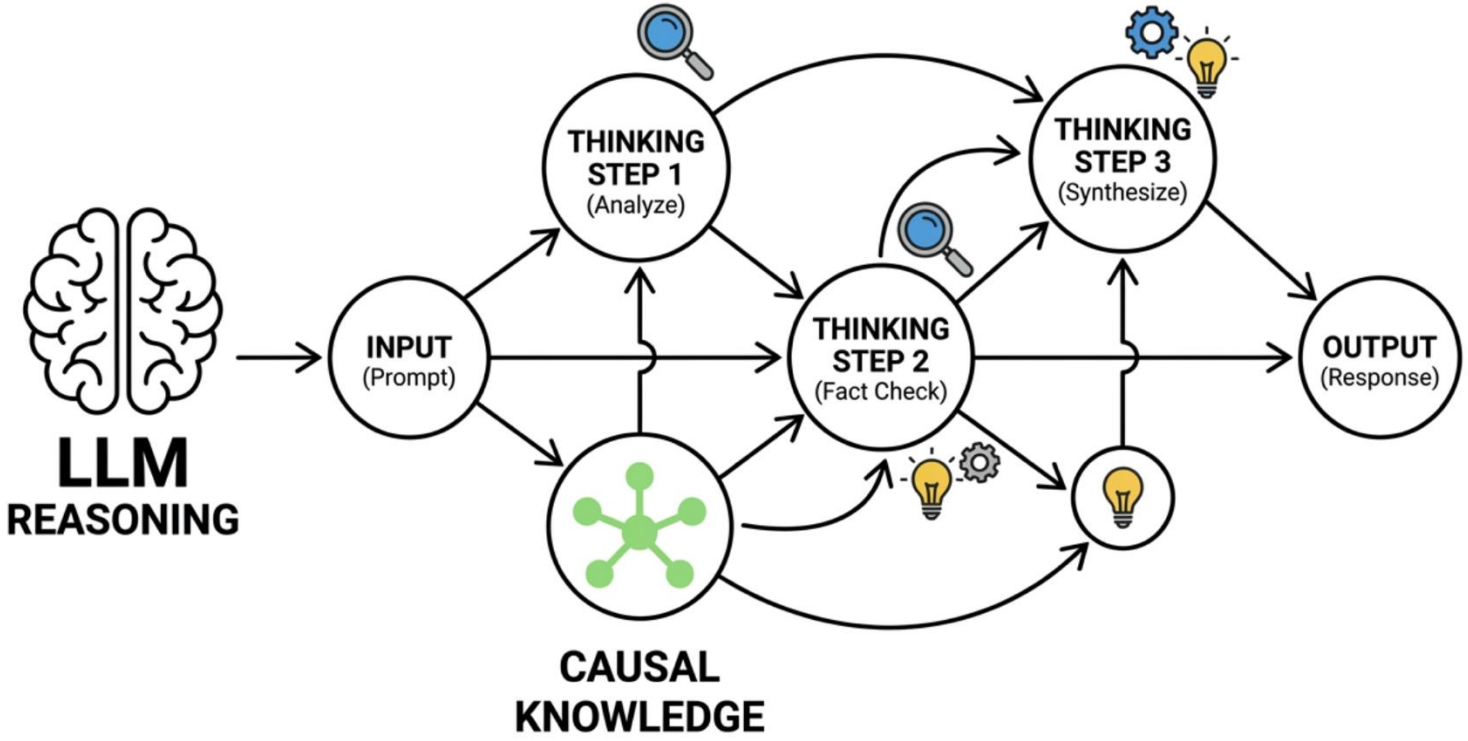
Grounded LLM Reasoning

Mengyue Yang

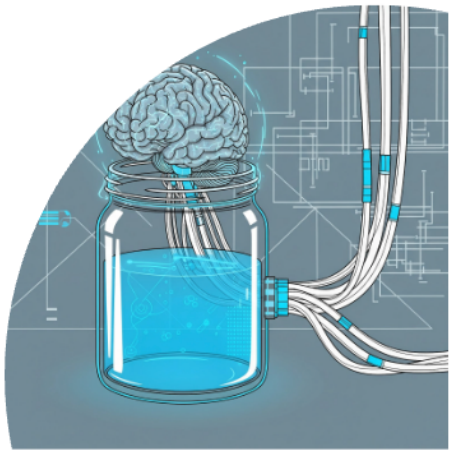
Lecturer in AI

University of Bristol



Grounded LLM Reasoning

Beyond Next-Token Prediction: Moving from Probabilistic Echoes to Reality-Anchored Intelligence



The "Brain in a Vat" Problem

Pure LLMs operate in a vacuum of tokens, predicting the next word without a physical anchor or true understanding of cause and effect.

01. Causal World Understanding

Models transition from statistical correlation to causal comprehension, understanding "why" things happen rather than just "what" word follows.

02. Evolution through First-Hand Experience

Intelligence emerges from active interaction with the environment, extracting knowledge from experience rather than relying solely on pre-trained text.

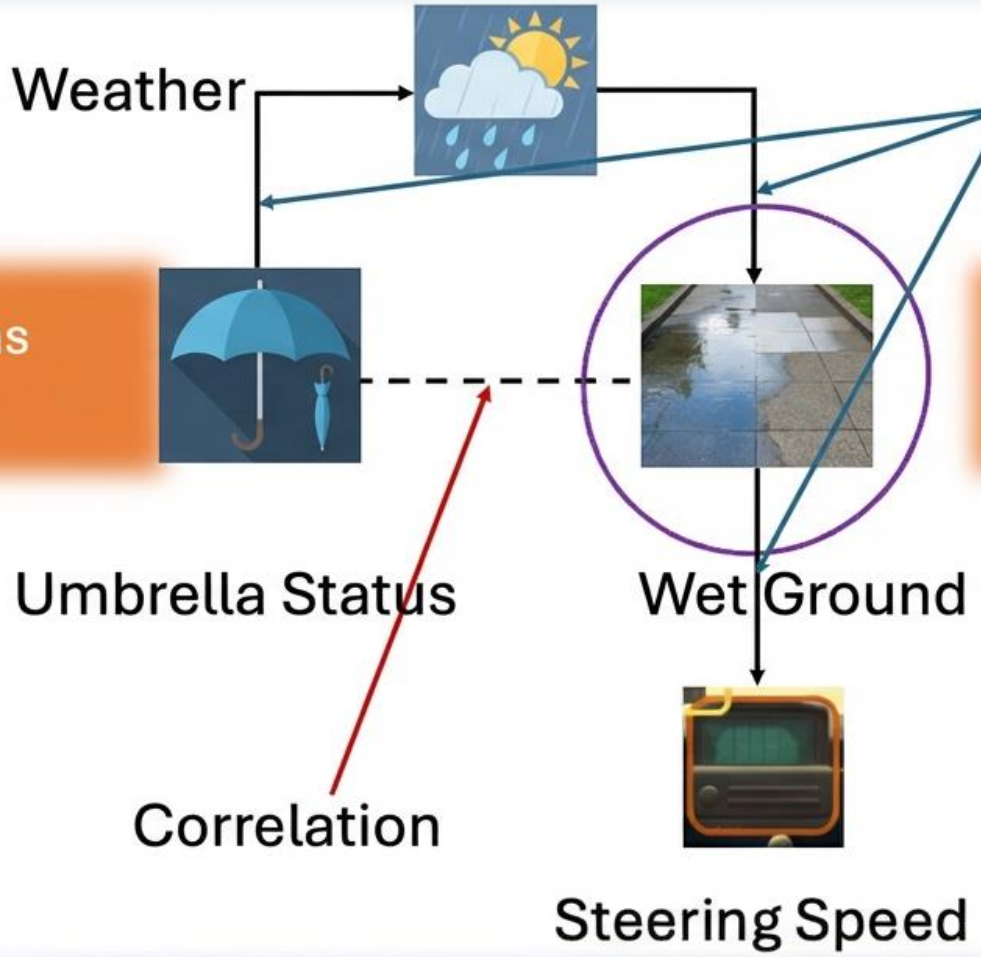
03. Explicit World Modeling

Constructing internal simulations of reality to ground reasoning in physical and logical constraints, moving away from "castles in the air."

