Reasoning: From the Lab to the Real World

Zeming Chen (Eric)

Debjit Paul

Antoine Bosselut

NLP applications

Machine Translation

Search Engines

Personal Assistants

Question Answer & ChatBoT

NLP applications

Generative AI empowering real-world NLP applications

Machine Translation

Search Engines

Personal Assistants

Question Answer & ChatBoT

Are they truly intelligent systems?

Machine Translation

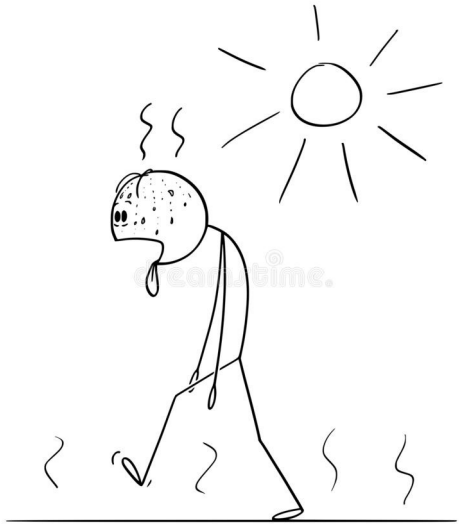
Search Engines

Personal Assistants

Question Answer & ChatBoT

Machines that understand humans

Peter walked for 5 km in the ***heat***.
He went home and shook his water bottle.
He was ***disappointed*** when it made ***no sound***.



Leave out important information

Rely on shared intuitive inferences to understand context

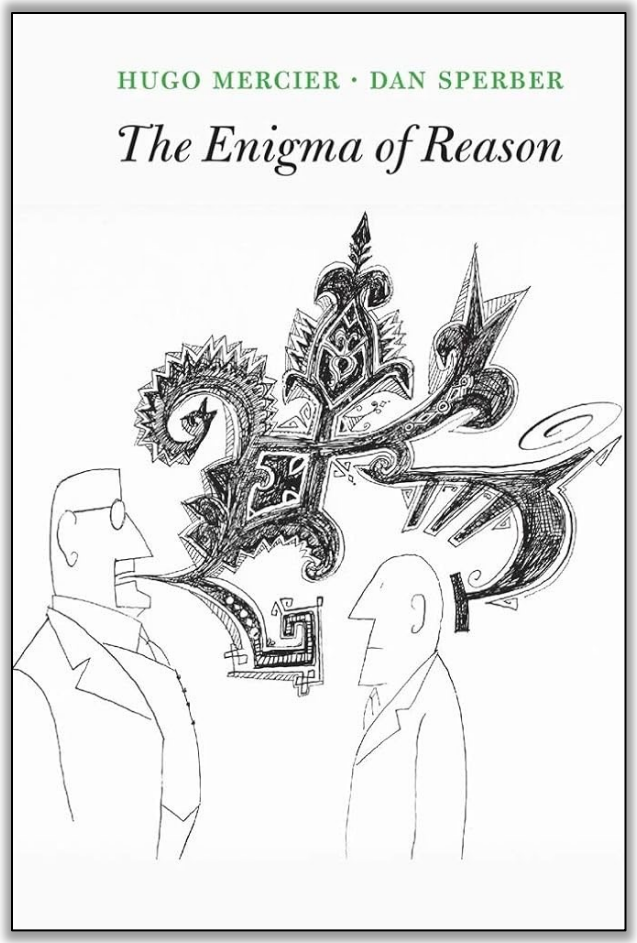
Reason about knowledge to reach understanding

How do humans use reasoning?

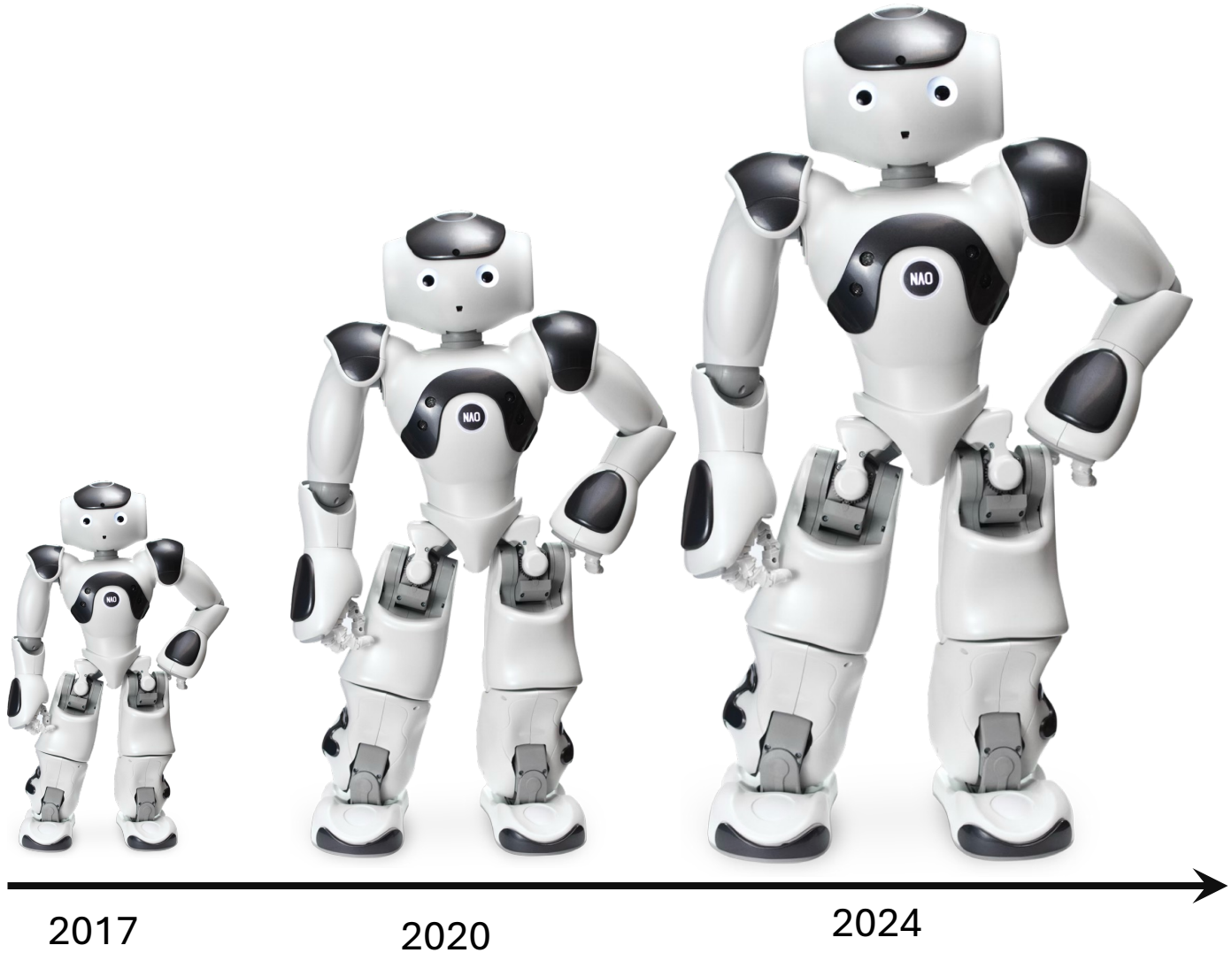
- Reasoning is tightly coupled with **explanation and justification** (Evans and Wason, 1976)

- Humans use reasoning to:
  - justify our intuitive inferences to convince others of our arguments

