days…

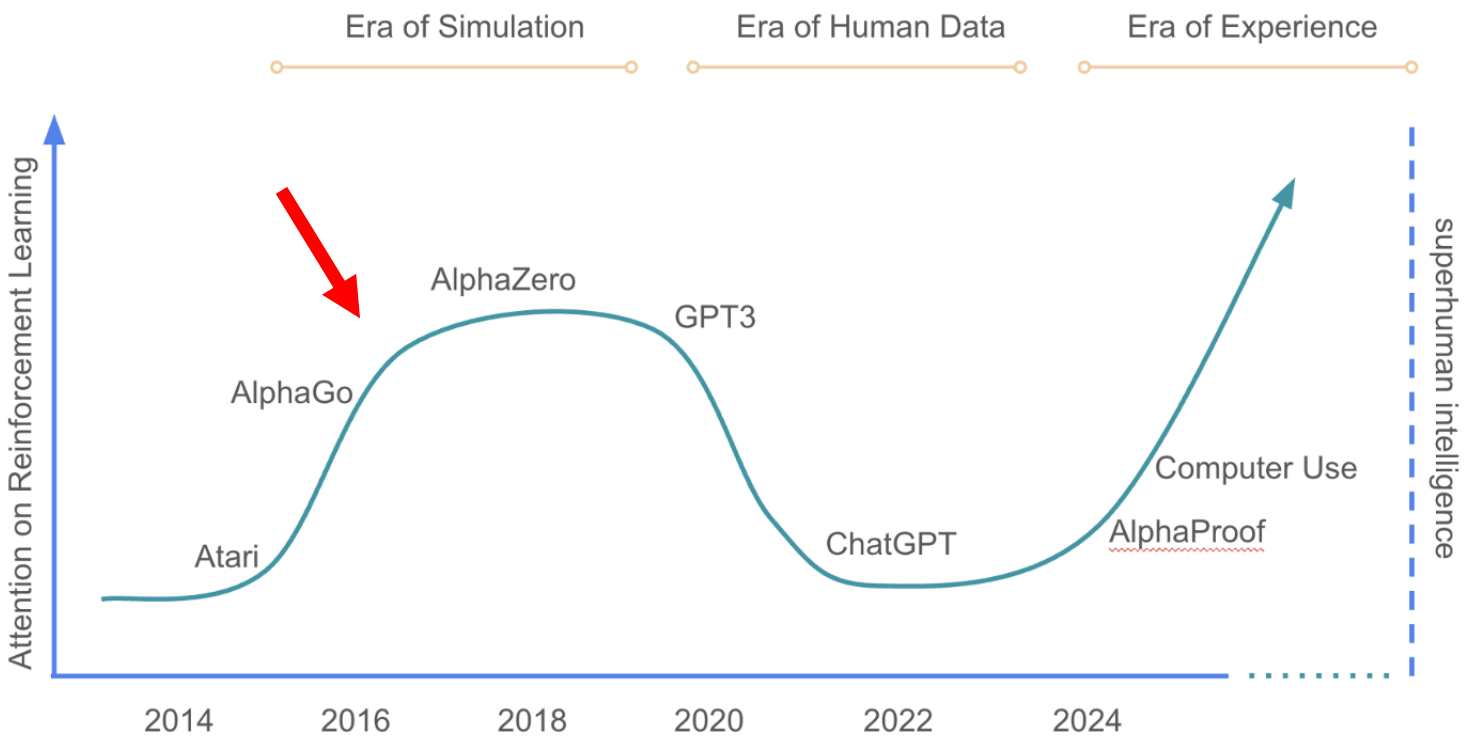
$$y(t) = f\left[\sum_{i=1}^{n} w_i(t)x_i(t) + \text{noise}(t)\right]$$





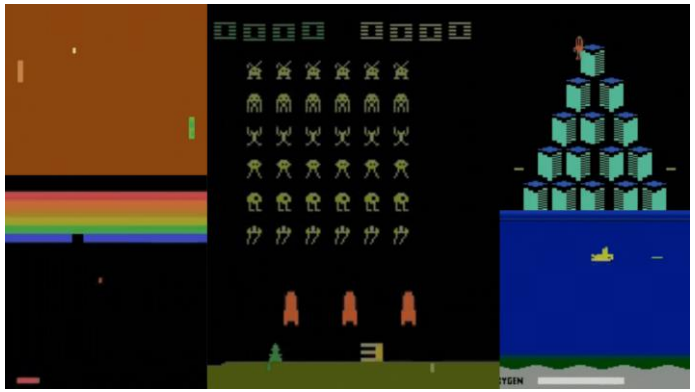
Barto, Sutton, Anderson, Neuronlike adaptive elements that can solve difficult learning control problems, IEEE Transactions on Systems, Man, and Cybernetics, 1983.

The RL roller coaster…



http://incompleteideas.net/papers/TheEraOfExperience.pdf

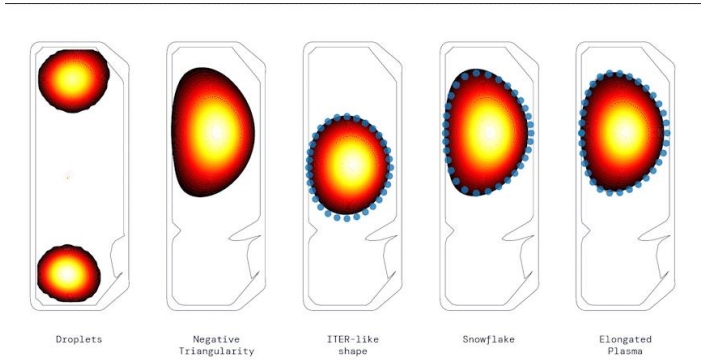
RL successes before LLMs



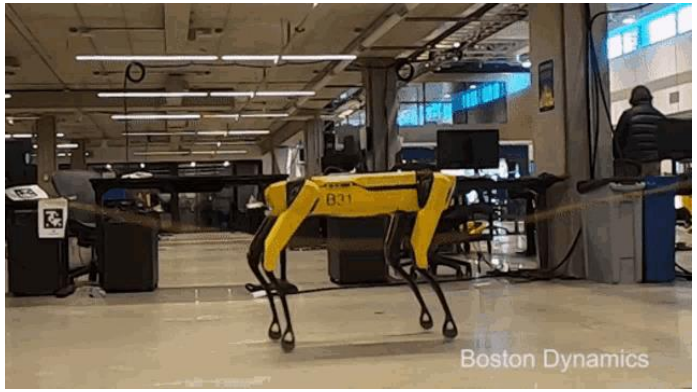
Atari games (DQN)