Reasoning ≠ Understanding

Is current reasoning model perfect?

Mimic the world data doesn't means model understand the world



Rainy weather causes people to open umbrellas and also makes the ground wet.

People open umbrellas when it rains.

Rain directly causes the ground to be wet.

Adjusting Umbrella Status (opening/closing) is a wrong decision to change Wet Ground, as it's only a correlation.

Pearl's Causal Hierarchy

Predict something haven't happened

Imagine based on something already happened

Seeing
Association 

Question:
What is?

What does the smoking tell us about the lung cancer.

Doing
Intervention 

Question:
What if?

What will happen if someone keep smoking

Imagining
Counterfactual 

Question:
Was it?

Would the lung cancer got worse if someone smoking.

Predicting of future
Reflecting of past

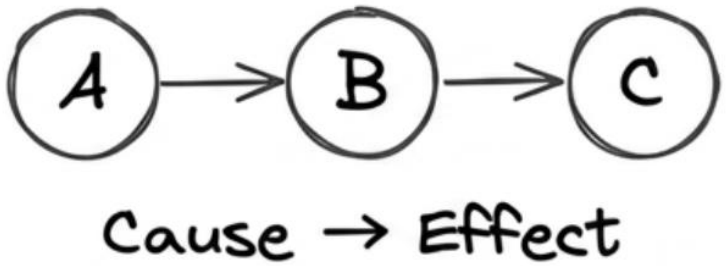
=

Intervention
Counterfactual

Reasoning steps as causal variables

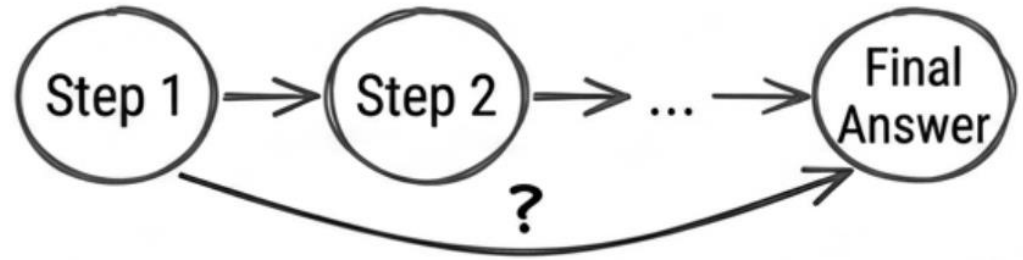
Can we identify which reasoning steps are truly causally responsible for the final answer?

General Causal Reasoning



Identify cause-effect in systems.

Chain-of-Thought as Causal Graph



Which steps truly cause the final answer?

Understand cause-effect.

Identify key reasoning steps.

Two core challenges in CoT

Sufficiency (Completeness)

- **Definition:** Ensure generated intermediate reasoning steps fully cover and validate the final conclusion.
- **Problem:** Incomplete steps or broken logic lead to incorrect answers, like missing key proof steps.

Ensure complete logic for valid conclusions.

Necessity (Redundancy)

- **Definition:** Identify truly indispensable steps for reaching the correct answer.
- **Problem:** "Overthinking" with redundant steps reduces efficiency and may introduce errors.

Remove redundant steps for efficiency and accuracy.

An intuitive reasoning patterns

Thinking Exercise: Calculate $99^2 + 2*99 + 1$ in your head

Pattern 1: Sufficient but Unnecessary

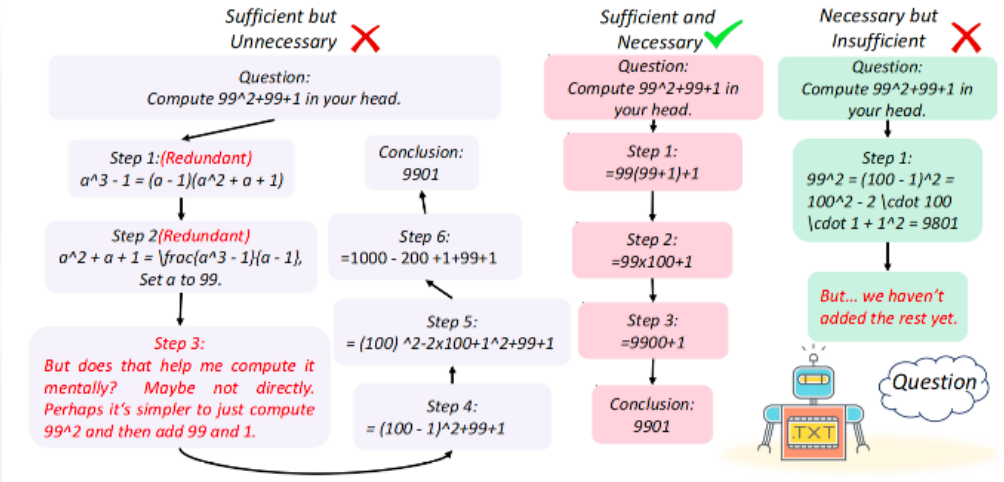
Expand and calculate 99^2 directly, then add 99 and 1. Lengthy steps with high error risk.

Pattern 2: Necessary but Insufficient

Attempts to find a shortcut but gets stuck midway, unable to derive the answer.

Pattern 3: Sufficient & Necessary (Optimal)

Identifies the formula: $99^2+2*99+1=(99+1)^2=10000$. Minimal steps, precise logic.



Core Insight

Optimizing CoT means guiding the model to generate "sufficient and necessary" optimal reasoning paths.

Probability of Sufficiency (PS)

Intuition

Measures if a given reasoning chain **S** is **enough** to produce the correct answer **y**.

"If the model had used this chain, would it have arrived at the correct answer, given that it originally didn't?"

Interpretation

The likelihood that inserting reasoning chain S would change an **incorrect answer** to a **correct one**.

It quantifies the reasoning chain's **power to fix mistakes**, regardless of the model's original reasoning path.

Probability of Necessity (PN)

Intuition

Measures if a specific reasoning step s_t is **required** for producing the correct answer **y**.

"If this step were corrupted or replaced, would the answer become incorrect?"

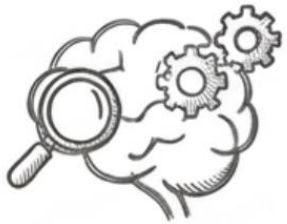
Interpretation

The probability that the answer would become incorrect if step s_t were replaced by an incorrect alternative $\tilde{s}_t$.

It quantifies how **critical** a single step is to the validity of the overall reasoning chain.

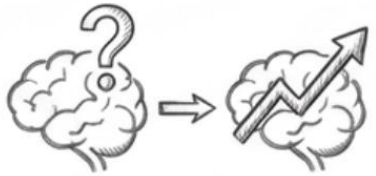
What should we do after calculating PNS

Causal Diagnosis (PNS)



PNS tells us which reasoning steps matter.

Policy Improvement (Learning)



How should the model learn from this diagnosis?

What Matters

- What was wrong
- Why it was wrong

How to Learn

- How to repair the trajectory
- What strategy should be internalized

PNS Framework: Probability of Necessity and Sufficiency

Quantify the causal value of each reasoning step.