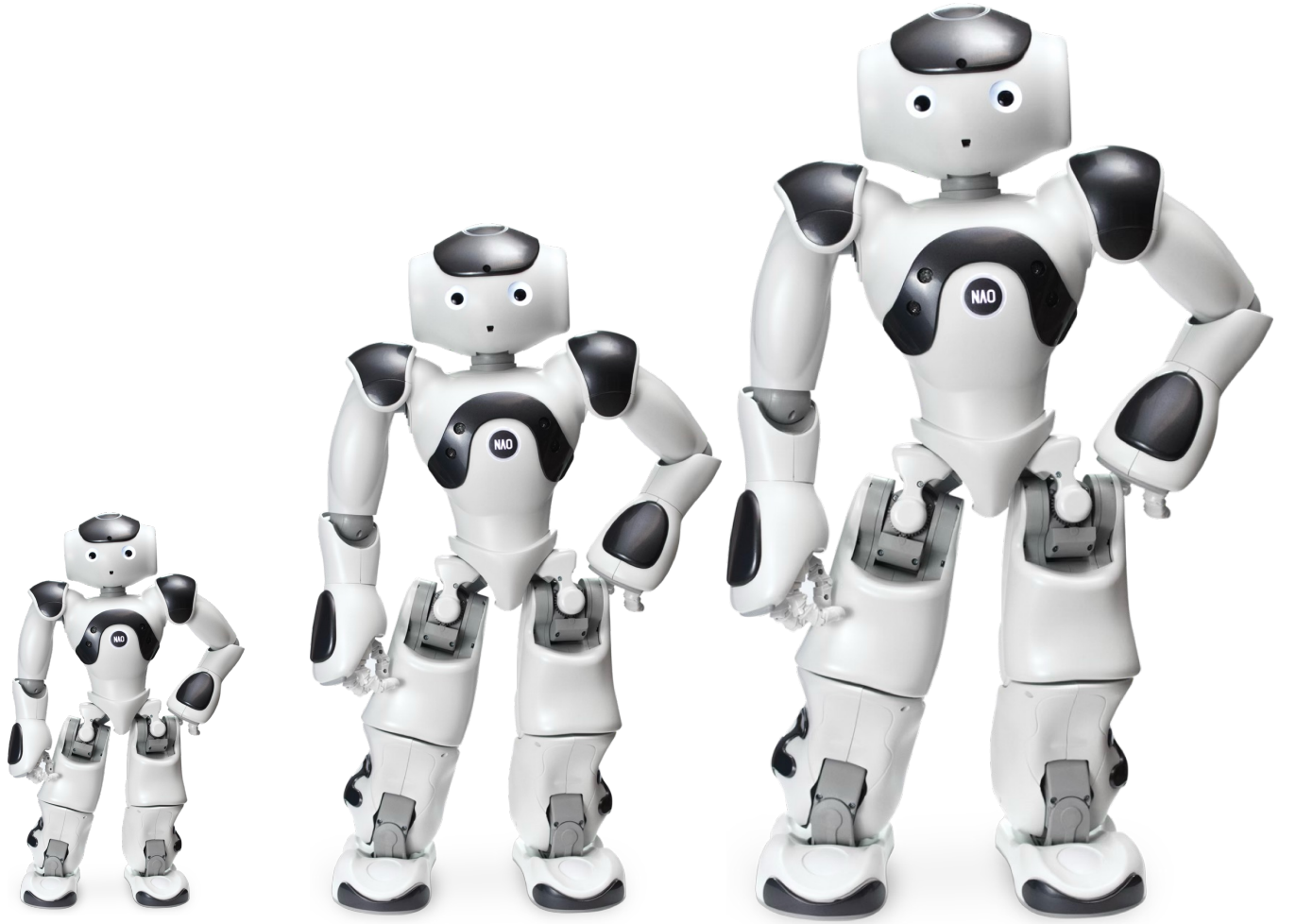
  - evaluate the arguments made by others