Computer Go

Data center cooling

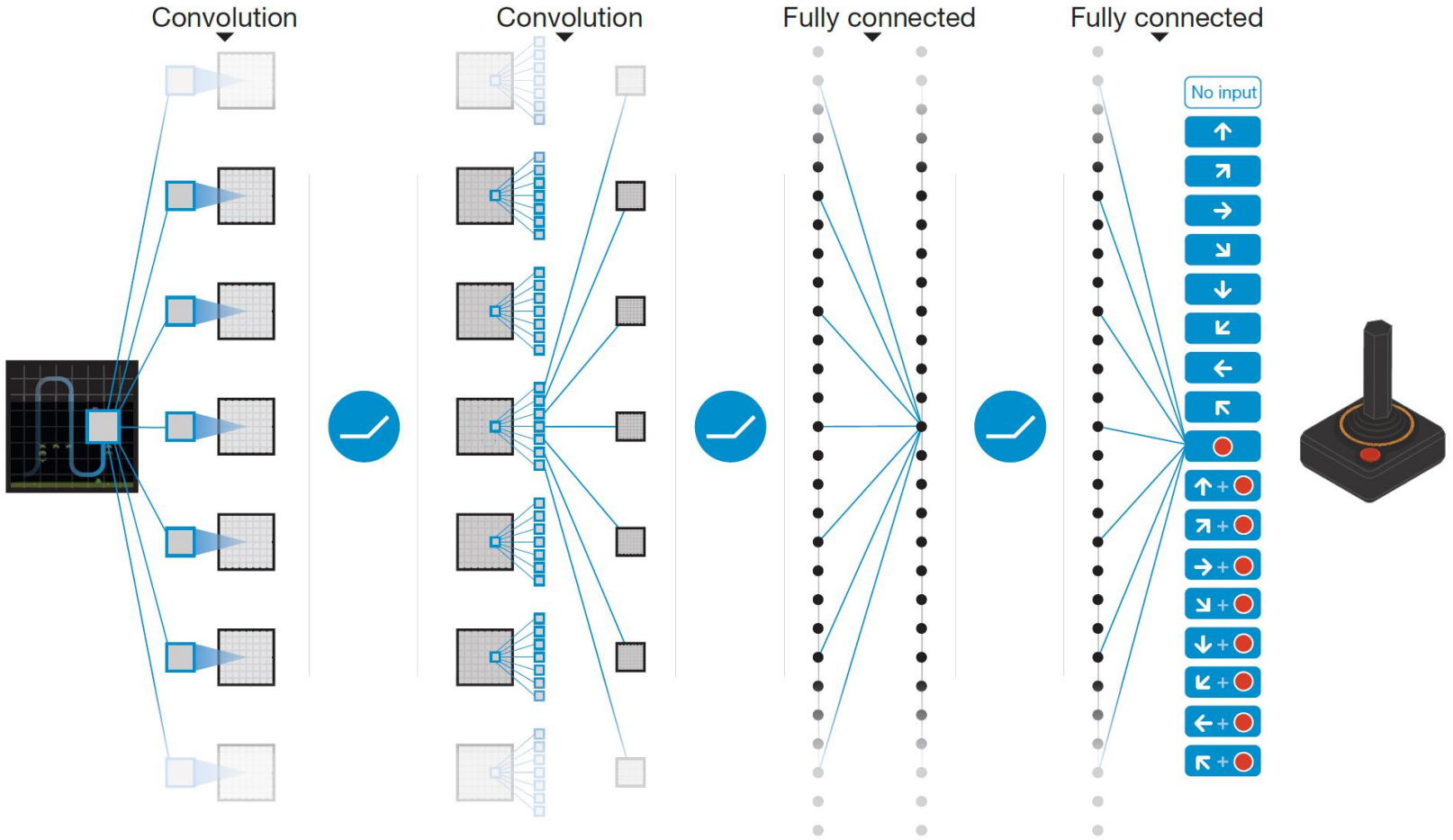


Nuclear fusion control



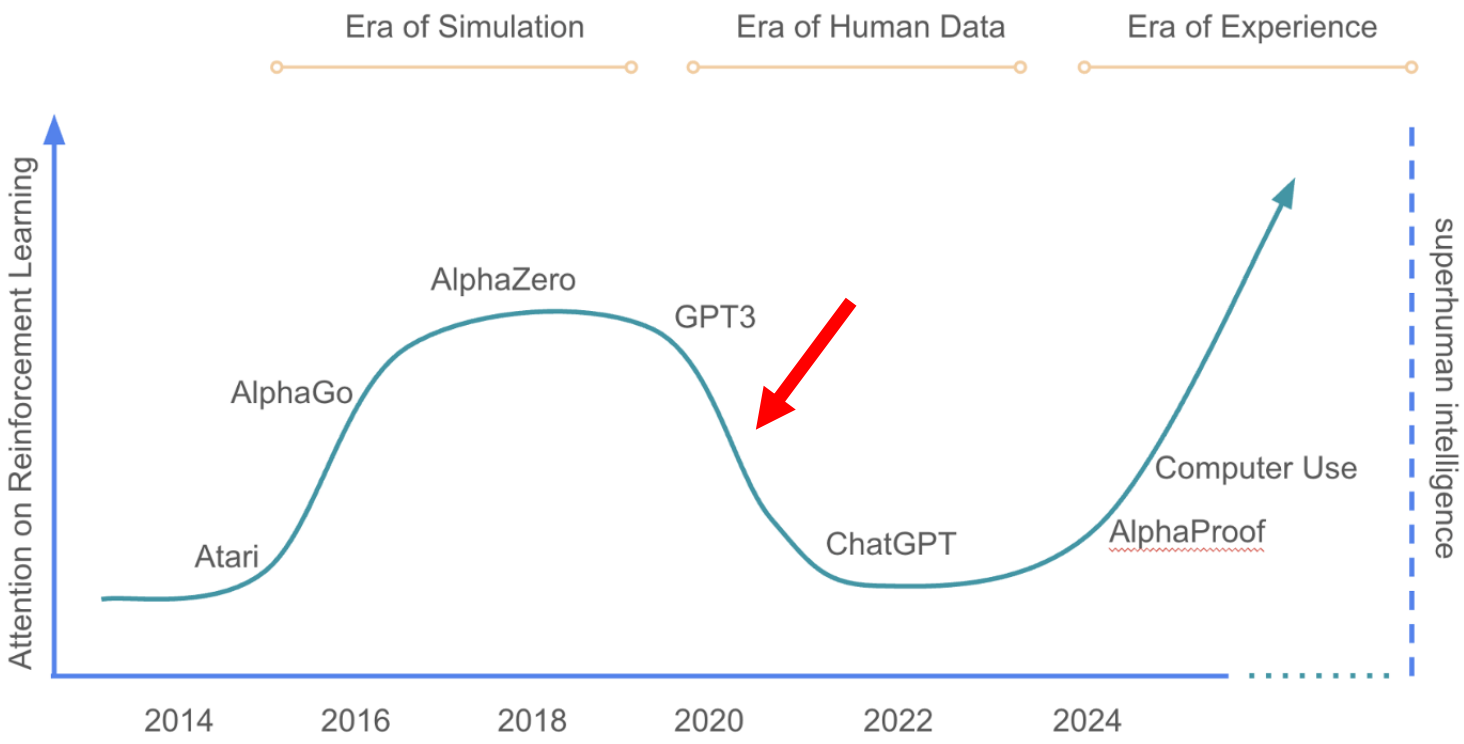
Robotic control

“Deep” RL



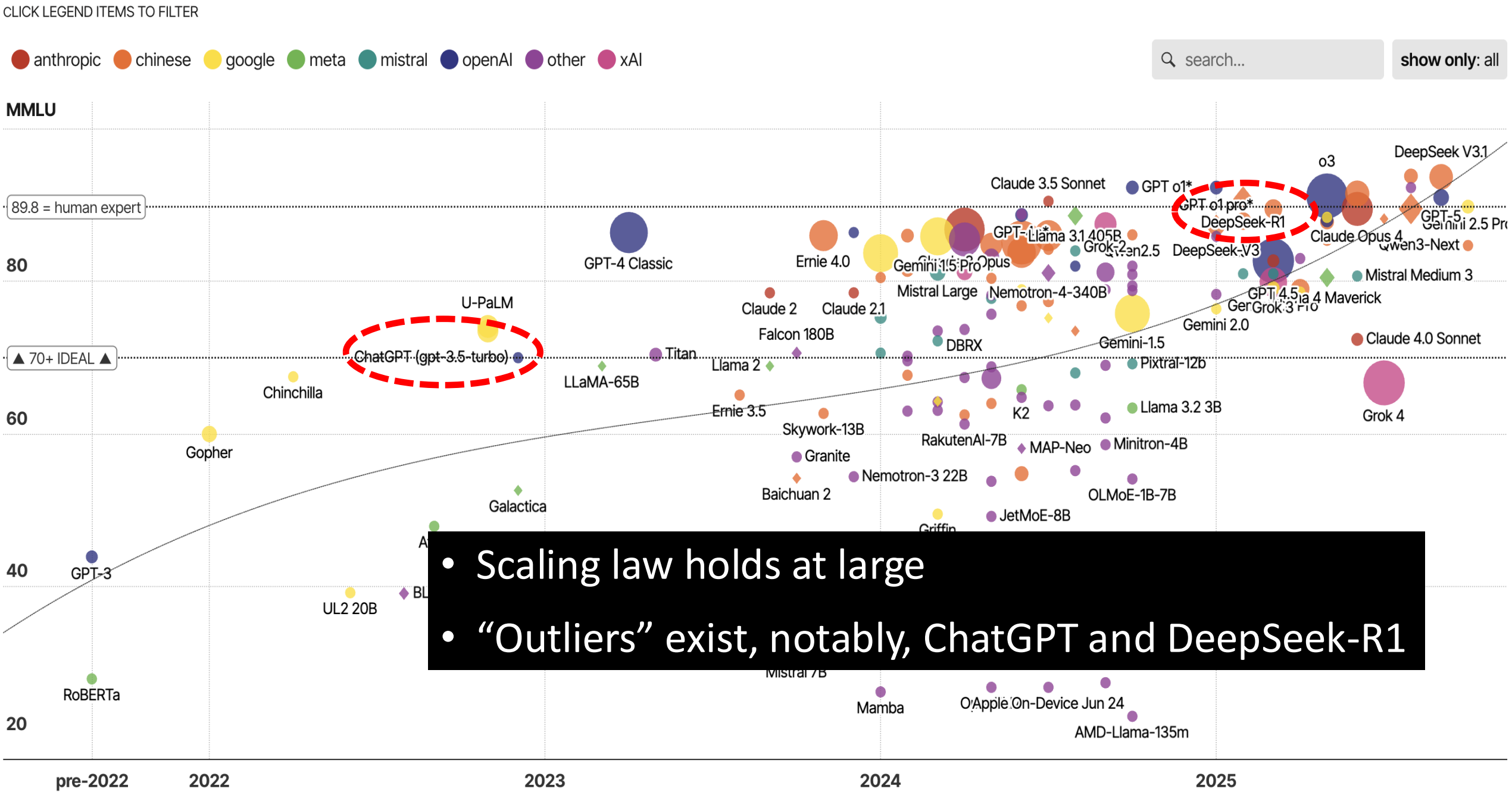
Minh et al, Human-level control through deep reinforcement learning, Nature, 2015.

The RL roller coaster…



- Data-hungry
- reward-sensitive
- hard to stabilize, and
- just don’t work …

http://incompleteideas.net/papers/TheEraOfExperience.pdf



David McCandless, Tom Evans, Paul Barton
Informationisbeautiful // Sep 2025

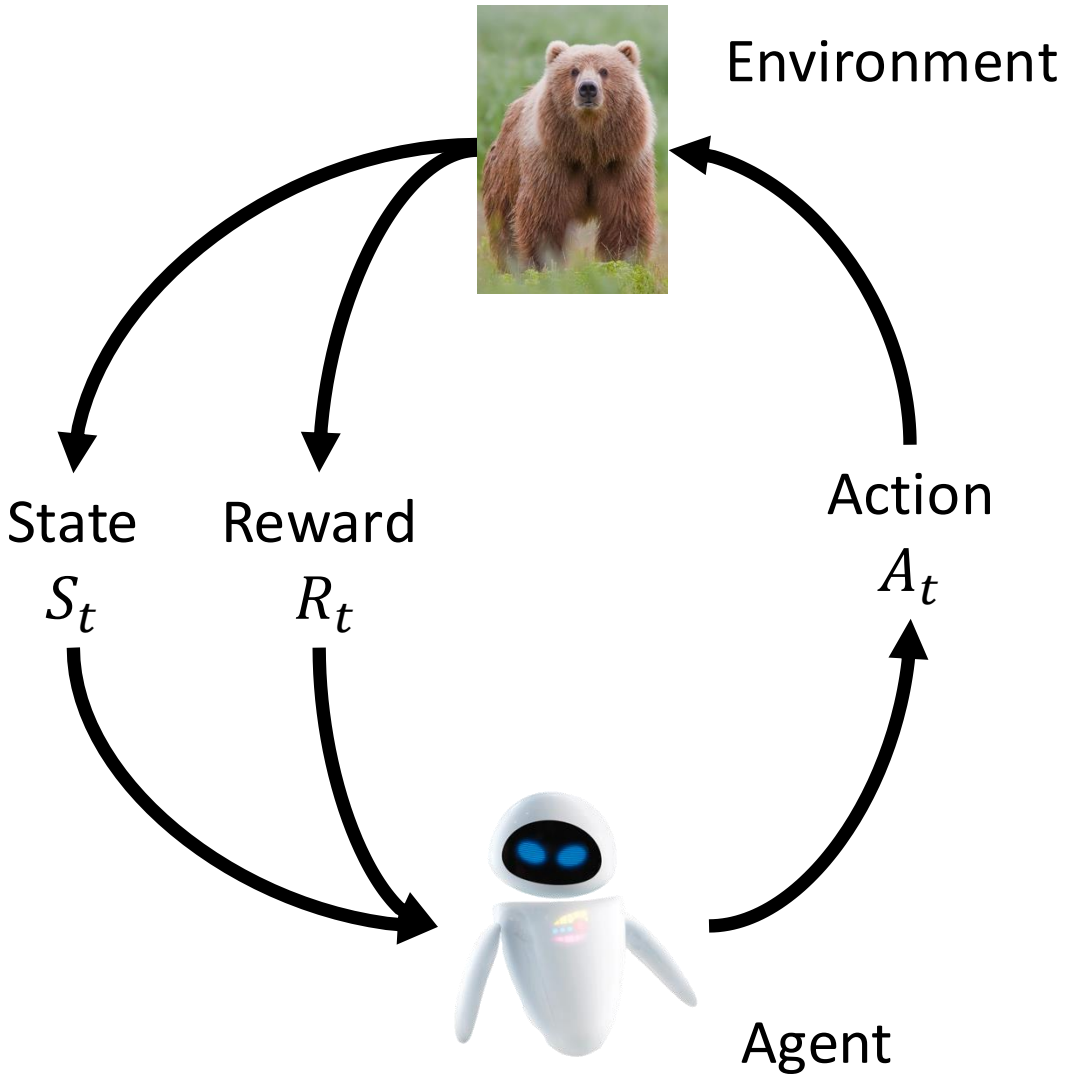
MMLU = benchmark for measuring LLM capabilities
* = parameters undisclosed // source: LifeArchitect // data

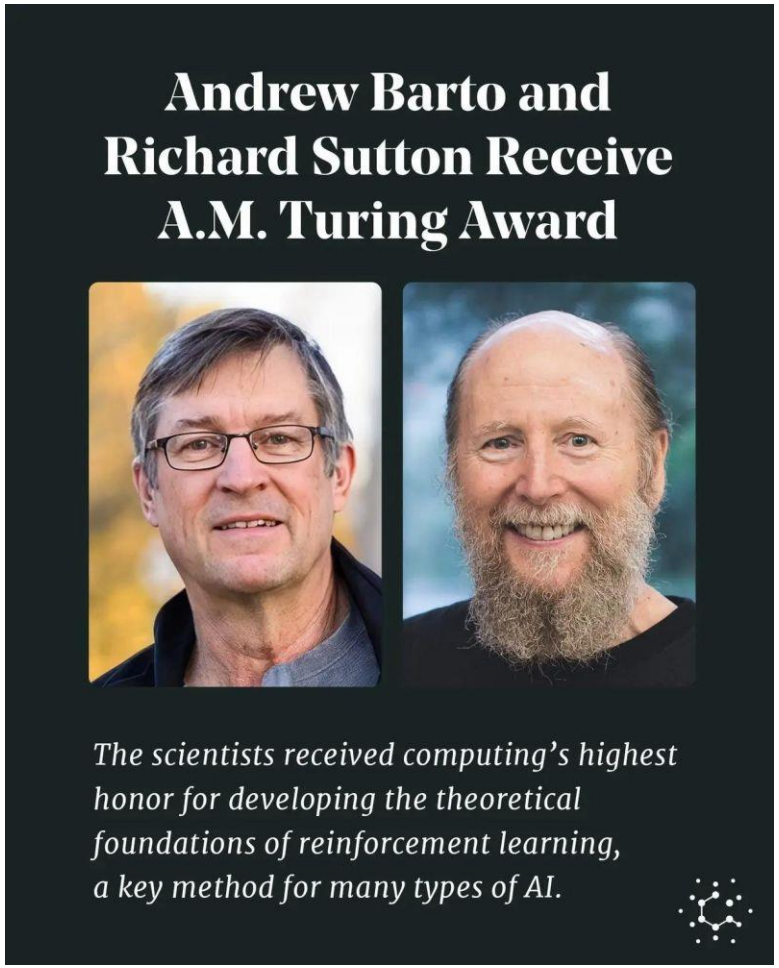
Reinforcement learning-based fine-tuning

InstructGPT – Reinforcement Learning from Human Feedback (**RLHF**)