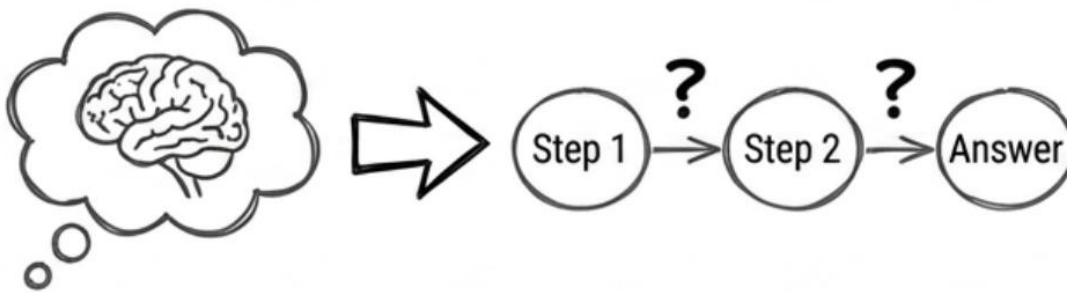
Core Components

Logic judgment: Judge if step is sufficient/necessary.

Impact quantification: Measure causal contribution via intervention.

Auto-optimization: Add missing steps, remove redundant steps.

Key Insight



This reframes CoT optimization as **causal credit assignment** over language reasoning steps.

Causal Credit Assignment

PNS Framework

Causal Sufficiency and Necessity Improves Chain-of-Thought Reasoning. NeurIPS 2025

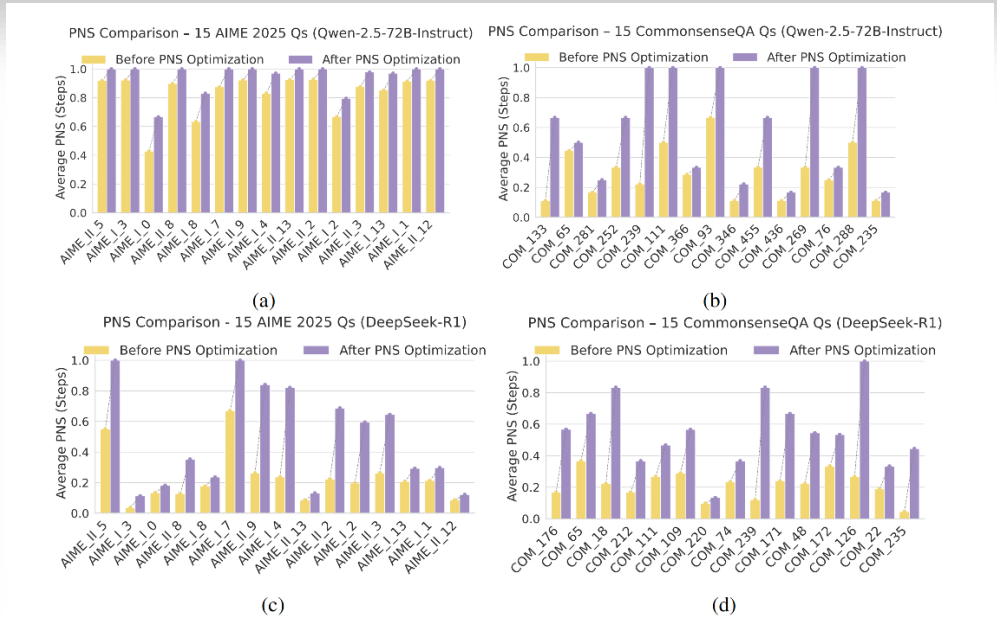
Result 1: PNS optimization of CoT trajectories

Key Finding: Our PNS optimization algorithm significantly reduces token consumption and reasoning steps while substantially improving final reasoning accuracy.

Test Data (Qwen-2.5-72B-Instruct @ GSM-8k)

⚡ **Baseline (Initial)**
Tokens: 113.8 | Steps: 8.1
Accuracy: 90.0%

🚀 **Optimized (PNS-Opt)**
Tokens: **26.6 (↓76.6%)**
Steps: **2.0 (↓75.3%)**
Accuracy: **97.0% (↑7.0%)**



Conclusion: PNS optimization removes redundant reasoning chains, extracting critical core steps. It achieves a comprehensive leap in reasoning performance while reducing computational costs and latency.

Causal Sufficiency and Necessity Improves Chain-of-Thought Reasoning. NeurIPS 2025

Result 2: enhancing non-reasoning models via Fine-tuning

Key Finding: Fine-tuning on the high-quality CoT dataset filtered by PNS enables the model to spontaneously generate more concise and accurate CoTs during inference, optimizing both reasoning efficiency and accuracy.

Original (Baseline)

Reasoning Steps: **77.4**

Accuracy (Acc): **84.6%**

Noncausal (Regular)

Reasoning Steps: **52.4**

Accuracy (Acc): **82.8%** (Slight Drop)

Causal (Ours)

Reasoning Steps: **36.3** (↓53.1%)

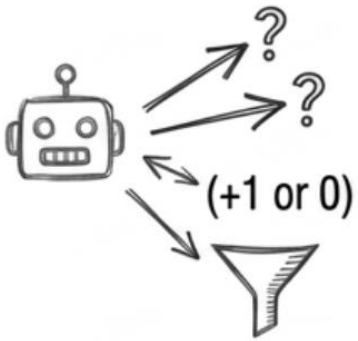
Accuracy: **91.7%** (↑8.4%)

Key Advantages Summary: The fine-tuned model not only drastically reduces reasoning steps (enhancing efficiency) but also improves accuracy, achieving comprehensive optimization. Most importantly, these advantages occur automatically during inference without additional computation cost.

Causal Sufficiency and Necessity Improves Chain-of-Thought Reasoning. NeurIPS 2025

A scalar reward is too poor for credit assignment

Traditional RL
(Sample & Filter/Amplify)



Sample trajectories.
Receive single scalar reward.
Upweight good trajectories.

Problem: Weak signal.
Doesn't explain "why" or
"how to fix".

Weak Learning Signal (Scalar)

Q1: What makes RL finally work?

- Better base model – **sample better trajectories**
- Better exploration – **explore new trajectories**
- Not pure pretraining strategy
  - Most reproductions are on Qwen models
  - Mixed raw and CoT data in the pretraining, or even long CoT data

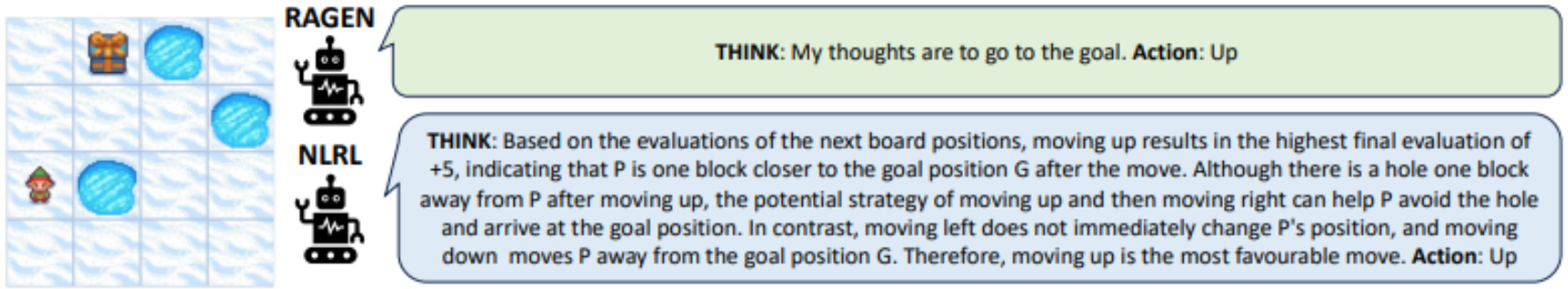
The dilemma of RL
You cannot sample it, you cannot optimize it

https://x.com/its_dibya/status/1883595705736163727

Q2: What makes (outcome-based) RL finally work?

- An implicit assumption that
  - Better outcome reward brings in stronger intermediate reasoning
  - True for Math and Code, where strong CoT prior preserves

- But may not hold for domains with less CoT training:
  - E.g., reward hacking by simply guessing the answer with non-sense CoT



https://x.com/its_dibya/status/1883595705736163727

Is it truly emergence in DeepSeek-R1 RL training?