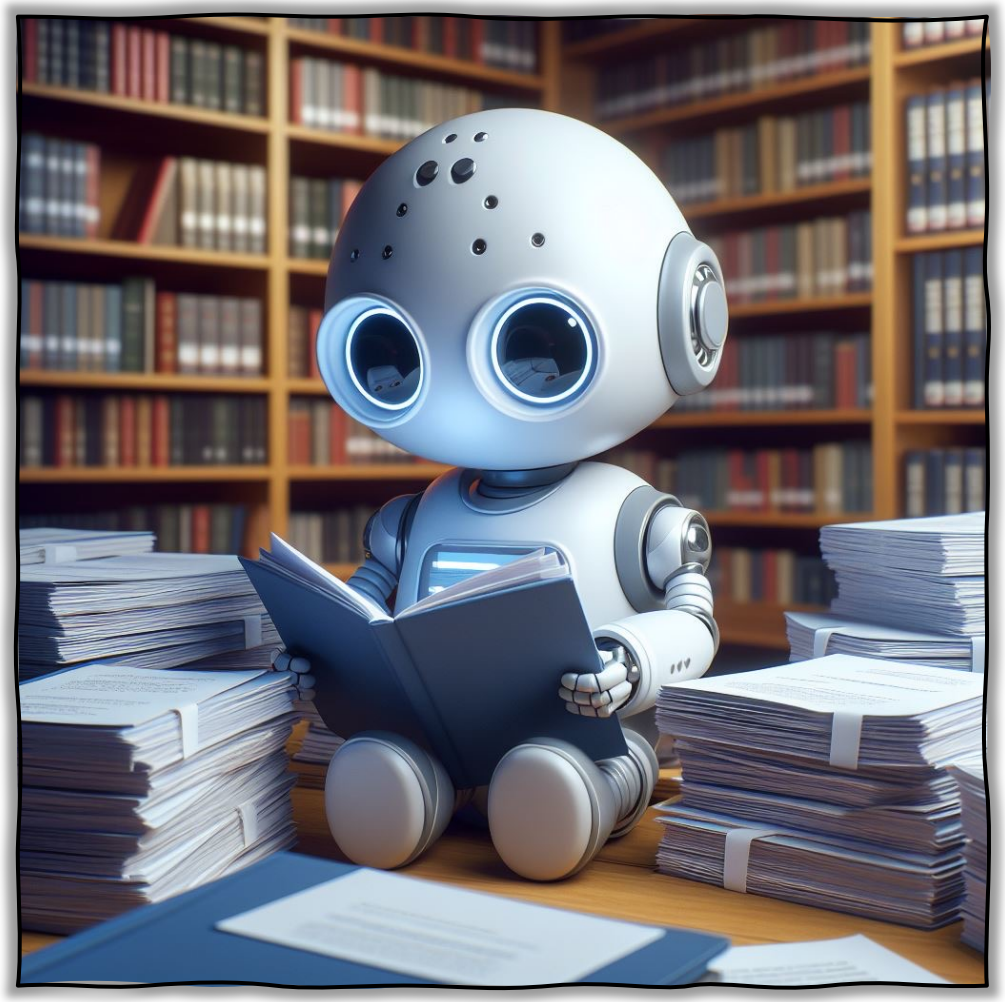


Bigger and better

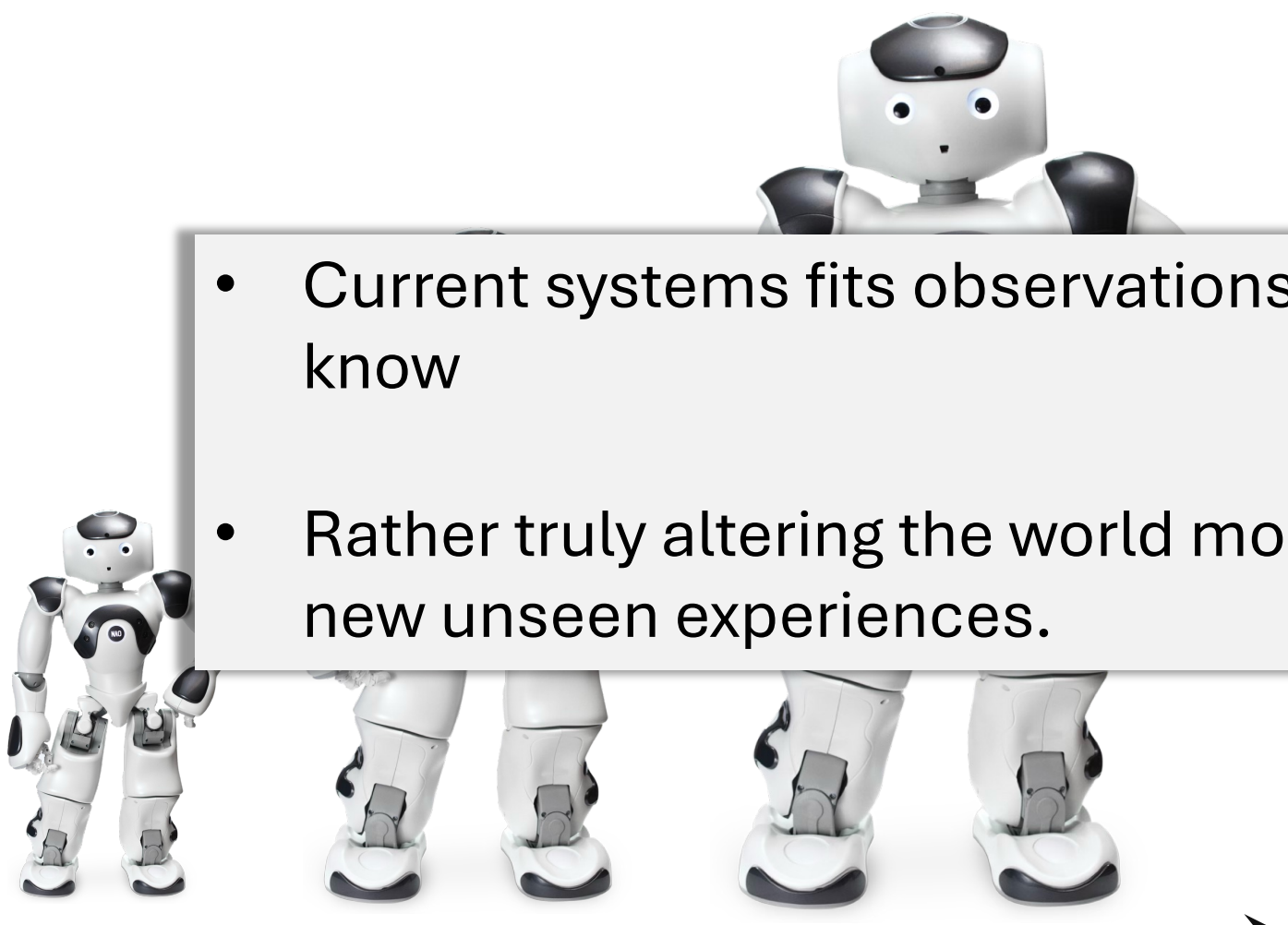


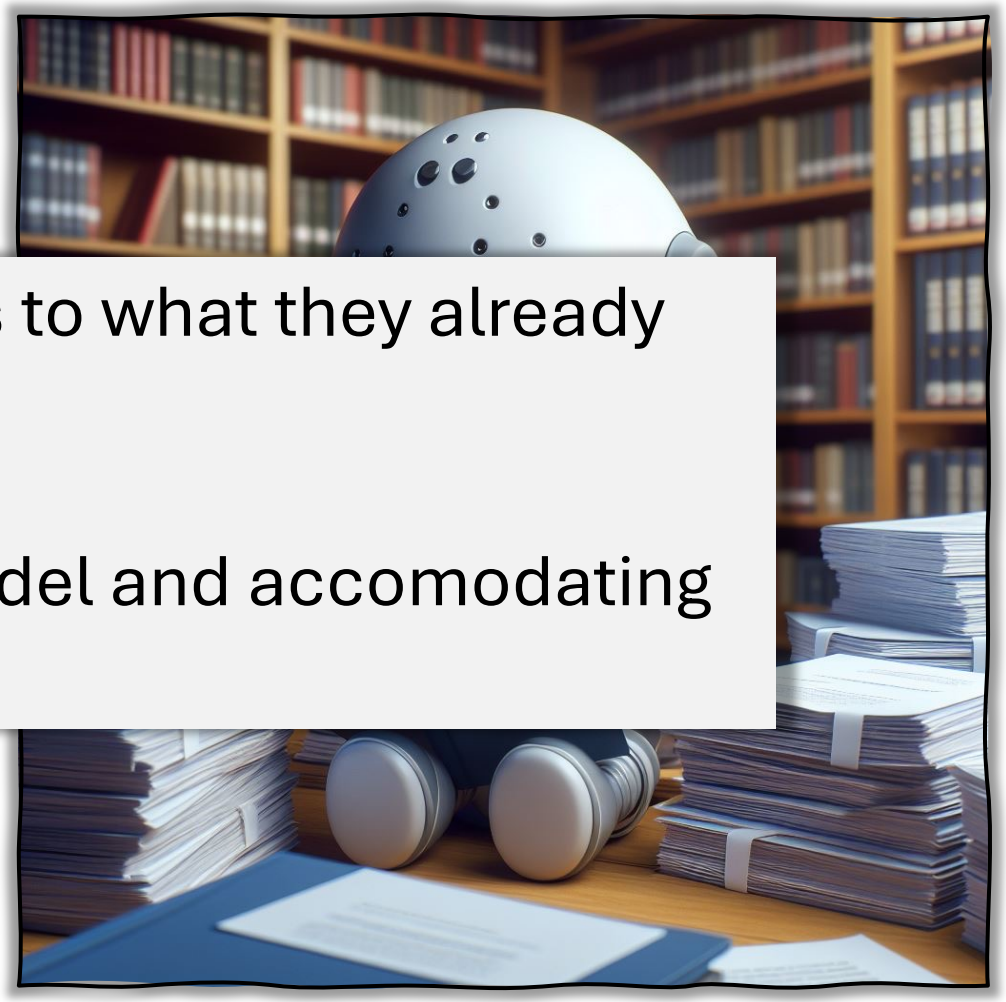
More knowledge





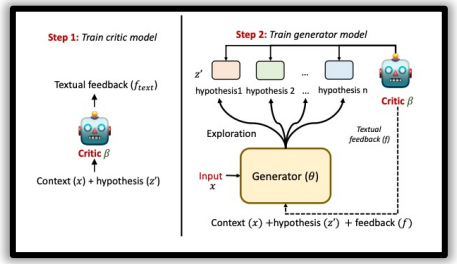
Generative AI lacks reasoning





- Current systems fits observations to what they already know
- Rather truly altering the world model and accomodating new unseen experiences.

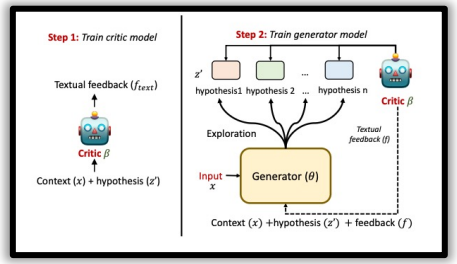
Outline



Refine knowledge using feedback

Paul et.al. 2024 (EACL)

Outline

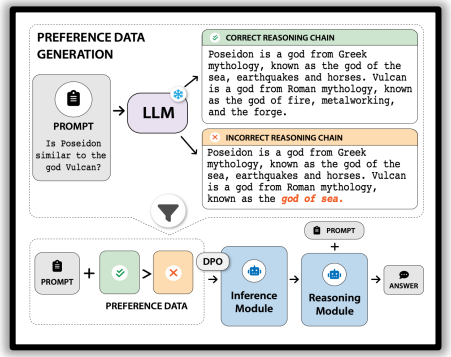


Refine knowledge using feedback

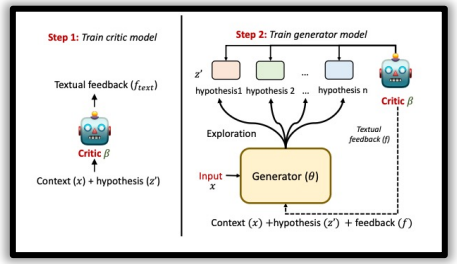
Paul et.al. 2024 (EACL)

Paul et.al. 2024 (EMNLP Findings)

Faithful Reasoning about knowledge

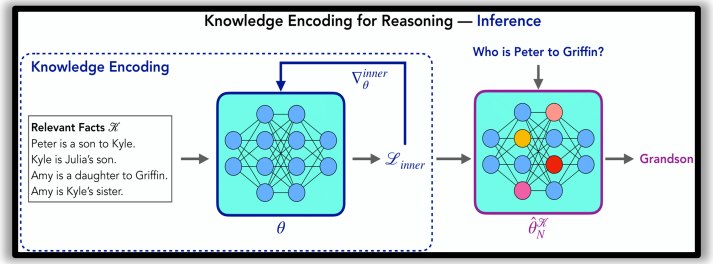


Outline



Refine knowledge using feedback

Paul et.al. 2024 (EACL)