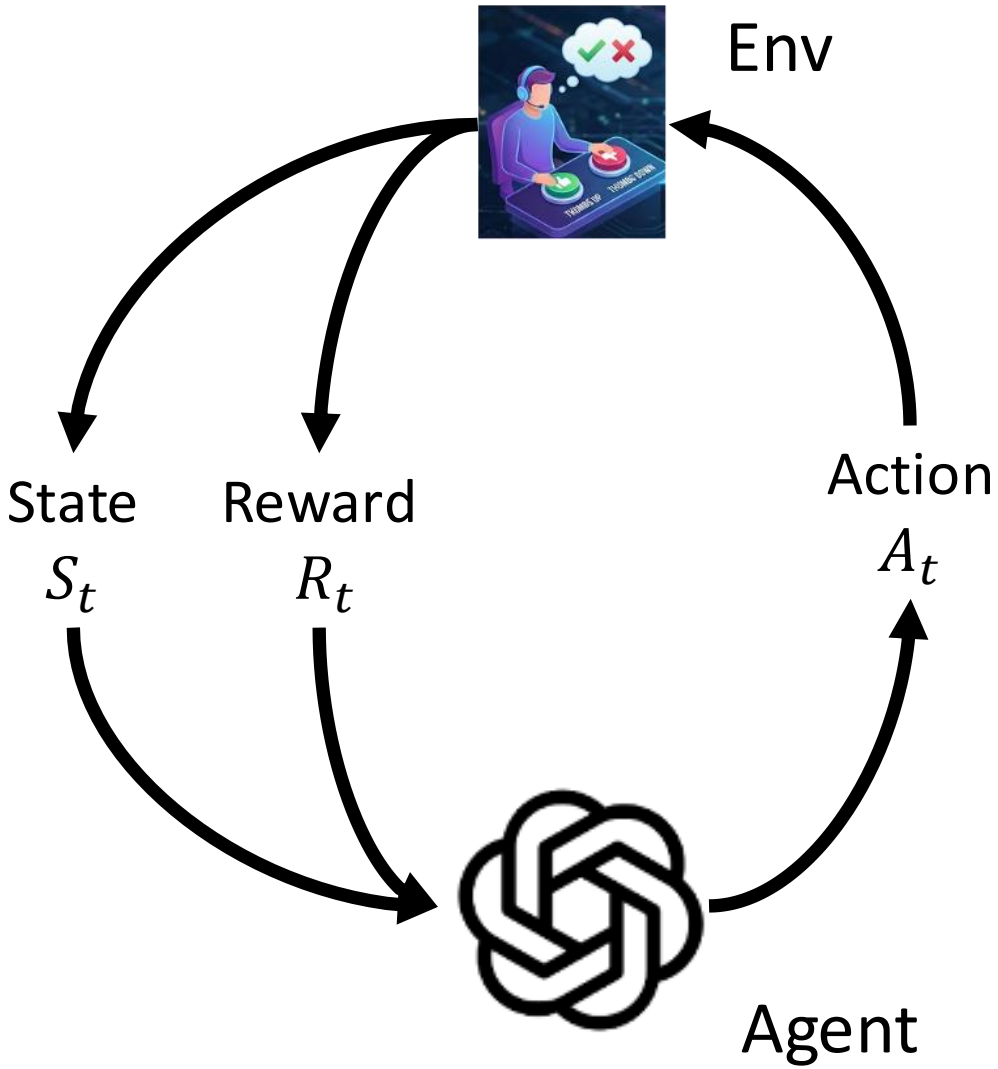
DeepSeek R1 – Reinforcement Learning from Verifiable Reward (**RLVR**)

Reinforcement Learning





Reinforcement Learning from Human Feedback

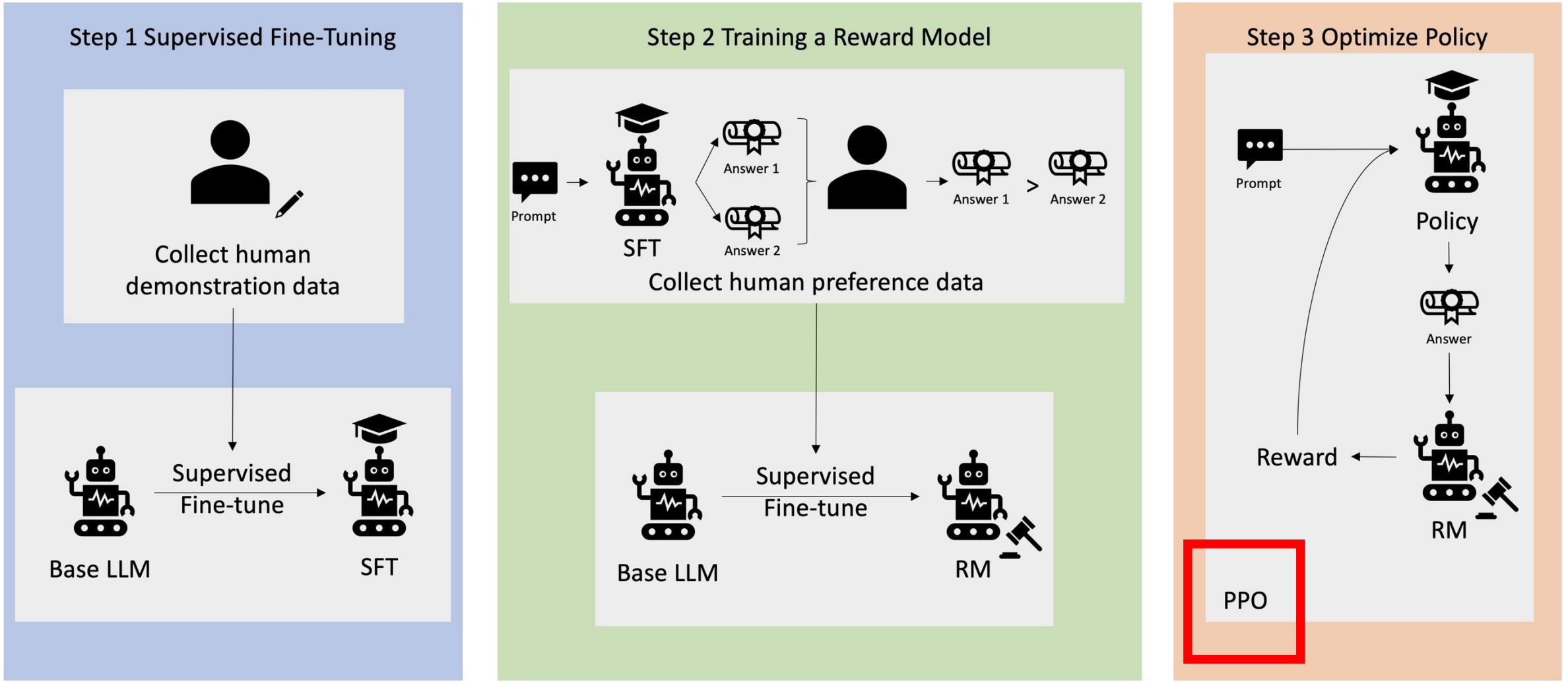


- **Env**: humans
- **State:** the user prompt + the tokens generated so far
- **Action:** the generation of a token
- **Reward:** human feedback

- **Agent**: LLM
- **Policy** π is next token distribution

Ouyang et al. "Training language models to follow instructions with human feedback." *NeurIPS, 2022*.

RLHF in OpenAI's InstructGPT



Source

Proximal policy optimization (PPO)

Algorithm 1 PPO, Actor-Critic Style

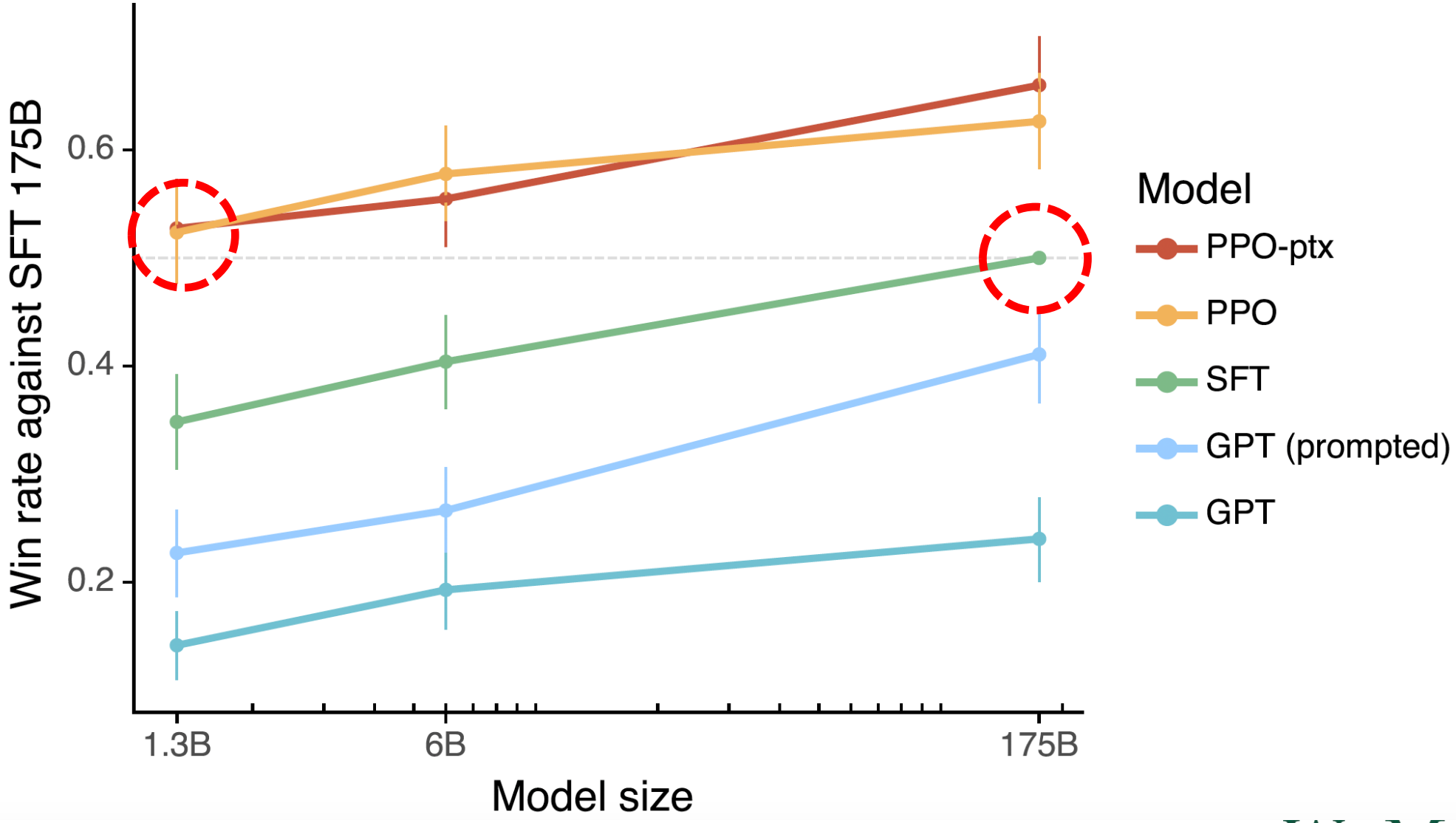
for iteration=$1, 2, \ldots$ **do**
  for actor=$1, 2, \ldots, N$ **do**
    Run policy $\pi_{\theta_{\text{old}}}$ in environment for T timesteps
    Compute advantage estimates $\hat{A}_1, \ldots, \hat{A}_T$
  end for
  Optimize surrogate L wrt θ, with K epochs and minibatch size $M \leq NT$
  $\theta_{\text{old}} \leftarrow \theta$
end for

$$L^{\text{PPO}}(\theta) = \mathbb{E}_{(s_t, a_t) \sim \pi_{\theta_{\text{old}}}} \left[\min \left(\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)} \hat{A}_t,\ \text{clip}\left(\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}, 1-\epsilon, 1+\epsilon \right) \hat{A}_t \right) \right]$$

$$\hat{A}_t = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l \left(r_{t+l} + \gamma V(s_{t+l+1}) - V(s_{t+l}) \right)$$

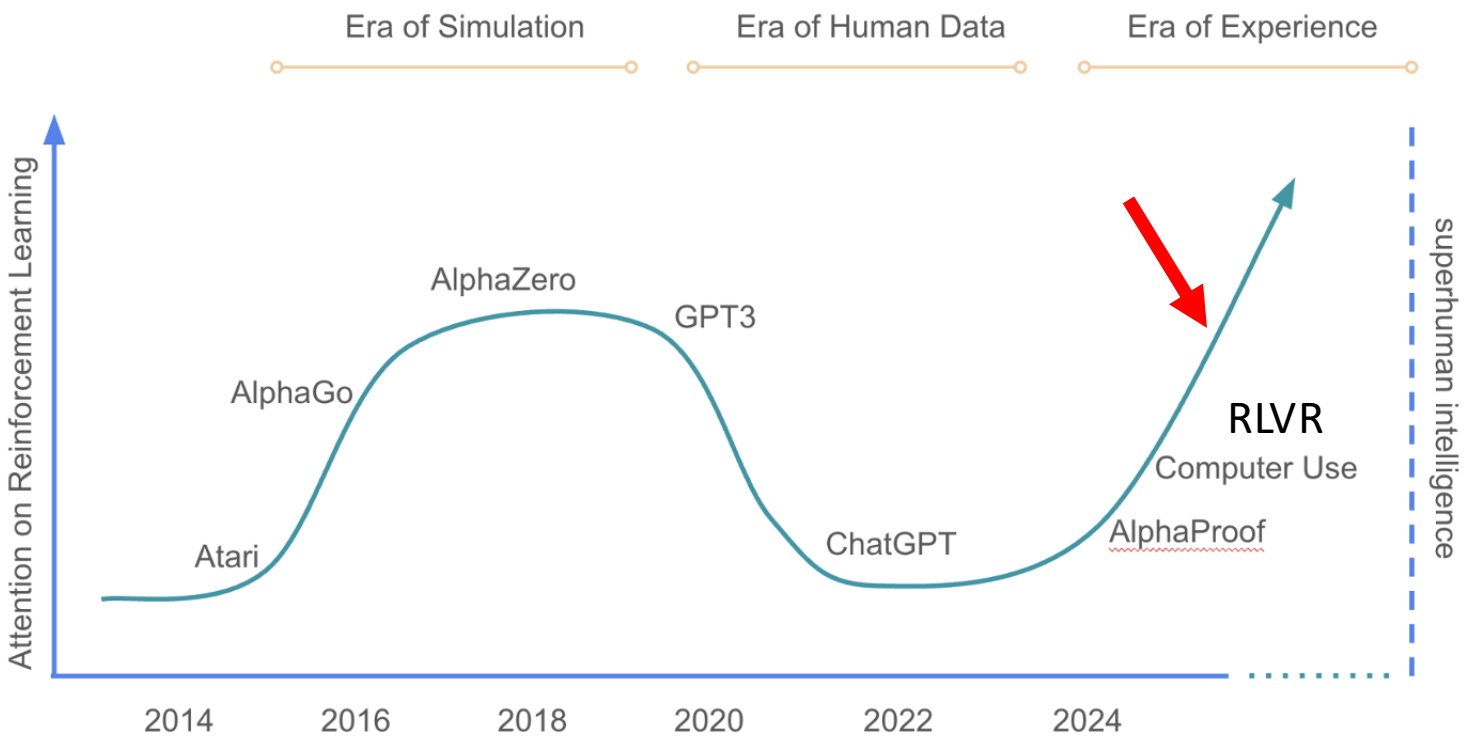
Schulman et al. “Proximal Policy Optimization Algorithms”. Arxiv, 2017

OpenAI InstructGPT (A sibling of ChatGPT)



Ouyang et al. "Training language models to follow instructions with human feedback." *NeurIPS, 2022*.