- Reflection, branching, and self-correction data exist on the Internet

```
...

I thought that let us fix of getting a third head at last that is at 10th flip, so
    that we would stop there, and the remaining - getting two heads can be
    accommodated in the 9 trials. so there are $$9$$ choose 2 ways of getting two
    heads so the probability that we stop flipping after exactly ten flips is $$^9C_
    {2}$$ . $$\frac{1}{4}^3$$.$$\frac{3}{4}^7$$. Is this
    correct?

EDIT - Now the probability of getting exactly 3 heads? I got it to be $$^{10} C_{3}
    \frac{1}{4}^3 \frac{3}{4}^7$$. Should we get the same as the previous one? any
    reason why they should/should not be same?
```

- Aha moment appears at the base model

There May Not be Aha Moment in R1-Zero-like Training — A Pilot Study

Zichen Liu*, Changyu Chen*, Wenjun Li*, Tianyu Pang, Chao Du, Min Lin

* Equal contributions.

07 Feb, 2025

Codes: https://github.com/sail-sg/oat-zero

https://arxiv.org/pdf/2502.03373; https://oatllm.notion.site/oat-zero

Understanding Q1: policy gradient's passivity

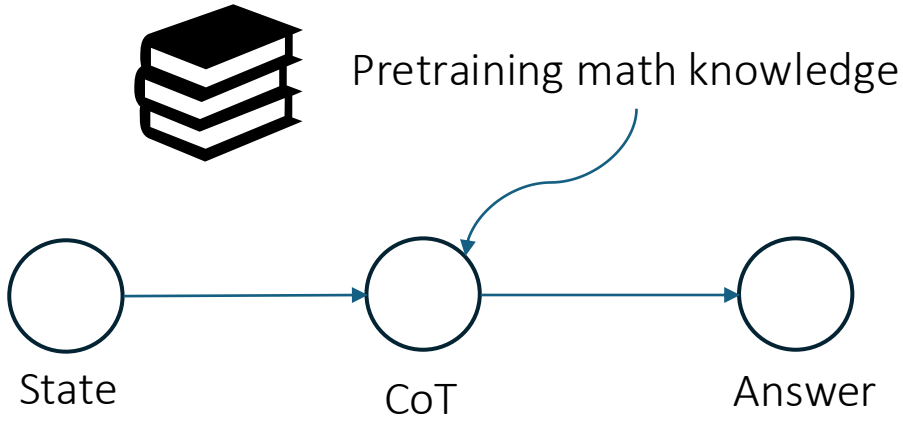
- Policy gradient is a bit passive!
- Physical meaning of policy gradient
  - Sample, Score, Optimize
  - Increasing probability of good action, decreasing probability of bad action
  - Kind of brute-force and passive

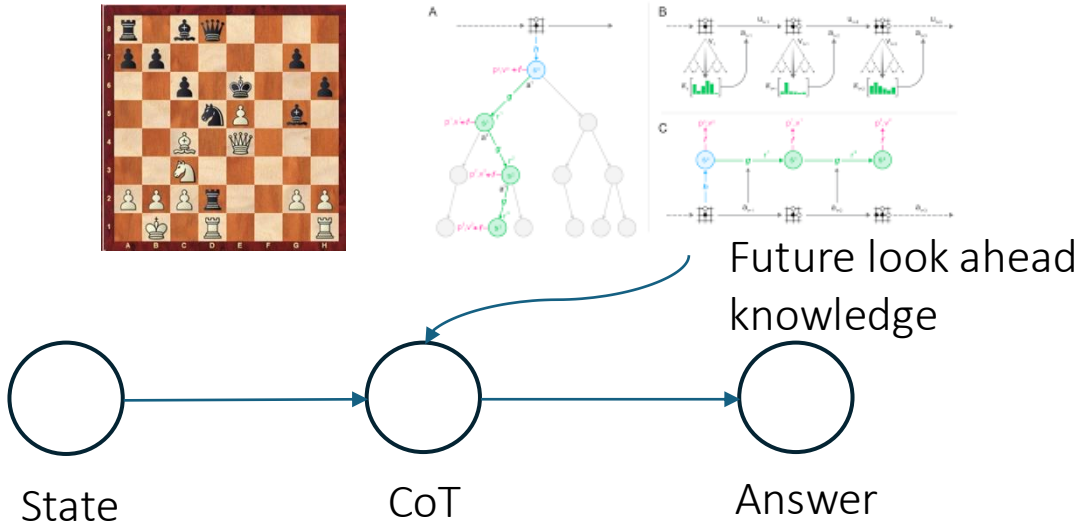
$$\nabla_\theta V_{\pi_\theta}(s_0)|_{\theta=\theta_{\text{old}}} = \mathbb{E}_{(\mathrm{s},\mathrm{a})\sim P_{\pi_{\theta_{\text{old}}}}}\left[\nabla_\theta \log \pi_\theta(a|s) Q_{\pi_{\theta_{\text{old}}}}(\mathrm{s},\mathrm{a})\Big|_{\theta=\theta_{\text{old}}}\right],$$

  - If you don't sample good (state, action) pairs, RL will fail

- Core issue – one dimensional scalar value function
  - Given only one number, that is the best policy gradient can do given scalar value!

Understanding Q2: forward reasoning and backward planning





- Math mostly involves forward reasoning
  - All conditions are given in the question
  - CoT is continuously retrieving math knowledge/reasoning learned from pre-training

- Agentic tasks requires backward planning and understanding
  - No environment information in pretraining – forward reasoning fails
  - Backward planning to understand the environment and gain knowledge

Potential solution to solve both issues

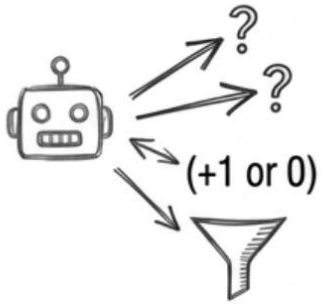
- How humans learn from experience?
- Humans learn in **a more deliberative way**
  - We won’t sample thousands/millions of trajectories or blindly optimize the probability with pure score
  - We actively analyze what happens specifically in the trajectories/data
  - Not only it’s outcome
- Language reward and explanation to improve the model.

Andrej Karpathy ✓
@karpathy

Scaling up RL is all the rage right now, I had a chat with a friend about it yesterday. I'm fairly certain RL will continue to yield more intermediate gains, but I also don't expect it to be the full story. RL is basically *"hey this happened to go well (/poorly), let me slightly increase (/decrease) the probability of every action I took for the future"*. You get a lot more leverage from verifier functions than explicit supervision, this is great. But first, it looks suspicious asymptotically - once the tasks grow to be minutes/hours of interaction long, you're really going to do all that work just to learn a single scalar outcome at the very end, to directly weight the gradient? Beyond asymptotics and second, this doesn't feel like the human mechanism of improvement for majority of intelligence tasks. There's significantly more bits of supervision we extract per rollout via a review/reflect stage along the lines of *"what went well? what didn't go so well? what should I try next time?"* etc. and the lessons from this stage feel explicit, like a new string to be added to the system prompt for the future, optionally to be distilled into weights (/intuition) later a bit like sleep. In English, we say something becomes "second nature" via this process, and we're missing learning paradigms like this. The new Memory feature is maybe a primordial version of this in ChatGPT, though

A scalar reward is too poor for credit assignment

Traditional RL (Sample & Filter/Amplify)



Sample trajectories.
Receive single scalar reward.
Upweight good trajectories.

Problem: Weak signal.
Doesn't explain "why" or "how to fix".