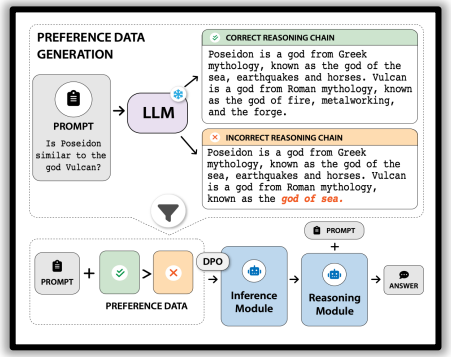


Robust Reasoning by updating world models

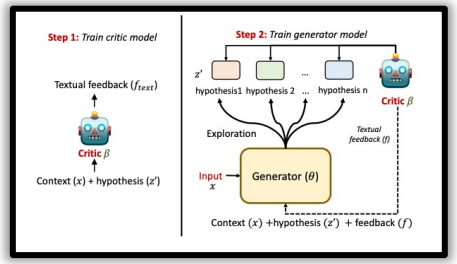
Chen et.al. 2023 (Neurips)

Paul et.al. 2024 (EMNLP Findings)

Faithful Reasoning about knowledge

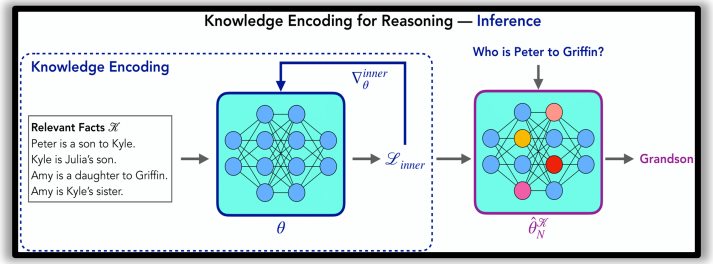


Outline



Refine knowledge using feedback

Paul et.al. 2024 (EACL)

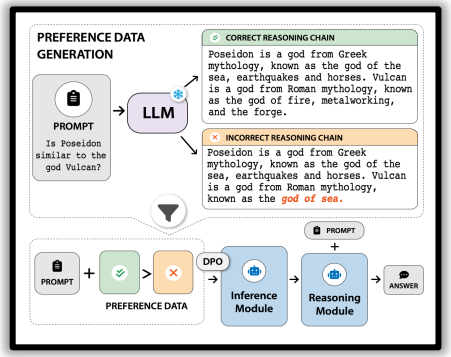


Robust Reasoning by updating world models

Chen et.al. 2023 (Neurips)

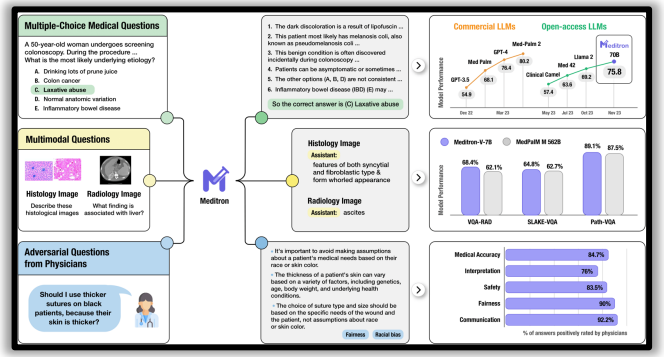
Paul et.al. 2024 (EMNLP Findings)

Faithful Reasoning about knowledge



Chen et.al. 2024

Real-World Reasoning



REFINER: Reasoning Feedback on Intermediate Representations







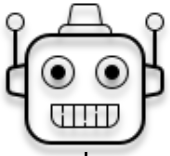
EACL 2024

Emergent abilities of large language models

- Chain-of-thought (CoT) prompting asks LLMs to generate reasoning traces before generating a final answer.

Wei et al. (2022)

Is travelling by plane more dangerous than traveling by car?
Let's Think Step by Step



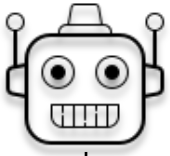
Step1: According to the National Safety Council, the lifetime odds of dying in a car accident are **1 in 102**.
Step2: The odds of dying in an air transport incident are substantially lower at **1 in 9,821**. **Answer: No**

Emergent abilities of large language models

- The reasoning steps described in the CoT can be incorrect and inconsistent

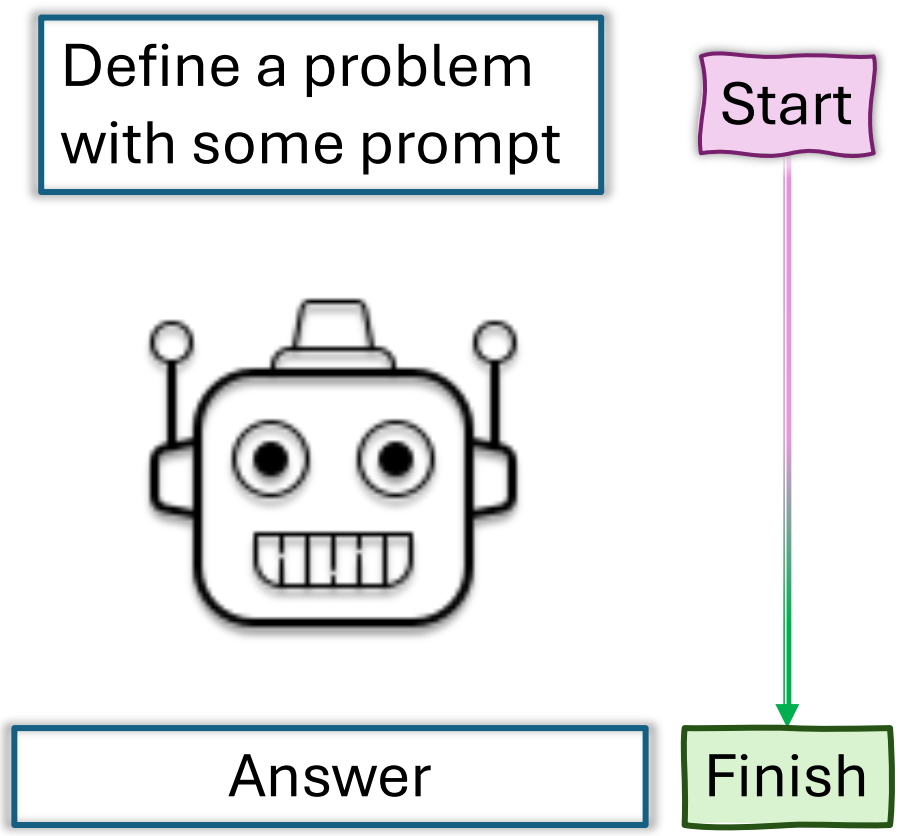
Turpin et al. (2023), Zhang et al. (2023)