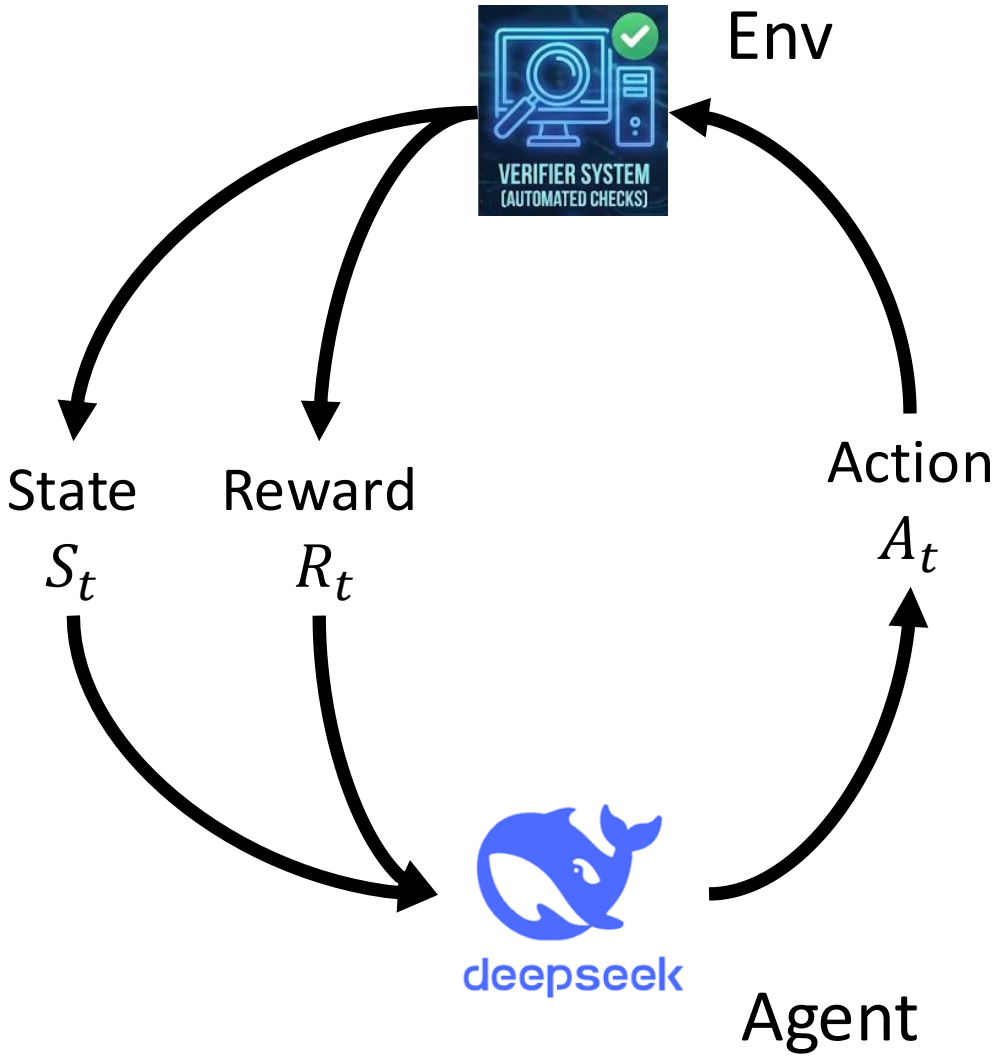
The RL roller coaster…



- RLHF still relies on tremendous labels

http://incompleteideas.net/papers/TheEraOfExperience.pdf

Reinforcement Learning from ~~Human Feedback~~ Verifiable Reward

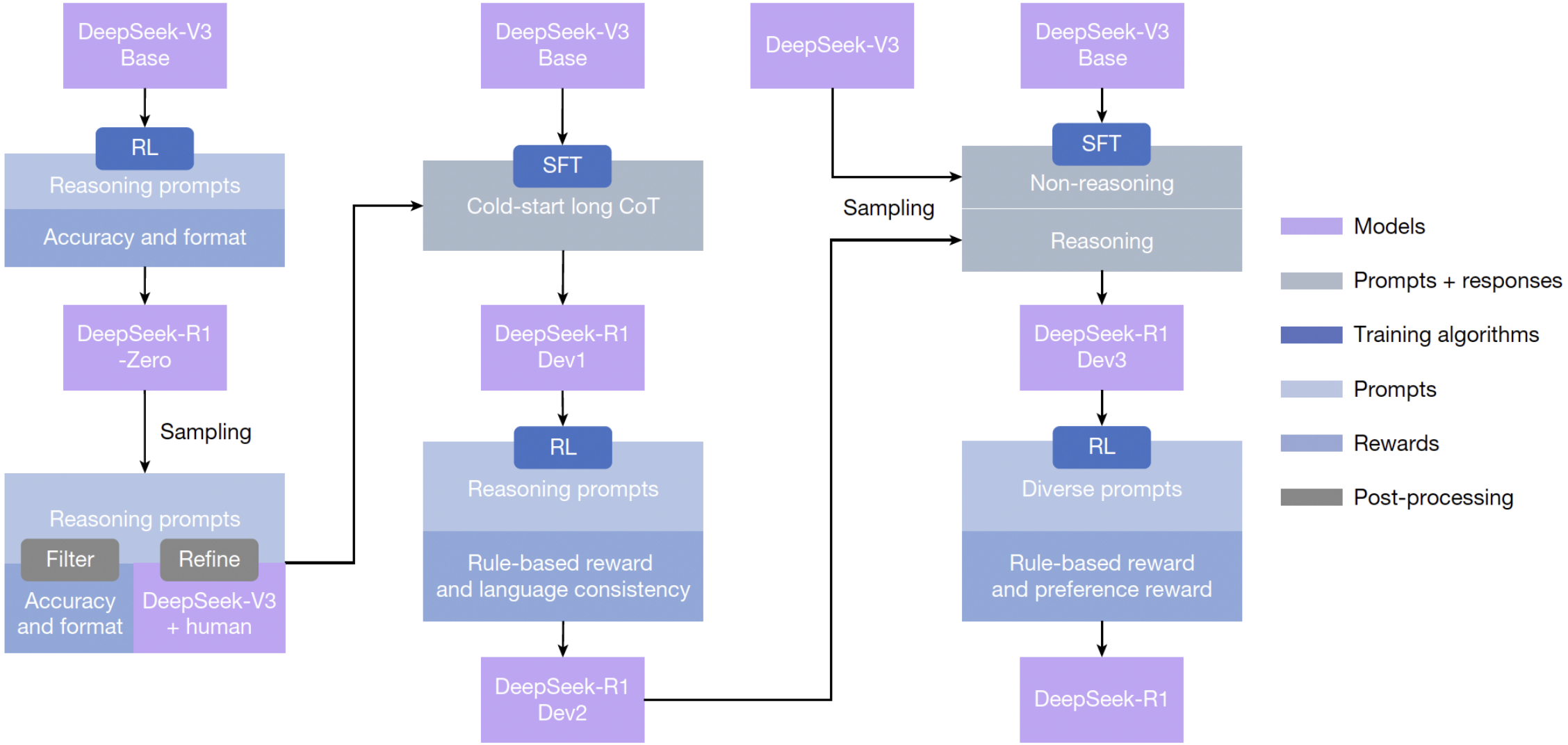


- **Env**: ~~humans~~ verifier system
- **State:** user prompt + tokens generated so far
- **Action:** the generation of a token in a *reasoning chain*
- **Reward:** ~~human~~ verifier feedback

- **Agent**: LLM
- **Policy** π is next token distribution

Guo et al. "DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning." *Nature,* 2025.

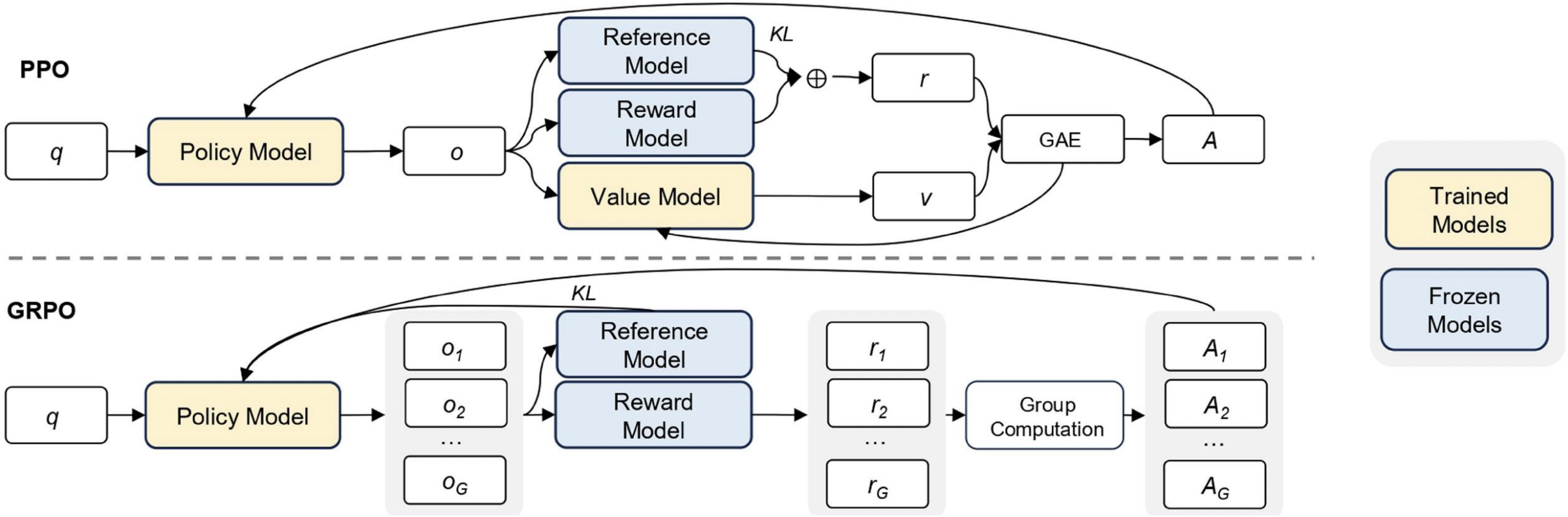
DeepSeek-R1 training pipeline



Guo et al. "DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning." *Nature,* 2025.