Weak Learning Signal (Scalar)

Key Insight: Language as Signal

LLMs **reason** in **language**, **fail** in **language**, and can **improve** through **language.**

⇨ So the learning signal should also be **language.**

Reason (Language) → **Fail** (Language) → **Improve** (Language)

Rich Learning Signal (Language)

Natural Language Reinforcement Learning. arXiv:2411.14251

Language Value Function: storing the “why” behind success

Scalar Value Function

How good is this state?

0.8 Numeric reward prediction

⇨ Weak credit assignment

Language Value Function (The “Why”)

Why is this state good or bad?

Strategy-level explanation

Rich causal / semantic credit assignment

Numeric Prediction

Strategic Explanation

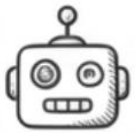
Key Insight: Experiential Memory as Language Value

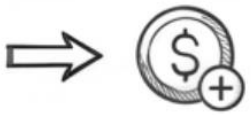
What people now call **experiential memory** for LLM agents is very close to this idea: not just retrieving past episodes, but storing **structured lessons about why a trajectory worked**.

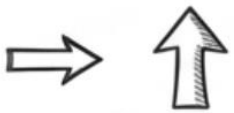
Natural Language Reinforcement Learning. arXiv:2411.14251

Language Policy Improvement: from feedback to better trajectories

Old Recipe (Scalar RL)

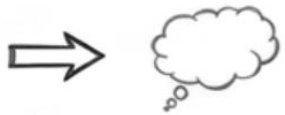

Sample


Reward (+/-)


Upweight

New Recipe (Language Policy Improvement)


Attempt


Language Feedback


Corrected Trajectory (Hindsight)


Distillation

Scalar Feedback
(Simple Penalty)

Language Feedback
(Correction & Distillation)

Key Insight: From Penalty to Correction

Instead of sample → reward → upweight, the new recipe is:
attempt → language feedback → corrected trajectory → distillation.

Natural Language Reinforcement Learning. arXiv:2411.14251

NLRL as an early attempt at **feedback-aware self-distillation**

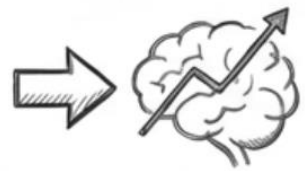
NLRL's Core Idea (Self-Distillation)

Experience -> Language -> Policy -> Internalization

Feedback-Aware Self-Distillation

Connection to PNS (Structured Feedback)

PNS: Identifies *which* reasoning steps are necessary/sufficient.

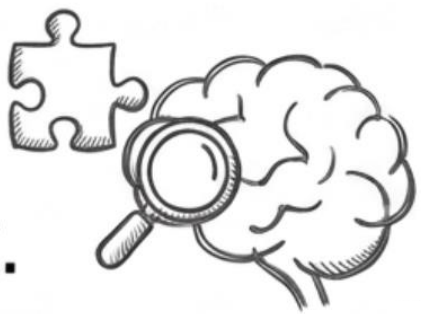


NLRL: Tells *how* to turn this into learning for future behavior.

Structured Feedback to Learning Signal

Key Insight: The Link

PNS provides the **structured feedback** (what).
NLRL provides the **mechanism** (how) to internalize it.



NLRL as an early attempt at **feedback-aware self-distillation**

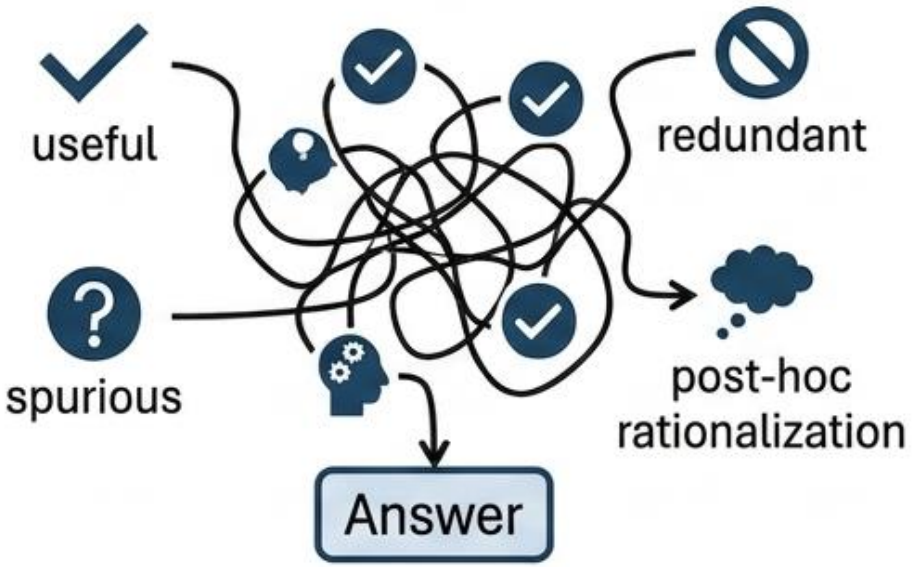
Next step: A model that stores and updates these lessons becomes a causal world model.

Causality tells us **what** to learn.
Language tells the model **how** to learn it.

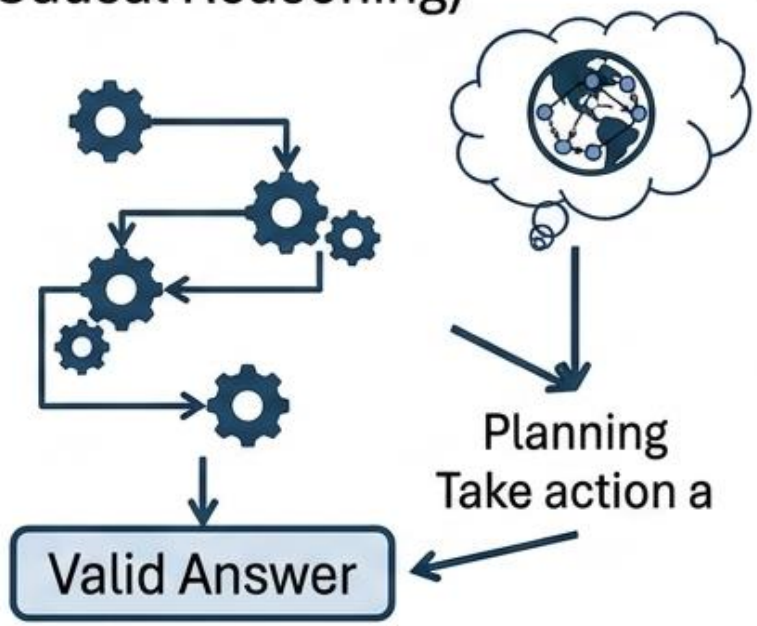
Causality for LLM Reasoning

Can we identify which reasoning steps are **truly causally** responsible for the final answer?

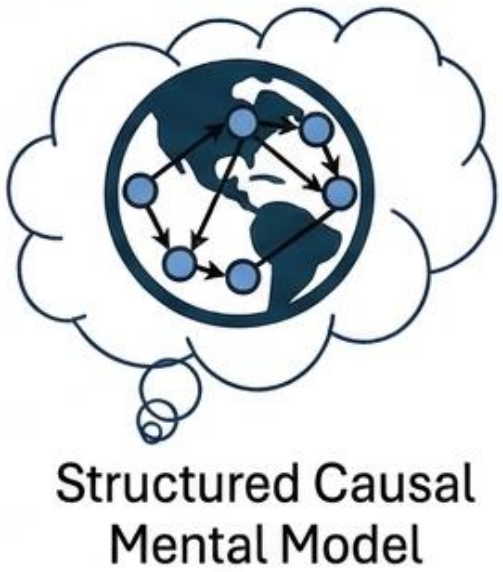
Model Free
(Chain of Thought)



Modelling the world
(Causal Reasoning)