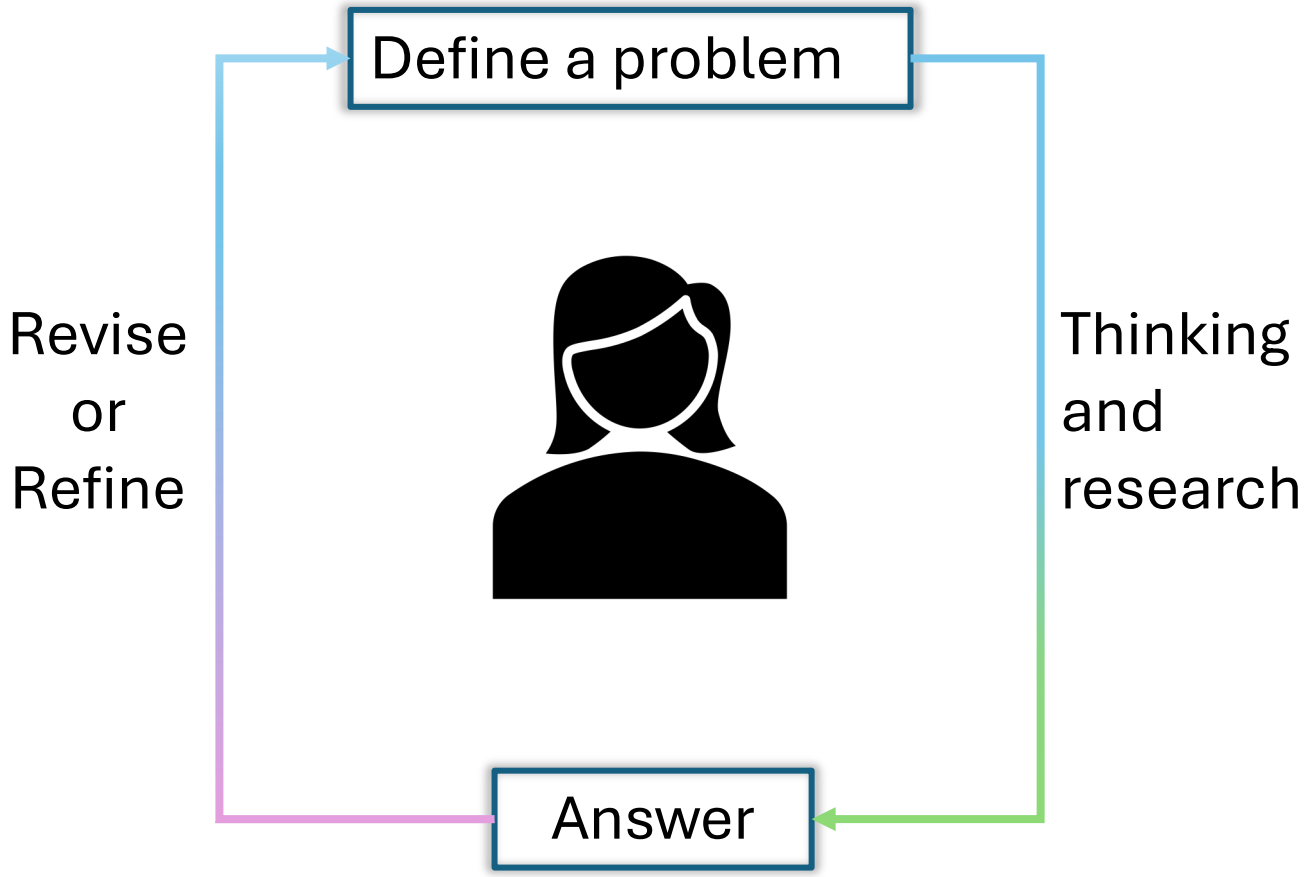
Is travelling by plane more dangerous than traveling by car?
Let's Think Step by Step



Step1: According to the National Safety Council, the lifetime odds of dying in a car accident are **1 in 102**.
Step2: The odds of dying in an air transport incident are substantially lower at **1 in 9** . **Answer: Yes**

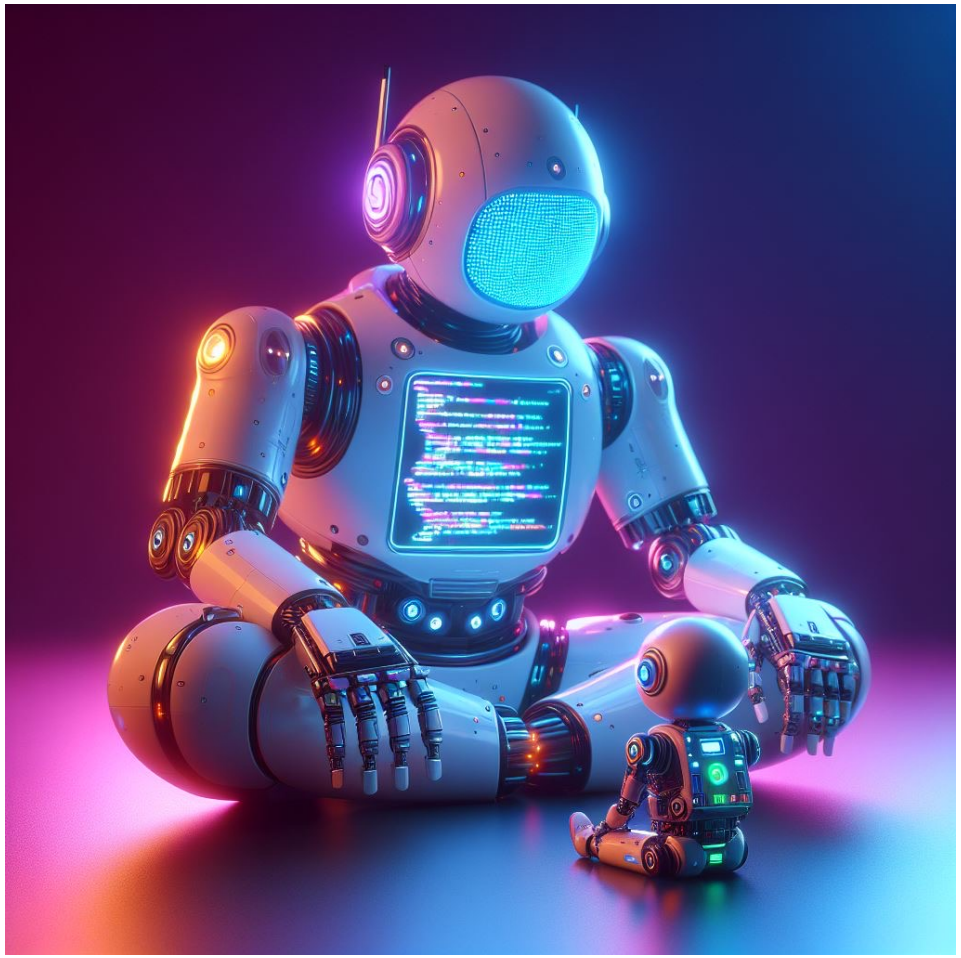
Recursive Reminding





Research Question

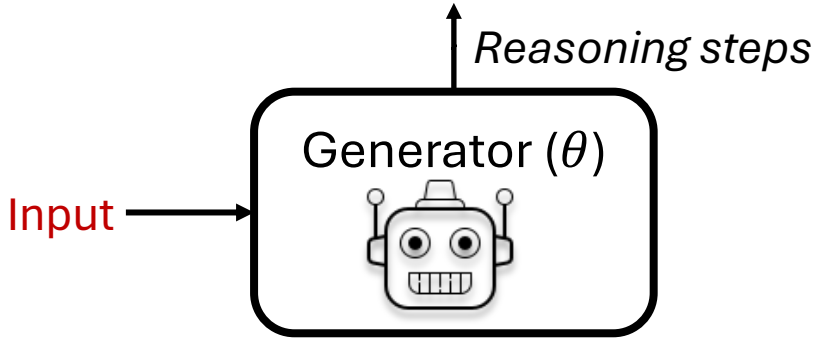
How do we build a method to refine the reasoning chains generated by LMs automatically?