W&M DATA SCIENCE

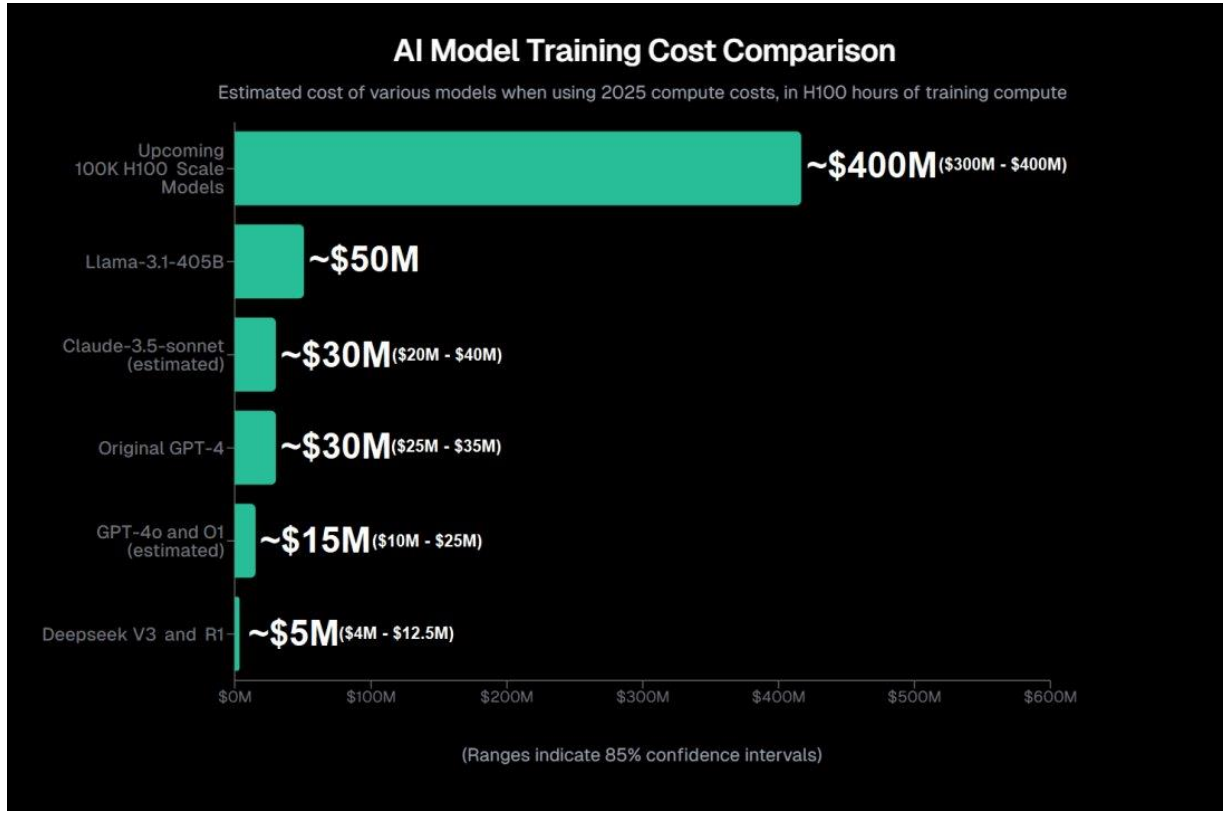
Group Relative Policy Optimization (GRPO)

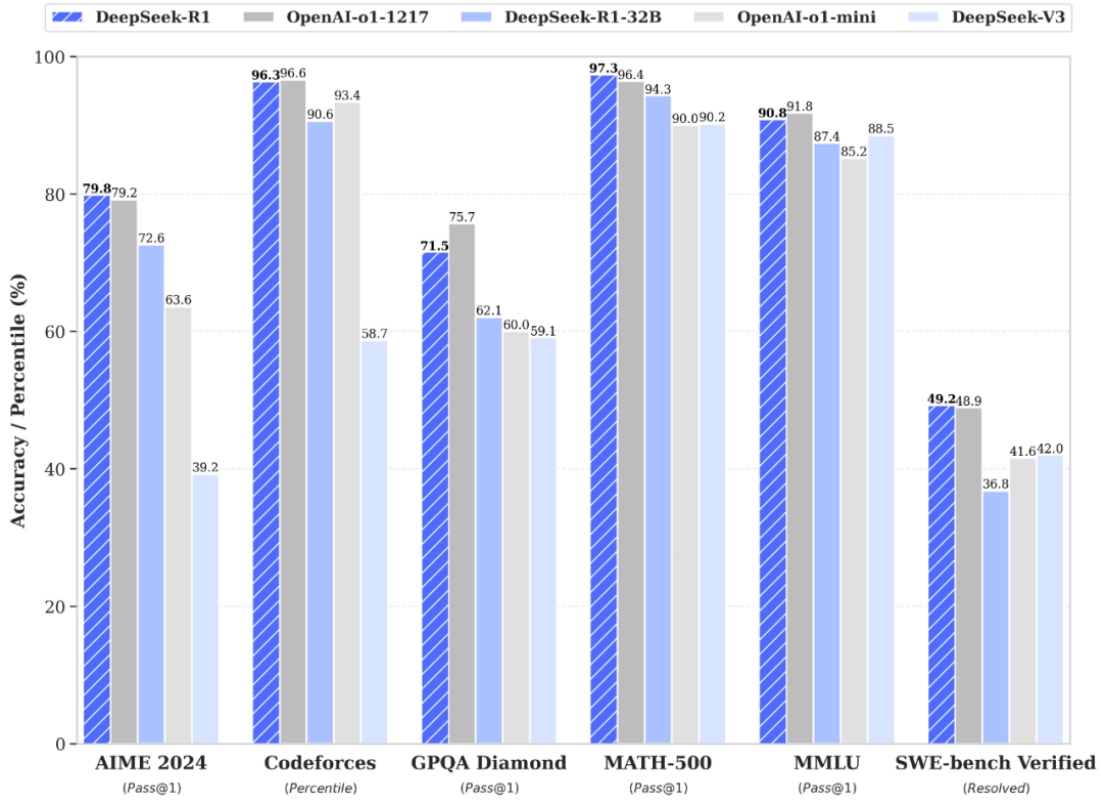


Guo et al. "DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning." *Nature,* 2025.

W&M DATA SCIENCE

DeepSeek R1





@arankomatsuzaki

https://api-docs.deepseek.com/news/news250120

Andrej Karpathy
@karpathy

2025 LLM Year in Review

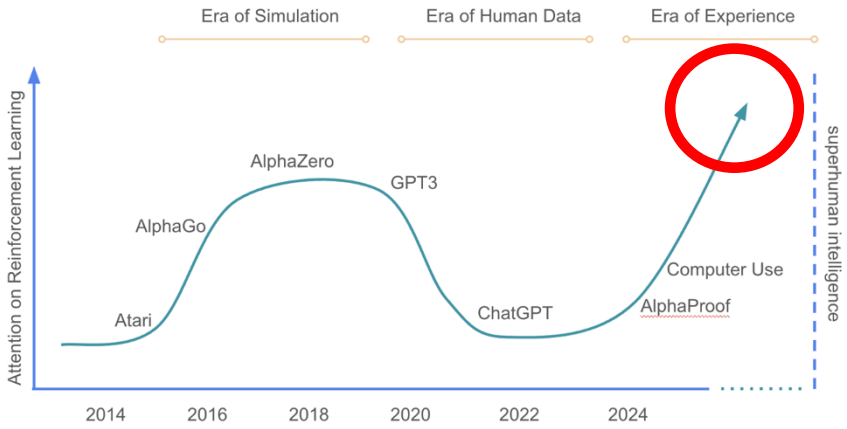
359 3.4K 15K 2.8M

2025 has been a strong and eventful year of progress in LLMs. The following is a list of personally notable and mildly surprising "paradigm changes" - things that altered the landscape and stood out to me conceptually.

1. Reinforcement Learning from Verifiable Rewards (RLVR)

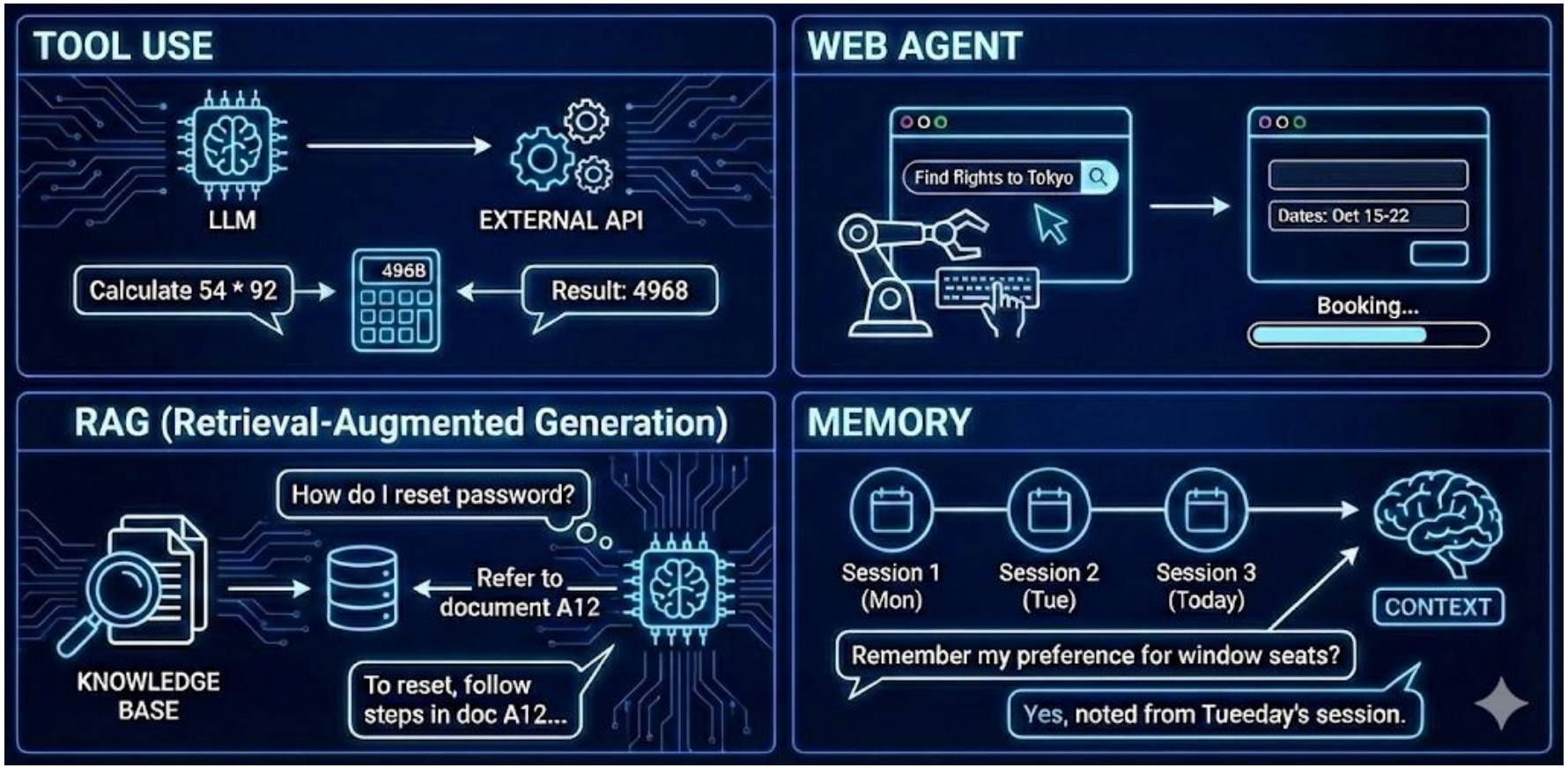
3:45 PM · Dec 19, 2025 · **2.8M** Views

(What I think) RL x LLMs research as of today?



- **RLHF is mature & standard recipe**
  Used mainly for *alignment and helpfulness*, not for discovering new core capabilities
- **RLVR works but is limited**
  Uses *verifiable / automatic rewards* (math, code, logic) to train reasoning without humans
- **Agents & "finer-tuning"**
  RL increasingly used for *tool use, planning, web-agents, …*

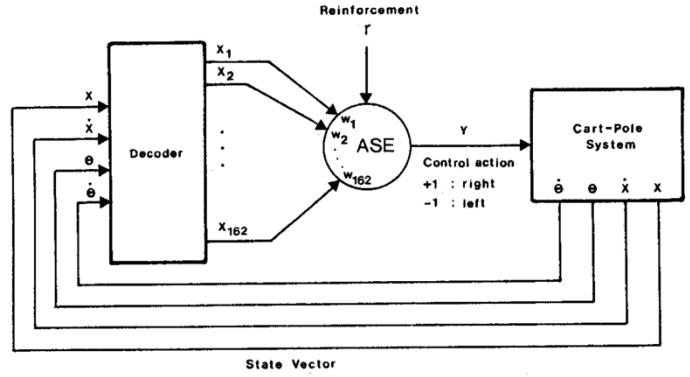
Taks-specific Steering in LLM via *finer-tuning*

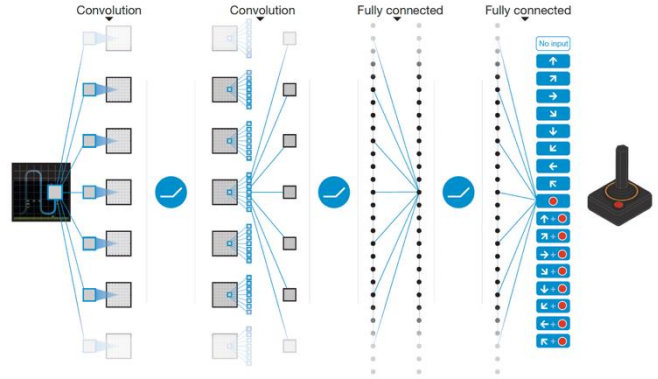


[1] Qian et al. ToolRL: Reward is all tool learning needs." NeurIPS, 2025.
[2] Qi et al. WebRL: Training LLM Web Agents via Self-Evolving Online Curriculum Reinforcement Learning. ICLR, 2025.
[3] Ma et al. Query rewriting in retrieval-augmented large language models. EMNLP, 2023.
[4] Yan et al. Memory-R1: Enhancing Large Language Model Agents to Manage and Utilize Memories via Reinforcement Learning. Arxiv, 2026.