Long-term Goal:
Causal World Models

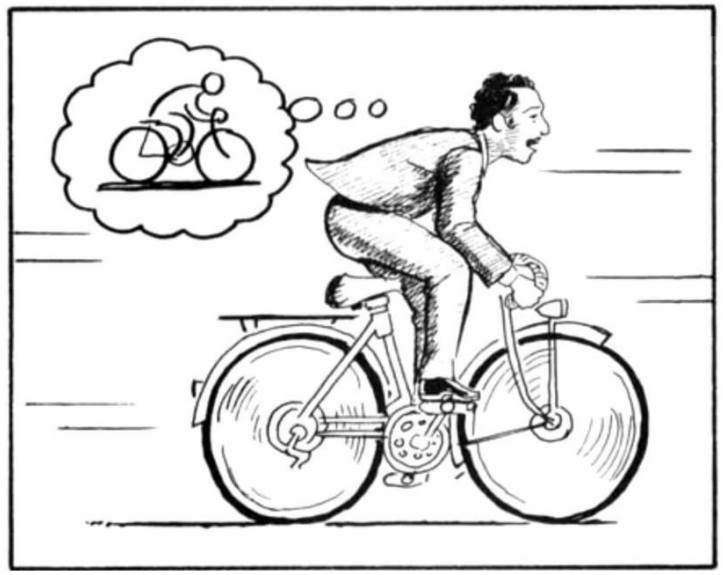


Sufficiency and necessity:
Which steps truly support the answer?

Intervention & Counterfactuals:
Can the model judge 'if this step changed, would the conclusion change?'

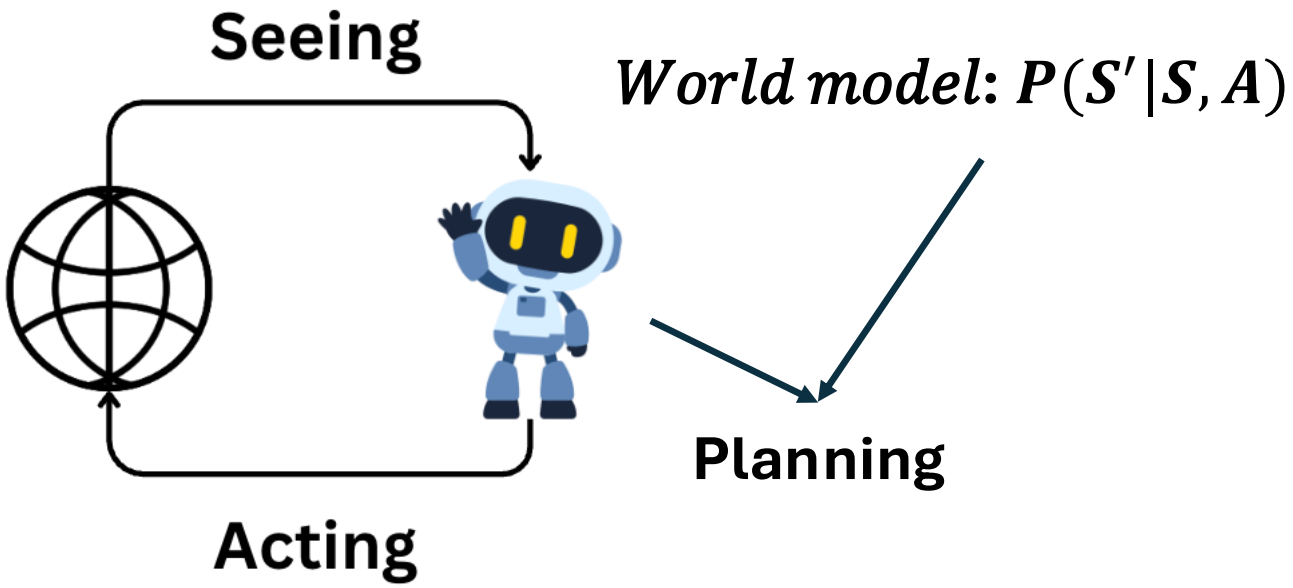
What is World Models

World Models modelling the **environment internal transition model**, makes agent 1. predict the future and 2. reflect the past possible.

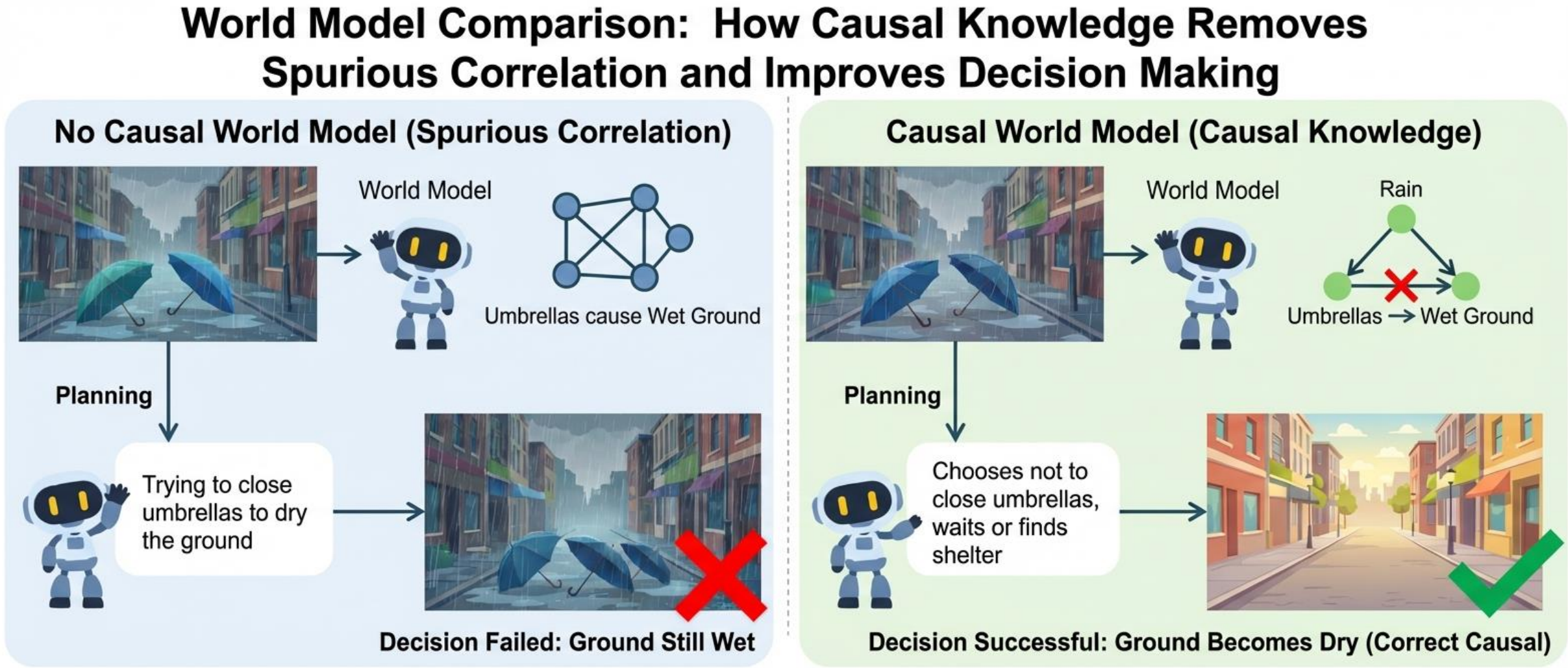


What if I ride slower now? Can I achieve the goal?
Was it a good choice to start earlier?