Paul et al. EACL 2024

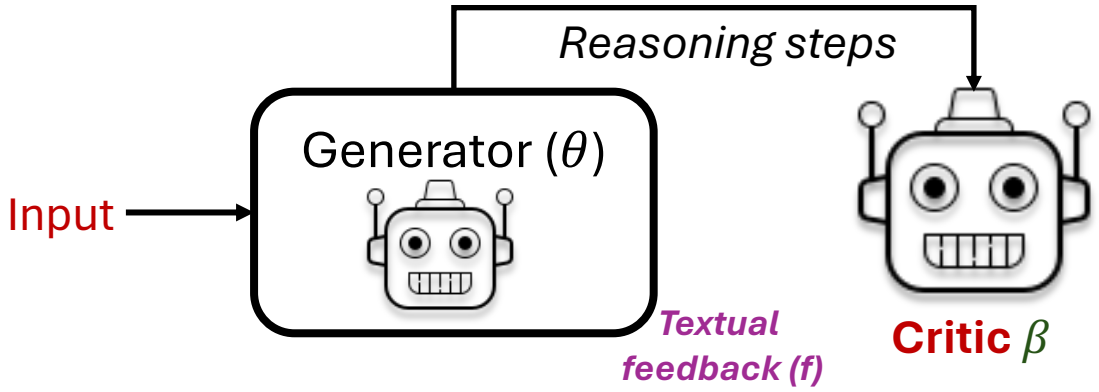
REFINER

- An interaction-based framework
  - ➢Iteratively refine the reasoning steps



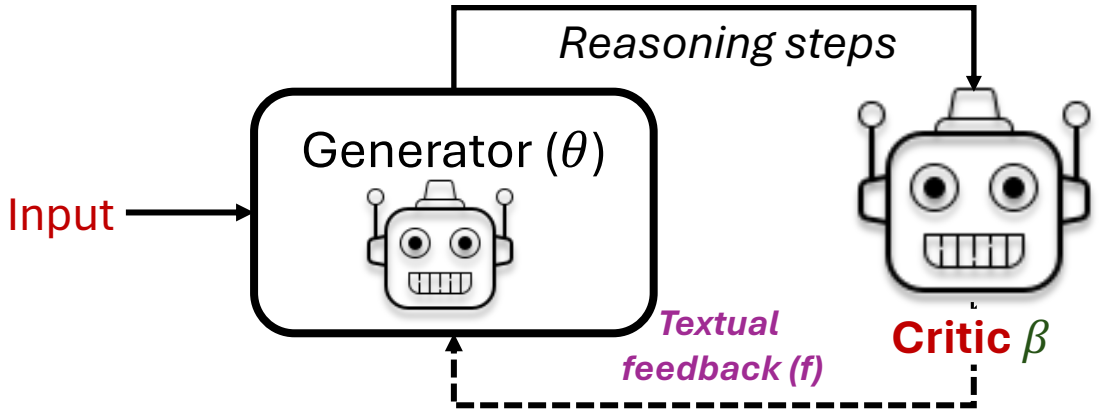
REFINER

- An interaction-based framework
  - ➢Iteratively refine the reasoning steps



REFINER

- An interaction-based framework
  - ➢Iteratively refine the reasoning steps



Algorithm 1 REFINER Training

for E epochs **do**
  for $i(batch) \leftarrow 1$ to N **do**
    Initialize (feedback) $f_0 \leftarrow No$
    for $t \leftarrow 1$ to T **do**
      $\hat{z}^k_{i,t} \sim \pi_\theta(y_i|c_i, f_{t-1}, \hat{z}_{i,t-1})$
      $f_t, \hat{z} \leftarrow \pi_\beta(c_i, z_i, \hat{z}^k_{i,t})$
      $\mathscr{L}^{lm}_i$ += $-\log p(z_i|c_i, f_{t-1}, \hat{z}_{i,t-1})$
    end for
  end for
end for
return π_θ

Feedback Data

- Identify different fine-grained reasoning errors.
- Create training data for the critic model by perturbing the correct reasoning steps.

Correct Reasoning Steps

Step1: First, you started with 10 apples.You gave away 2 apples to the neighbor and 2 to the repairman, so you had <<10-2-2>> 6 apples left.
Step2: Then you bought 5 more apples, so now you had 11 apples.

Feedback Data

- Introduce different reasoning errors and create incorrect reasoning steps.

Correct Reasoning Steps

Step1: First, you started with 10 apples.You gave away 2 apples to the neighbor and 2 to the repairman, so you had <<10-2-2>> 6 apples left.
Step2: Then you bought 5 more apples, so now you had 11 apples.

Incorrect Reasoning Steps

Step1: First, you started with 10 apples.You gave away 2 apples to the neighbor and 4 to the repairman, so you had <<10+2-4>> **10 apples left**.
Step2: Then you bought 5 more apples, so now you had <<10 + 5 >> 15 apples.

Feedback Data

- Based on the error types we create semi-structured textual feedback.

Correct Reasoning Steps

Step1: First, you started with 10 apples.You gave away 2 apples to the neighbor and 2 to the repairman, so you had <<10-2-2>> 6 apples left.
Step2: Then you bought 5 more apples, so now you had 11 apples.

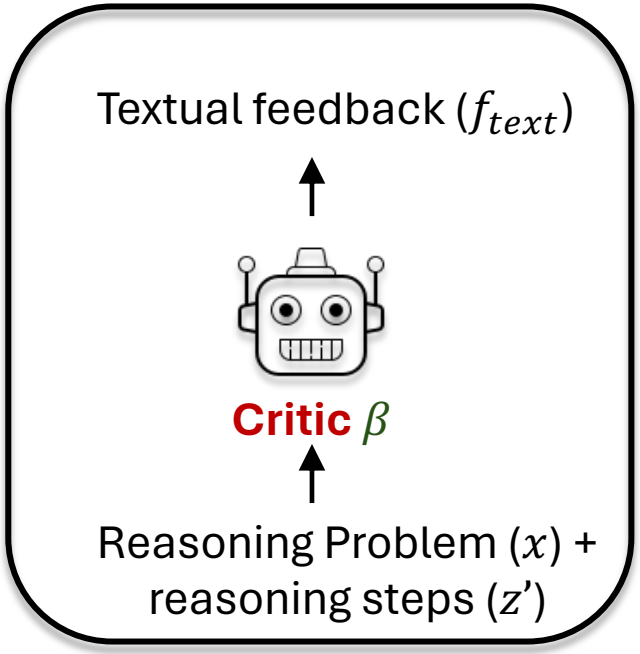
Incorrect Reasoning Steps

Step1: First, you started with 10 apples.You gave away 2 apples to the neighbor and 4 to the repairman, so you had <<10+2-4>> **10 apples left**.
Step2: Then you bought 5 more apples, so now you had <<10 + 5 >> 15 apples.

Textual Feedback: The operand and operator in the step 1 are incorrect.

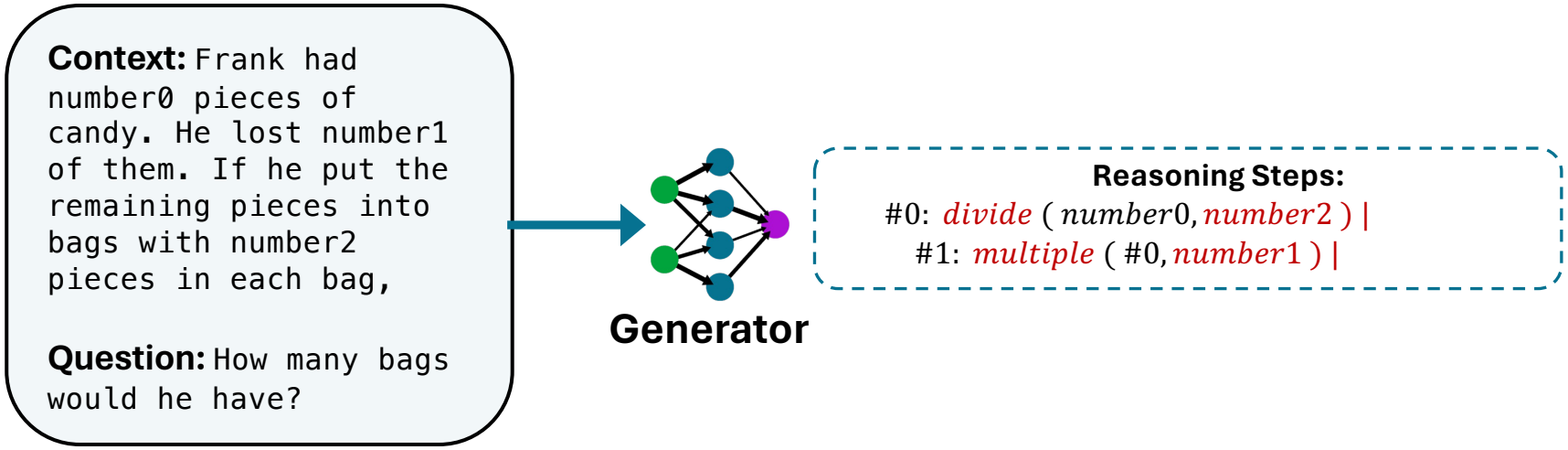
Train a critic model

- Critic model learns to provide textual feedback.
- We train a supervised model with the reasoning question (context) and correct or incorrect reasoning steps as input and the textual feedback as output.



$$\text{Loss} = -\sum_{i} logP(f_i^* | x, z)$$

REFINER



Algorithm 1 REFINER Training

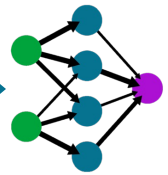
for E epochs **do**
  for $i(batch) \leftarrow 1$ to N **do**
    Initialize (feedback) $f_0 \leftarrow No$
    for $t \leftarrow 1$ to T **do**
      $\hat{z}^k_{i,t} \sim \pi_\theta(y_i|c_i, f_{t-1}, \hat{z}_{i,t-1})$
      $f_t, \hat{z} \leftarrow \pi_\beta(c_i, z_i, \hat{z}^k_{i,t})$
      $\mathscr{L}^{lm}_i$ += $-\log p(z_i|c_i, f_{t-1}, \hat{z}_{i,t-1})$
    end for
  end for
end for
return π_θ

REFINER – Example

Context: Frank had number0 pieces of candy. He lost number1 of them. If he put the remaining pieces into bags with number2 pieces in each bag,

Question: How many bags would he have?