Finer-tuning **is great, but is it all we need?**

- Finer-tuning is awesome
- large decision-transformers are great
- but wait… policy models used to be small/tiny (and they worked!)
- are small models no longer useful?





Ilya Sutskever – We're moving from the age of scaling to the age of research

Dwarkesh Patel
1.16M subscribers

28K | Share | Ask | Save

1,114,742 views Nov 25, 2025 Dwarkesh Podcast

Finer-tuning **LLMs may fail**

- Task-specific finer-tuning of LLMs usually leads to **model bias**
- Still needs substantial **compute**
- Not feasible when deploying in local, lesser devices (e.g., **edge device**)
- Needs **white-box** access to the model

Auxiliary Policy Models (APMs)

- Key idea: use RL to train **separate, light-weight APMs** to steer LLMs in down-stream tasks
- Still follows the **RLVR** framework
- Down stream tasks are formalized as Markov Decision Processes (MDPs)
- A **frozen LLM** is treated as part of the environment

Why APMs?

- More **flexible** solution
  - No need to touch the LLMs, model bias is injected to the APMs

- **Low training cost**
  - Ideal for low-resource computing, e.g., academia 🙂

- **Low inference cost**
  - Great for time-sensitive tasks and low-end device like edge computing

- (Sometimes) Works with a **black-box** model

Our exemplary studies

- APMs for Speculative Sampling
- APMs for In-Context Knowledge Editing

Our exemplary studies

- APMs for Speculative Sampling
- APMs for In-Context Knowledge Editing

Speculative Sampling (SpS)

- A technique used to accelerate LLM inference by parallelizing token generation through a “draft-then-verify” cycle

Leviathan et al. Fast inference from transformers via speculative decoding. *ICML*, 2023.
Chen et al. "Accelerating large language model decoding with speculative sampling." *ArXiv,* 2023.

Background: Tree-based drafting

- **Tree-based SpS:** Uses tree-based drafting to verify multiple token paths in parallel
- **Current SOTA (EAGLE 2 & 3**, Li et al., 2024-2025**)**

- **Gap**: Relies on **static hyperparameters** for the tree structure, failing to adapt to contexts of varying complexity. E.g, total tokens, tree depth, expansion factor

Miao et al. "Specinfer: Accelerating large language model serving with tree-based speculative inference and verification." ASPLOS 2024.
Li et al. "EAGLE: Speculative Sampling Requires Rethinking Feature Uncertainty." *ICML*, 2024.
Li et al. "Eagle-3: Scaling up inference acceleration of large language models via training-time test." *ArXiv,* 2025.

ReSpS: Speculative Sampling with Reinforcement Learning

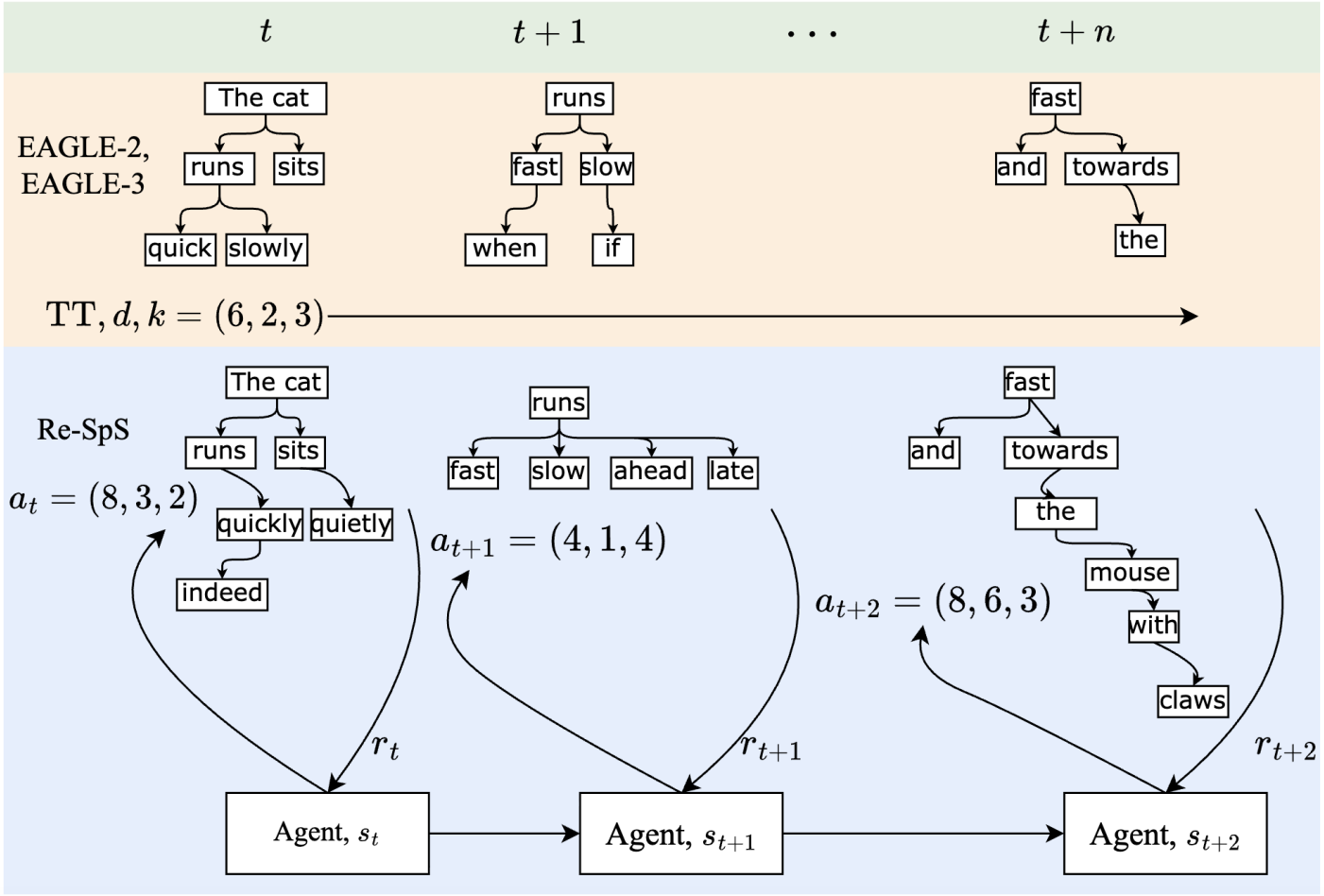

Chenan Wang, Daniel Shi, Haipeng Chen.

Speculative Sampling with Reinforcement Learning. AAAI, 2026.

Code: https://github.com/wmd3i/ReSpS

Re-SpS: Speculative Sampling with Reinforcement Learning

- **EAGLE-2 & 3**: Relies on static hyperparameters for the tree structure
- **Re-SpS**: Dynamically optimizes draft tree hyperparameters based on the context



Re-SpS vs. EAGLE-2 & 3

Wang, Shi, Chen. *Speculative Sampling with Reinforcement Learning*. AAAI, 2026.

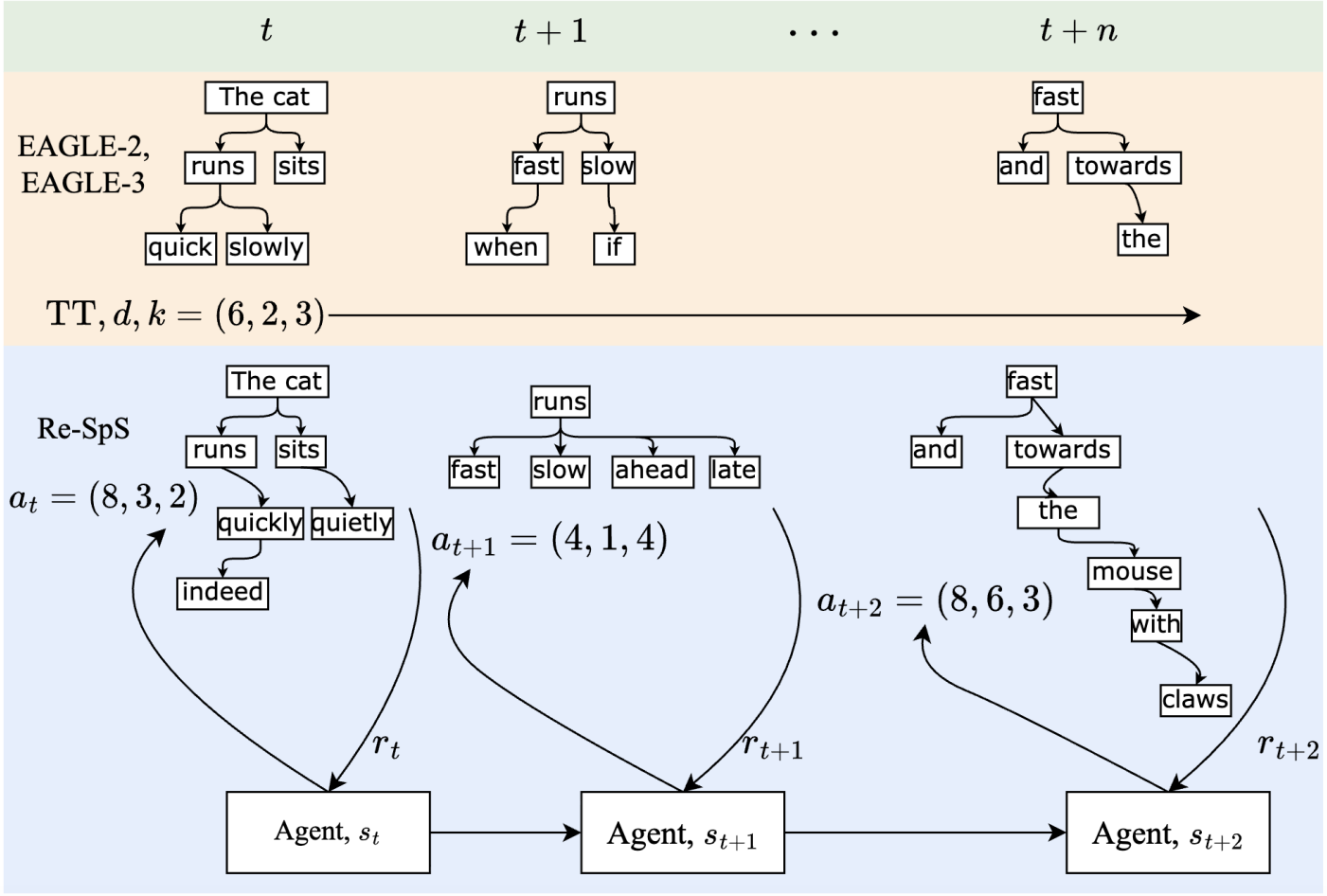
W&M DATA SCIENCE

The RL formulation of SpS

- **Env:** frozen LLM + speculative decoding framework (EAGLE-3)
- **State:** prompt + all previous tokens
- **Action:** tree structure hyperparameters (TT=total tokens, d=depth, k=expansion factor)
- **Reward:** token/second

$$r_t = \frac{\text{accepted tokens}}{\text{elapsed time (seconds)}}$$

- **Agent**: **APMs (2-layer MLP)**



Re-SpS vs. EAGLE-2 & 3

Wang, Shi, Chen. *Speculative Sampling with Reinforcement Learning*. AAAI, 2026.

Computation overhead

- Optimizing SpS with RL introduces computation/memory overhead:
  - **State representation overhead:** Generating rich semantic embeddings (e.g., SentenceBERT) at every generation step incurs non-negligible latency and memory overhead.
  - **Frequent policy calls**: Performing frequent policy calls (e.g., per drafting step) further aggravated the above overhead.