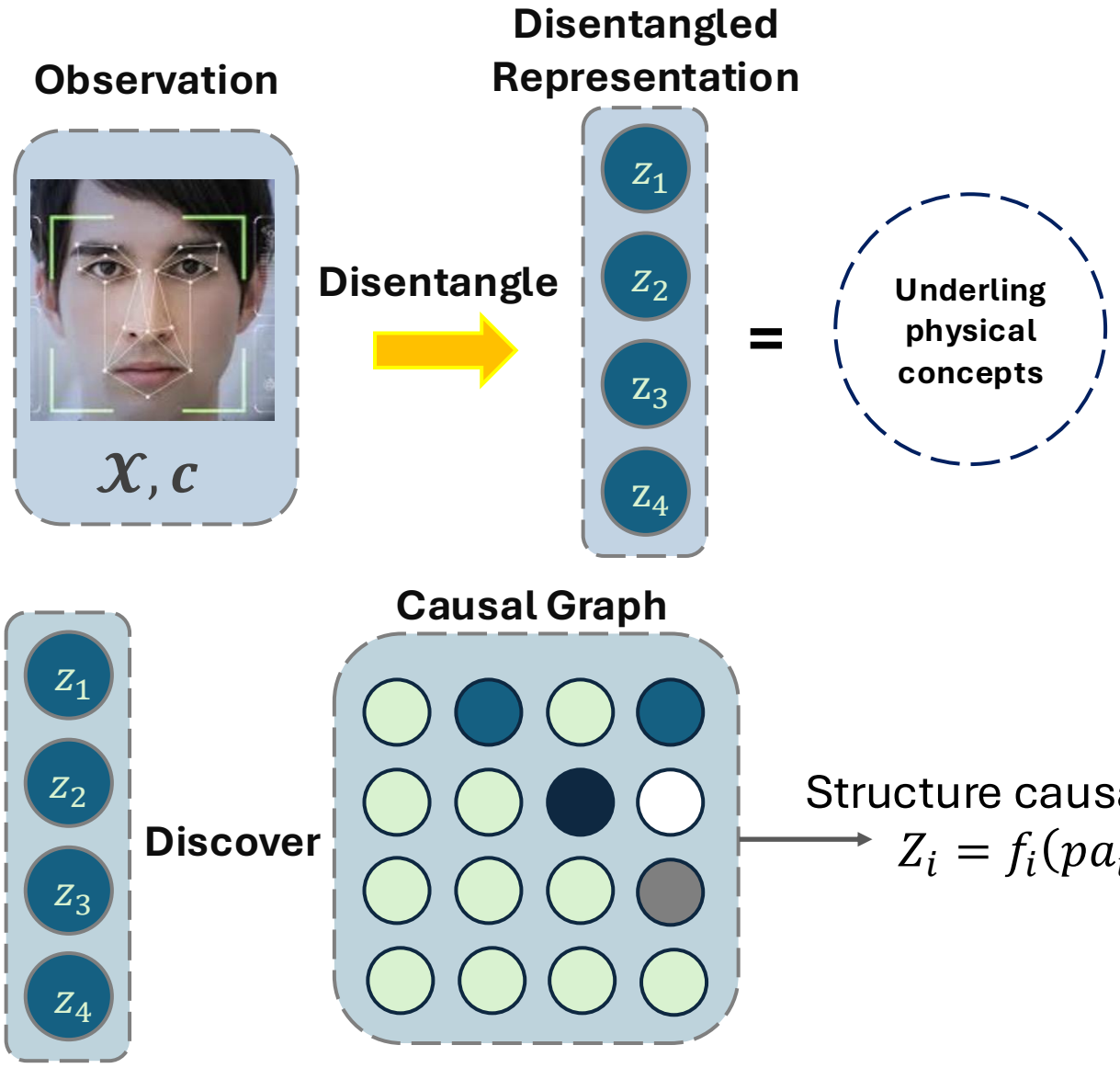
Picture borrow from Ha and Schmidhuber



Causal World Models



Causal World Models – Representation + Causal Structure



Intervene causes Smile, Mouth Open will change

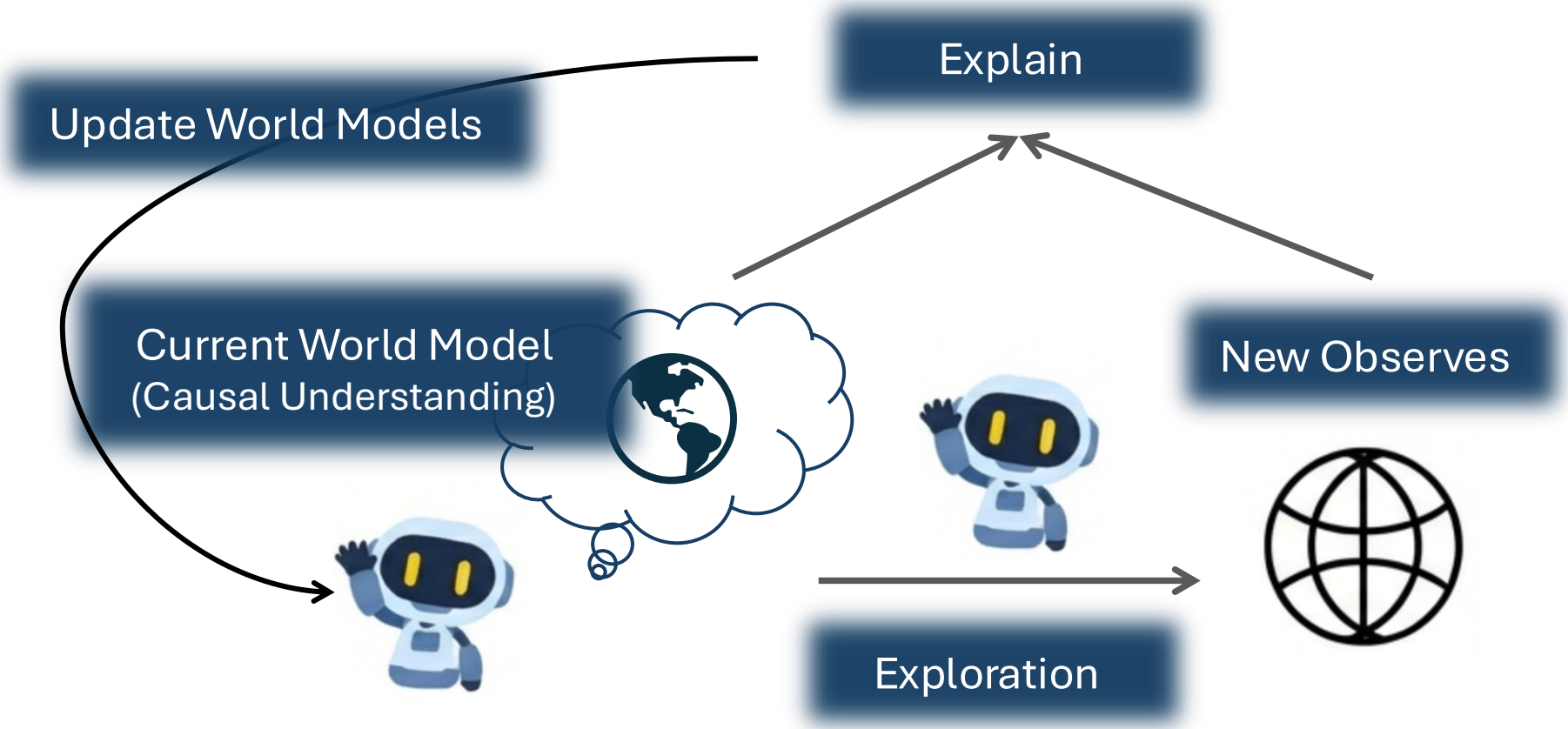
Smile = -0.75

But Intervene effect concept Mouth Open, Smile will not be influenced

Mouth Open = -0.75

CausalVAE: Disentangled representation learning via neural structural causal models. CVPR 2021