Trainable



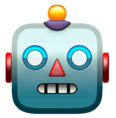
Generator

Reasoning Steps:

#0: *divide* (*number0*, *number2*) |
#1: *multiple* (#0, *number1*) |

Textual Feedback: The operator in #0 is incorrect, the second number in #0 is incorrect, the operator in #1 is incorrect, and the second number in #1 is incorrect.

Trained



Critic

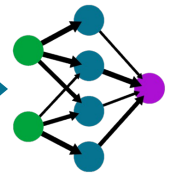
Algorithm 1 REFINER Training

for E epochs **do**
  for $i(batch) \leftarrow 1$ to N **do**
    Initialize (feedback) $f_0 \leftarrow No$
    for $t \leftarrow 1$ to T **do**
      $\hat{z}^k_{i,t} \sim \pi_\theta(y_i|c_i, f_{t-1}, \hat{z}_{i,t-1})$
      $f_t, \hat{z} \leftarrow \pi_\beta(c_i, z_i, \hat{z}^k_{i,t})$
      $\mathscr{L}^{lm}_i$ += $-\log p(z_i|c_i, f_{t-1}, \hat{z}_{i,t-1})$
    end for
  end for
end for
return π_θ

REFINER – Example

Context: Frank had number0 pieces of candy. He lost number1 of them. If he put the remaining pieces into bags with number2 pieces in each bag,

Question: How many bags would he have?

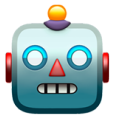


Generator

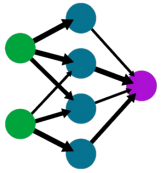
Reasoning Steps:

#0: *divide* (*number0*, *number2*) |
#1: *multiple* (#0, *number1*) |

Textual Feedback: The operator in #0 is incorrect, the second number in #0 is incorrect, the operator in #1 is incorrect, and the second number in #1 is incorrect.



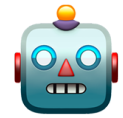
Critic



Generator

Reasoning Steps:

#0: *subtract* (*number0*, *number1*) |
#1: *divide* (#0, *number0*) |



Critic

Textual Feedback: The second number in #1 is incorrect.

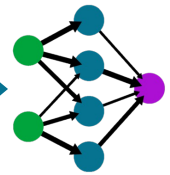
Algorithm 1 REFINER Training

1: **for** E epochs **do**
2: **for** $i(batch) \leftarrow 1$ to N **do**
3: Initialize (feedback) $f_0 \leftarrow No$
4: **for** $t \leftarrow 1$ to T **do**
5: $\hat{z}^k_{i,t} \sim \pi_\theta(y_i|c_i, f_{t-1}, \hat{z}_{i,t-1})$
6: $f_t, \hat{z} \leftarrow \pi_\beta(c_i, z_i, \hat{z}^k_{i,t})$
7: $\mathscr{L}^{lm}_i$ += $-\log p(z_i|c_i, f_{t-1}, \hat{z}_{i,t-1})$
8: **end for**
9: **end for**
10: **end for**
11: **return** π_θ

REFINER – Example

Context: Frank had number0 pieces of candy. He lost number1 of them. If he put the remaining pieces into bags with number2 pieces in each bag,

Question: How many bags would he have?

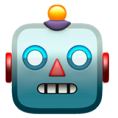


Generator

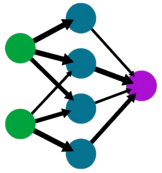
Reasoning Steps:

#0: *divide* (*number0*, *number2*) |
#1: *multiple* (#0, *number1*) |

Textual Feedback: The operator in #0 is incorrect, the second number in #0 is incorrect, the operator in #1 is incorrect, and the second number in #1 is incorrect.



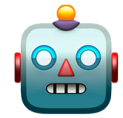
Critic



Generator

Reasoning Steps:

#0: *subtract* (*number0*, *number1*) |
#1: *divide* (#0, *number2*) |



Critic

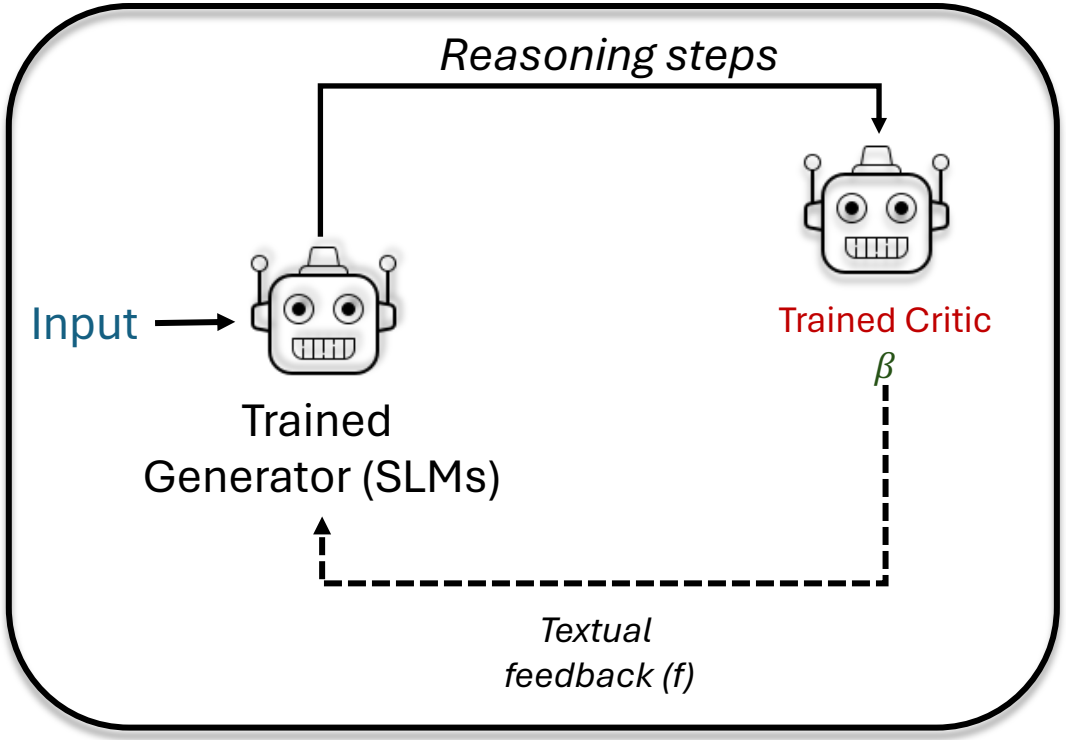
Textual Feedback: No hint

Algorithm 2 REFINER Inference

```
Initialize answers ← empty list
for i(batch) ← 1 to N do
    Initialize (reward) r_i ← 0, p_i ← 1
    Initialize (hint) h_0, ŷ_{i,0} ← No, []
    for (turn) t ← 1 to T do
        ŷ ← π_θ(y_i | c_i, h_{t−1}, ŷ_{i,t−1})
        h_t ← π_β(c_i, ŷ_i)
        if h_t == No then
            answers.append(ŷ)
            break
        end if
    end for
    answers.append(ŷ)
end for
return answers
```