Overcoming RL overhead

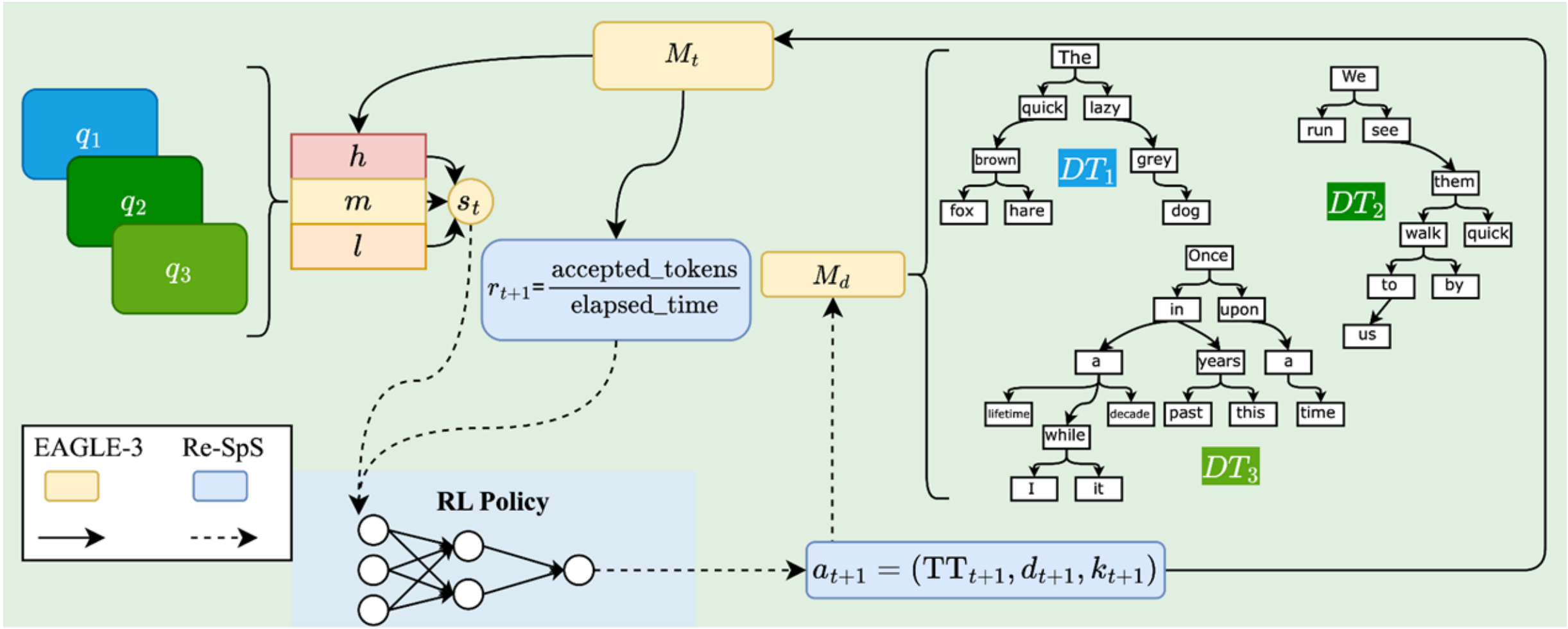
- **Innovation 1: Efficient feature reuse**
  - Instead of costly external embeddings (e.g., SentenceBERT), we reuse the **target model's internal hidden states**:

$$s_t = [h_{\text{LM}}^{(h,m,l)}]$$

  🙂 *Free lunch - Rich context representation with zero extra inference cost*

- **Innovation 2: Multi-step action persistence**
  - Amortize policy inference cost by caching the selected action for every N decoding steps (e.g., N = 10 or 30)

Overview of Re-SpS



Wang, Shi, Chen. *Speculative Sampling with Reinforcement Learning*. AAAI, 2026.

W&M DATA SCIENCE

Experimental setup

- Target models: LLaMA 3.1-8B, Vicuna-13B, LLaMA 3.3-70B
- Across 5 benchmarks (MT-Bench, HumanEval, GSM8K, Alpaca, CNN/DM)

- LLaMA 3.1-8B and Vicuna-13B runs on a **single NVIDIA A40 GPU (40GB)**
- LLaMA 3.3-70B runs on **4 NVIDIA H100 GPUs (80GB each)**

- All experiments use PPO as the RL algorithm
- **APM represented as a 128x128 MLP**
- **Training time**: LLaMA3.3-8B – **8.65 hrs**; Vicuna-13B – **10.13 hrs**; LLaMA3.1-70B – **11.05 hrs**

Re-SpS empirical results

Backbone	Method	MT-Bench	HumanEval	GSM8K	Alpaca	CNN/DM	Mean	p-value
LLaMA 3.1-8B	Medusa	2.07×	2.50×	2.23×	2.08×	1.71×	2.12×	–
	Hydra	2.88×	3.28×	2.93×	2.86×	2.05×	2.80×	–
	EAGLE-3	3.39×	3.65×	3.52×	3.67×	**2.96**×	3.44×	–
	Re-SpS	**3.43**×	**3.89**×	**3.62**×	**3.90**×	2.87×	**3.54**×	$< 10^{-4}$
Vicuna-13B	Medusa	2.07×	2.50×	2.23×	2.08×	1.71×	2.12×	–
	Hydra	2.88×	3.28×	2.93×	2.86×	2.05×	2.80×	–
	EAGLE-3	3.75×	4.28×	3.85×	3.76×	**3.35**×	3.80×	–
	Re-SpS	**3.76**×	**4.64**×	**3.99**×	**3.99**×	3.24×	**3.92**×	$< 10^{-9}$
LLaMA 3.3-70B	EAGLE-3	4.35×	4.87×	4.74×	4.77×	**4.09**×	4.46×	–
	Re-SpS	**4.47**×	**5.45**×	**5.13**×	**5.34**×	4.03×	**4.88**×	$< 10^{-29}$

- **Average speedup vs EAGLE-3:**
  1) **LLaMA 3.1-8B:** 1.03× 2) **Vicuna-13B:** 1.04× 3) **LLaMA 3.3-70B:** 1.06×
- **Fidelity**: Lossless (exact match to greedy decoding)

Wang, Shi, Chen. *Speculative Sampling with Reinforcement Learning*. AAAI, 2026.

Our exemplary studies

- APMs for Speculative Sampling

- APMs for In-Context Knowledge Editing

In-context knowledge editing

- **In-Context Knowledge Editing**: Edit model knowledge via prompted demonstrations

🕐 Before Editing

👤 **Question:** *Who is the current Prime Minister of the United Kingdom?*

📄 **Model's Stored Knowledge (Pre-Update):** "Rishi Sunak has been serving as Prime Minister from 2022"

🤖 **LLM:** *Rishi Sunak.* ❌

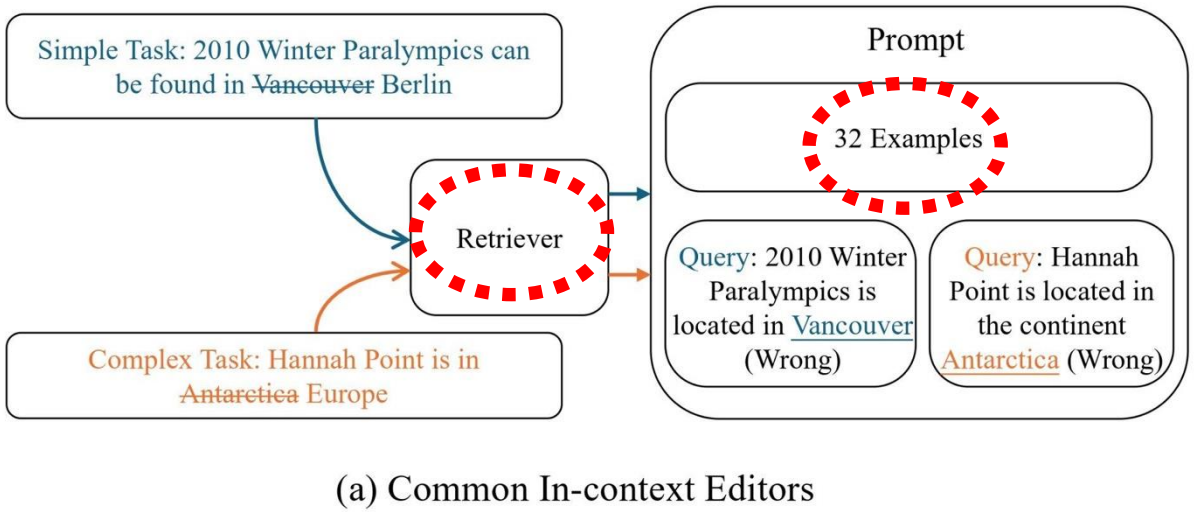
🧩 After Editing

👤 **Question :** *Who is the current Prime Minister of the United Kingdom?*

📄 **Model's Injected Knowledge (Post-Update):** "Keir Starmer is the UK Prime Minister now"

🤖 **LLM (with in-context prompt or modified parameters):** *Keir Starmer.* ✅

In-context knowledge editing



(a) Common In-context Editors

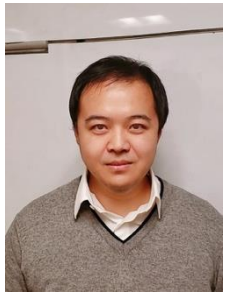
Current in-context editors (e.g., IKE [Zheng et al. 2023])

- Use **static retrieval** and **fixed number of prompt templates**
- **Ignore task difficulty** — simple and complex edits get equal-length prompts
- Become unstable and less generalizable when examples are **redundant or poorly aligned**

Zheng et al. "Can We Edit Factual Knowledge by In-Context Learning?." *EMNLP, 2023*.

DR-IKE

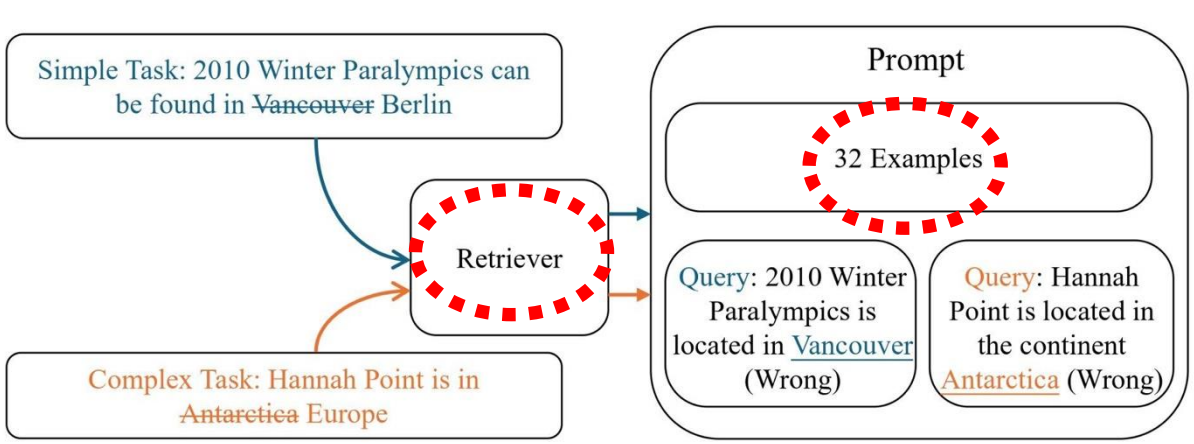




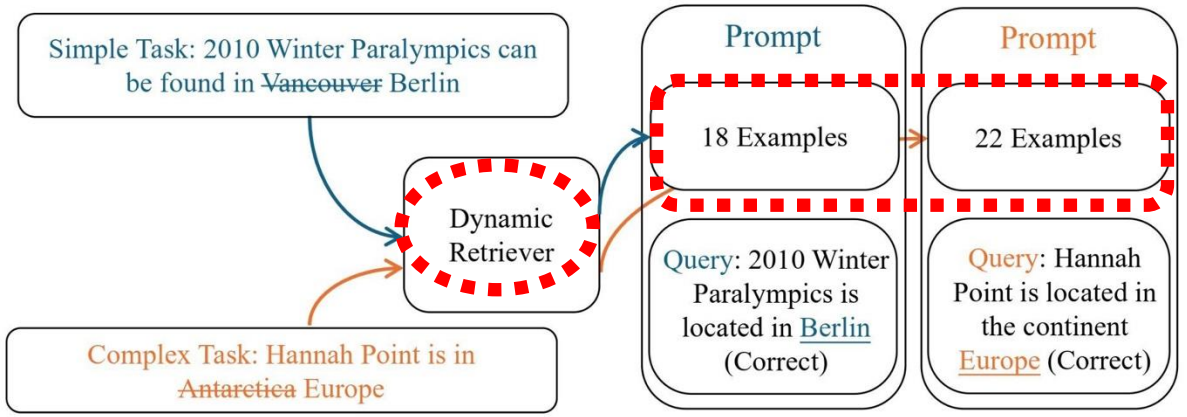
Mahmud Wasif Nafee*, Maiqi Jiang*, Haipeng Chen†, and Yanfu Zhang†.
DR-IKE: Dynamic Retriever for In-Context Knowledge Editing via Policy Optimization.
EMNLP, 2025.