Continual Causal Learning in Open-Ended Worlds



Causal Discovery in Open-Ended World

[NeurIPS 2025]
Curious Causality-Seeking Agents in Open-Ended World





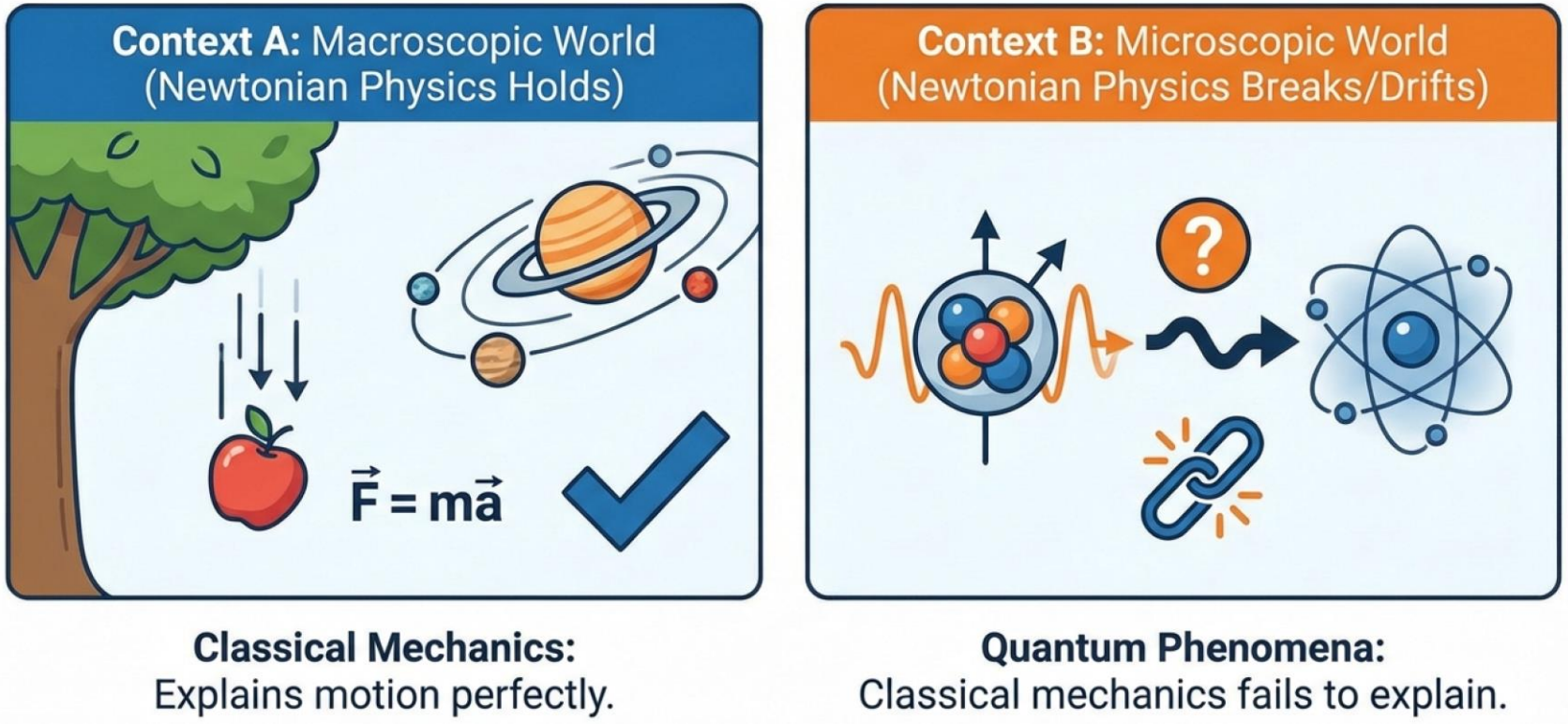


Zhiyu Zhao, Haoxuan Li, Haifeng Zhang, Jun Wang, Francesco Faccio, Jürgen Schmidhuber, Mengyue Yang

Motivation: Causal “Drift” in Open Worlds

Causal Discovery in Open-Ended World

Causal “Drift” in Open Worlds



Curious Causality-Seeking Agents in Open-Ended World. NeurIPS 2025

Future Outlook: Towards Generalizable Causal Representations

Current: Task-Specific Causal Learning
(RL Exploration)

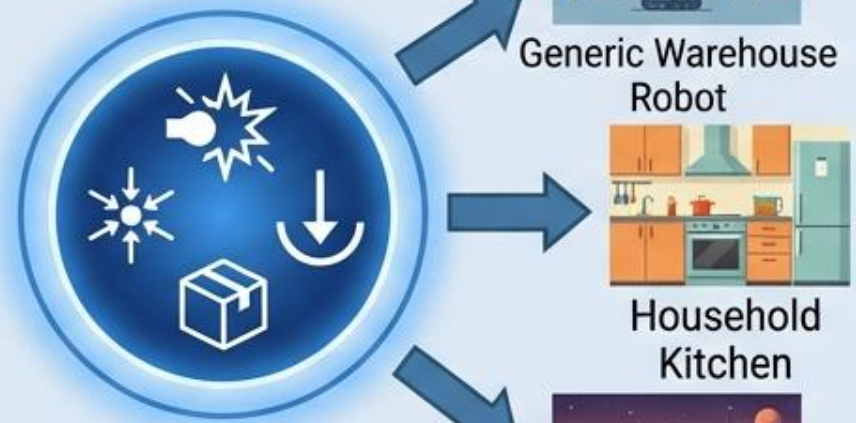
Football AI

Chat Agent/ Driving

Limited transferability. Re-learns basic physics from scratch for each task.

Goal: Learn Universal Causal Primitives

Future: Generalizable Causal Foundation



Generic Warehouse Robot

Household Kitchen

Sci-Fi Planet Rover

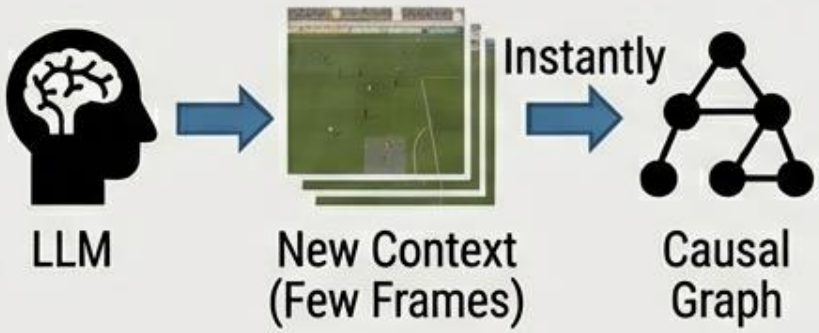
Learn invariant mechanisms across domains

Enable zero-shot generalization to new environments.

Moving from learning **'how to play football'** to learning **'how objects interact physically'**.

Future Outlook: Causal Foundation World Models & Scaling

In-Context Causal Learning



Leverage LLM reasoning for fast, in-context causal discovery

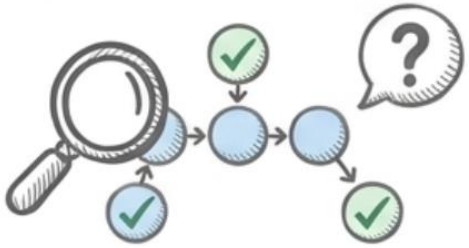
The Scaling Hypothesis

Causal understanding emerges and robustifies with large-scale pre-training.

Final Goal: A unified, scalable world model with robust causal reasoning capabilities.

Take Away

1. Beyond Fluency: Causal Contribution



Judge reasoning steps by their **causal contribution** (sufficiency & necessity), not just final answer or fluency.

Causal Credit Assignment

2. Beyond Scalar: Language Feedback

LLMs reason and fail in **language**, so they must improve through **language feedback**, not just scalar rewards.

Language-Based Self-Improvement

3. Long-Term Goal: Causal World Models

Self-improving LLMs need **causal world models** to predict, reflect, and update their causal understanding.

Causal World Models

The future of LLM reasoning is not just longer chains of thought, but **causal credit assignment** and **language-based self-improvement**.