Code: https://github.com/mwnafee/DR-IKE

Dynamic Retriever for In-Context Knowledge Editing via Policy Optimization



(a) Common In-context Editors

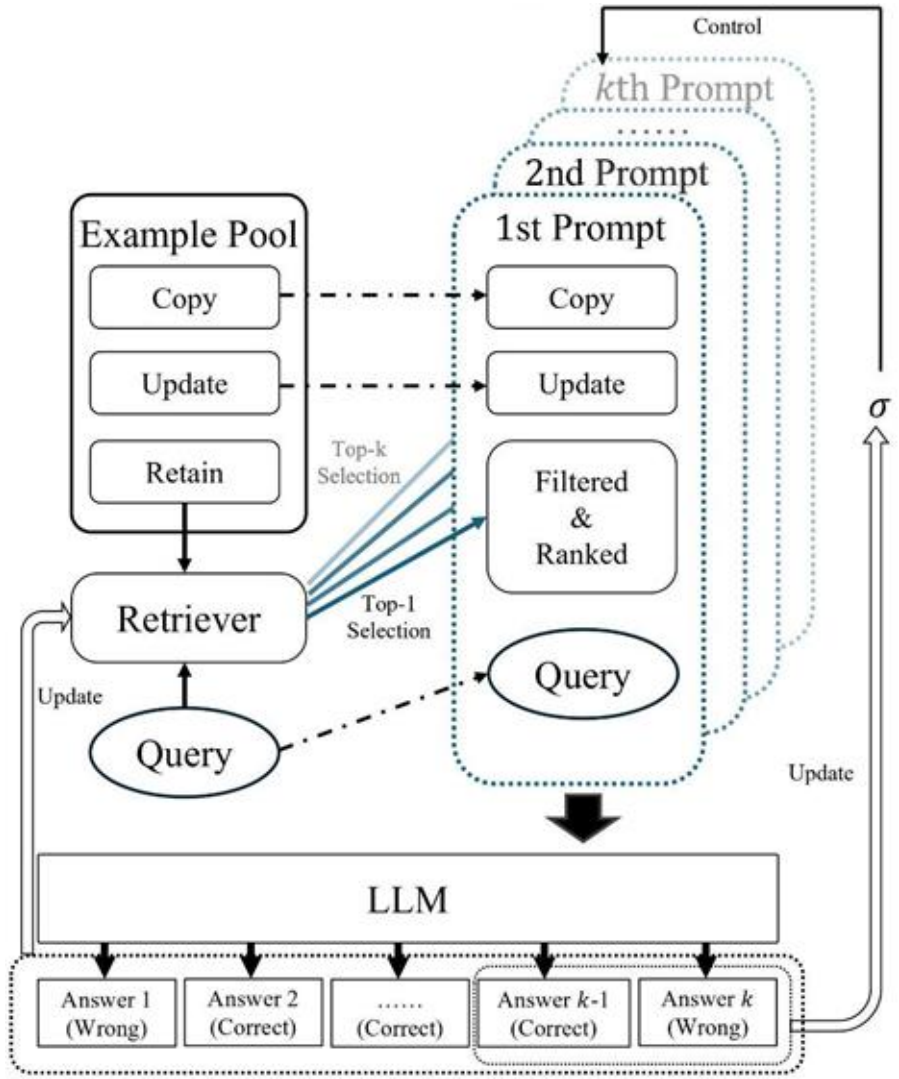


(b) Our Dynamic Retriever In-context Editor

DR-IKE (Dynamic Retriever for In-Context Knowledge Editing)

- **Adaptive context construction:** Adjusts the **ranking and number** of examples per query
- **Auxiliary retrieval policy model:** Incorporates LLM response correctness as reward signals to train a retrieval APM

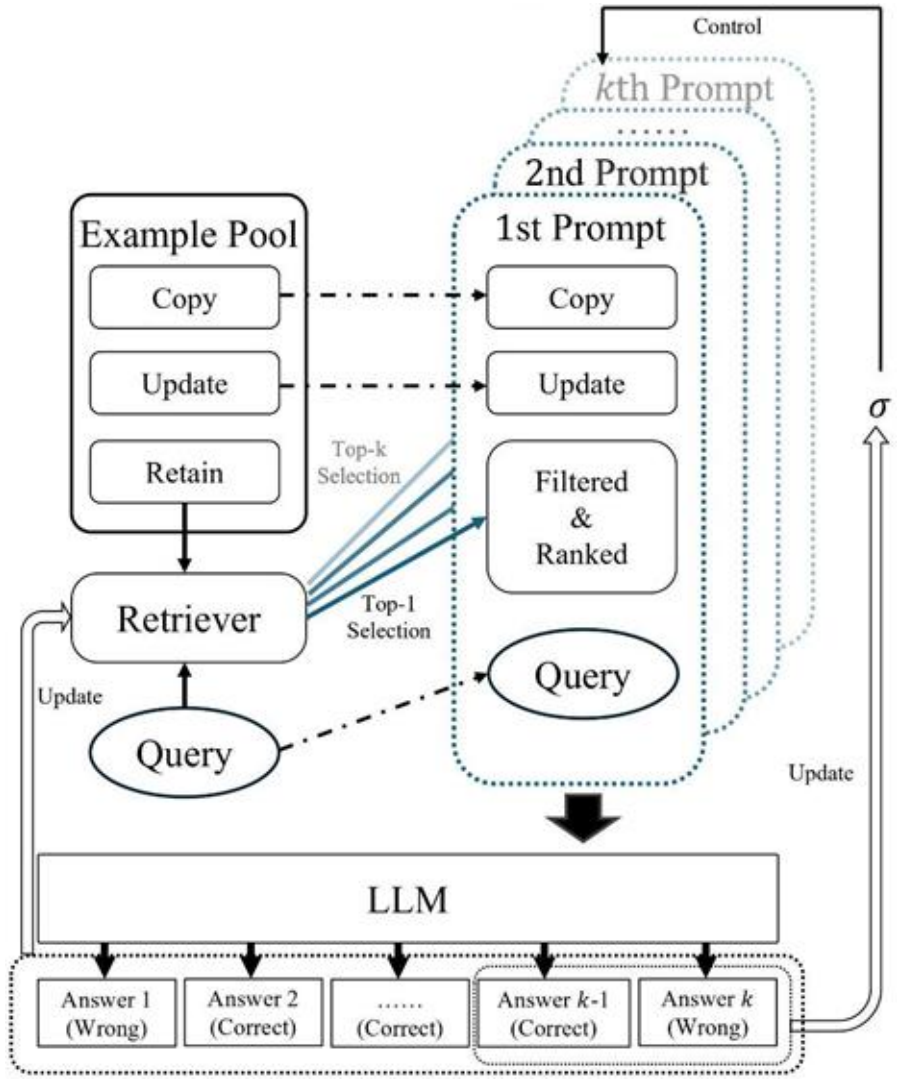
Overview of DR-IKE



- **Example pool:** pool of examples to retrieve
  - Three types: Copy, Update, Retain
- **Retriever:** scores examples by relevance
- **LLM editor:** Generates response & reward signal drives training.
- **Budget controller (σ):** Controls how many examples enter the prompt
  - Can be fixed or a learnable parameter

Nafee, Jiang, Chen, Zhang. DR-IKE: Dynamic Retriever for In-Context Knowledge Editing via Policy Optimization. EMNLP, 2025

Overview of DR-IKE

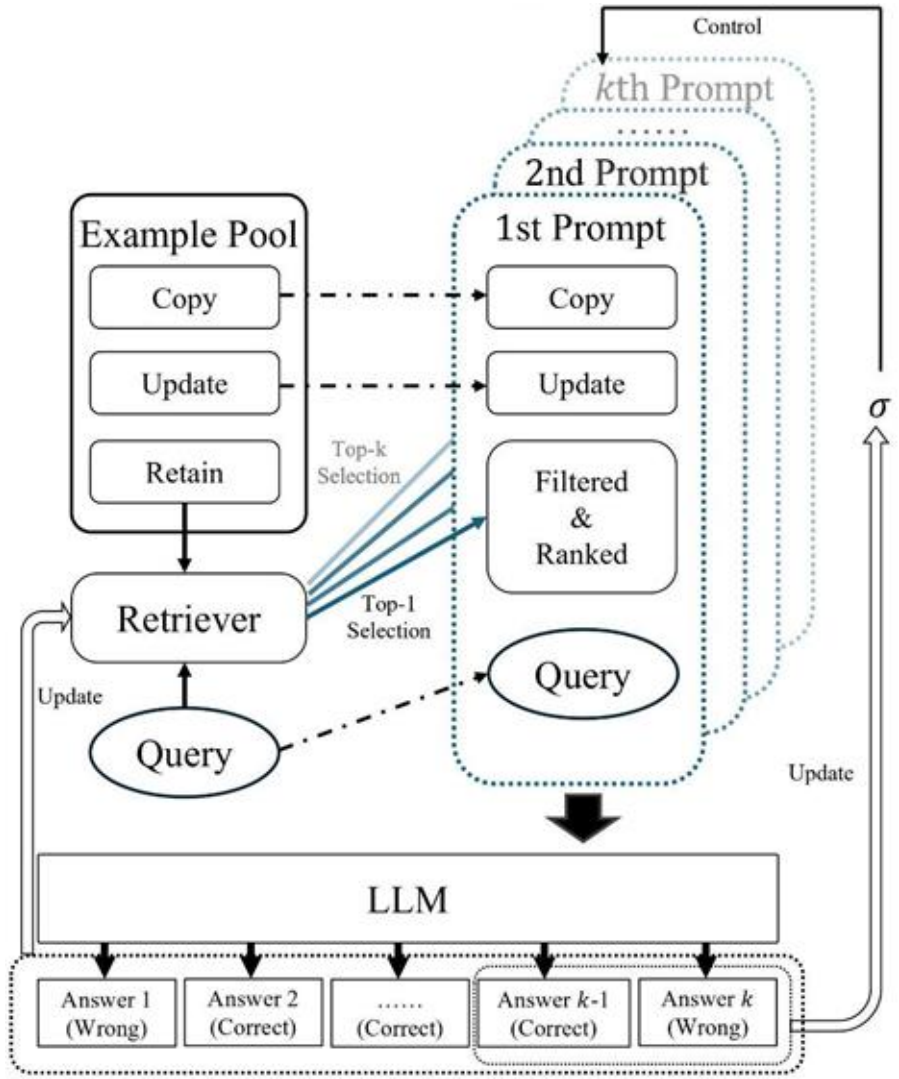


Dynamic Retriever as the APM

- **Frozen BERT encoder** for contextual features
- **Trainable linear head** producing candidate scores
- Softmax over scores gives selection probabilities
- High-probability retains → presented first

Nafee, Jiang, Chen, Zhang. DR-IKE: Dynamic Retriever for In-Context Knowledge Editing via Policy Optimization. EMNLP, 2025

Overview of DR-IKE



The RL formulation

- **Env**: Frozen target LLM + in-context knowledge editing task and feedback
- **State**: Query + candidate retain examples
- **Action**: Ranking of examples for the prompt
- **Reward**: +1 → if LLM's output matches the correct answer, –1 → otherwise
- **Agent: APM (linear head** after frozen Sentence-BERT**)**

Nafee, Jiang, Chen, Zhang. DR-IKE: Dynamic Retriever for In-Context Knowledge Editing via Policy Optimization. EMNLP, 2025

W&M DATA SCIENCE

Experimental setup

- **Three evaluation datasets** (CounterFact, zsRE, Wikidata-CF)
- Evaluation **Metrics**:
  - **ESR** — Edit Success Rate: Accuracy on edited facts
  - **PC** — Paraphrase Consistency: Consistency across reworded queries
  - **RR** — Retain Rate: Preservation of unrelated original facts

Experimental setup

- **APM training**:
  - Same frozen LLM backbone (Llama-3.1-8B-Instruct) is used across all datasets
  - BERT-small (29M) as the **frozen encoder**, a **trainable linear layer** as the APM
  - Only needs **300 examples** for the APM training
  - All training done with a **single NVIDIA L4 GPU (24G), training time < an hour**

Results

CounterFact

Method	ESR ↑	PC ↑	RR ↑
FactPrompt	0.61	0.34	0.43
EditCoT	0.70	0.33	0.40
IKE	0.76	0.67	**0.76**
DR-IKE	**0.89**	**0.81**	0.66

- DR-IKE outperforms all baselines in edit accuracy and consistency
- Offers a better trade-off between successful edits and knowledge retention

Nafee, Jiang, Chen, Zhang. DR-IKE: Dynamic Retriever for In-Context Knowledge Editing via Policy Optimization. EMNLP, 2025

Results

zsRE

Method	ESR ↑	PC ↑	RR ↑
IKE	0.30	0.22	0.49
DR-IKE	**0.33**	**0.26**	**0.51**

Wikidata-CF

Method	ESR ↑	PC ↑	RR ↑
IKE	0.39	0.40	0.63
DR-IKE	**0.42**	**0.43**	**0.64**

Nafee, Jiang, Chen, Zhang. DR-IKE: Dynamic Retriever for In-Context Knowledge Editing via Policy Optimization. EMNLP, 2025

Follow-up works…

- Tejal Nair, Mahmud Wasif Nafee, Maiqi JIANG, Ashley Gao, Haipeng Chen, Yanfu Zhang. ***Confidence-Aware Ranker Ensembles for Robust In-Context Knowledge Editing***. ACL Findings, 2026.

APM takeaways

- Lightweight APMs can be used to steer LLM downstream tasks such as speculative sampling and in-context knowledge editing (and many more
- **Low-cost, fast, flexible**, and (sometimes) works for **black-box/edge** settings
- Down stream tasks are formalized as MDPs (often the key)
    - A frozen LLM is part of the environment
- **Code repositories for both projects are released on GitHub**

Random thoughts…

- How small is small, when are models big/small enough?
- Tradeoff between quality, generalizability, efficiency

- Resembles Action models in VLAs and WAMs
- Resembles the “system 2” thinking

- Diffusion language models are a more flexible family of LMs
- But how to steer/control its generation?

The future of AGI lies not in a monolithic model, but a "community" of big and small, composable and modularized models

Thanks! Questions and thoughts?

Haipeng Chen

Data Driven Decision Intelligence (D3i) Lab

Assistant professor, William & Mary