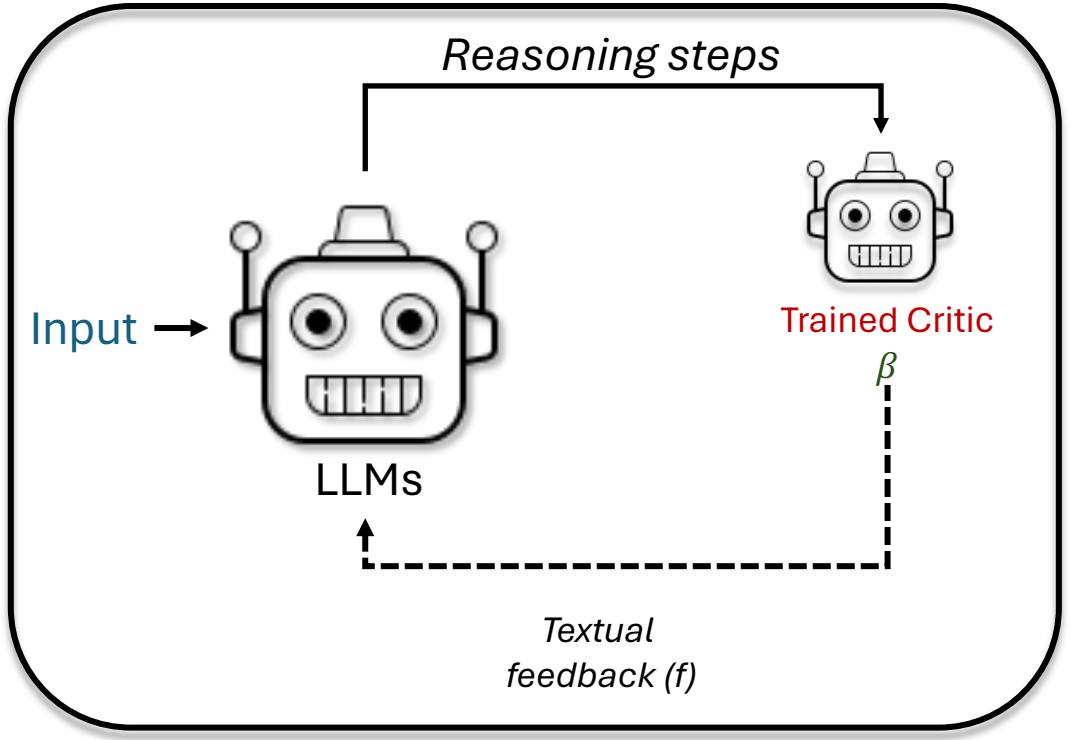
Inference

- We use the trained critic along with the trained generator to generate a trajectory $z_0, z_1, \ldots, z_T$ and stop when either z_T is generated by the generator or "*No hint*" is generated by the critic.

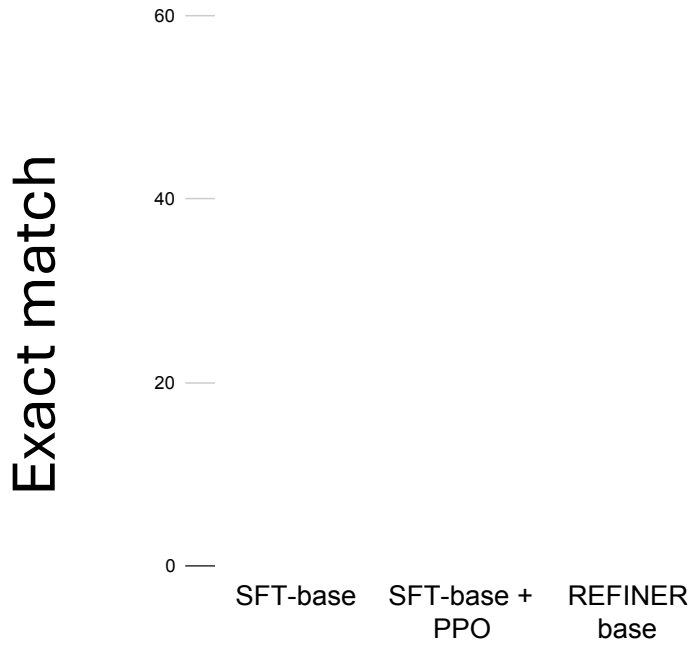


Inference – Trained Critic as a Tool

- We also experimented with **LLMs** (chain of thought prompting), where the generator generates a trajectory $z_0, z_1, \ldots, z_T$ and stops when the critic generates "*No hint*".



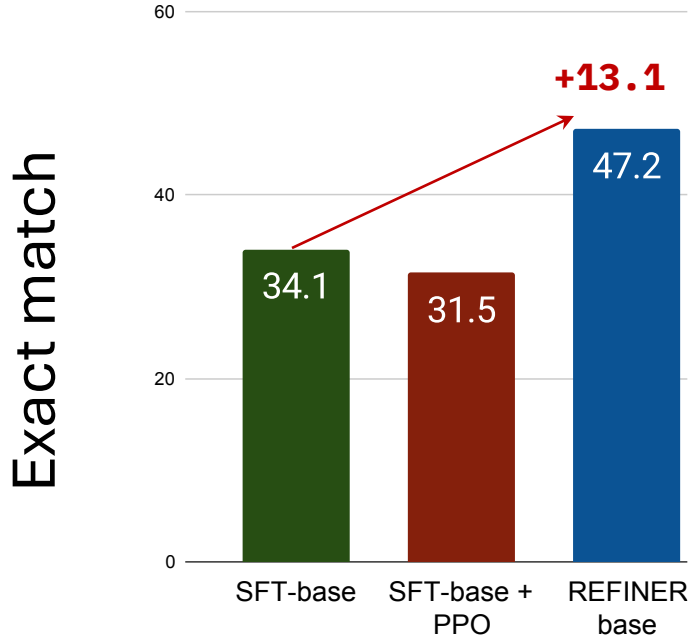
How well REFINER works?



Math Word Problem

Dataset: SVAMP
Base Model : T5

How well REFINER works on supervised models?



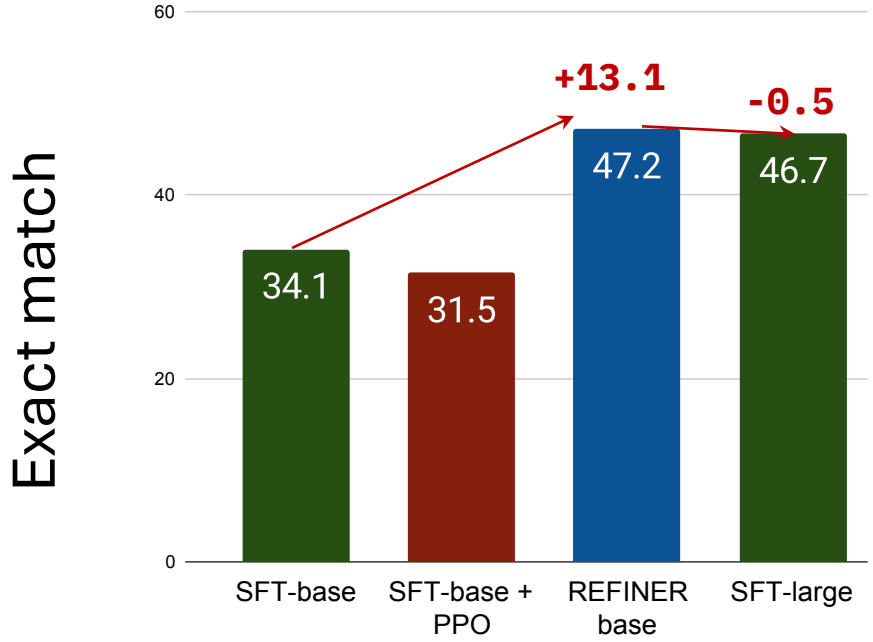
Math Word Problem

Dataset: SVAMP
Base Model : T5

Dataset: SVAMP

- We compare REFINER with SFT and PPO.
- Adding a critic in the loop improves +13.1 exact match scores over SFT.

How well REFINER works on supervised models?



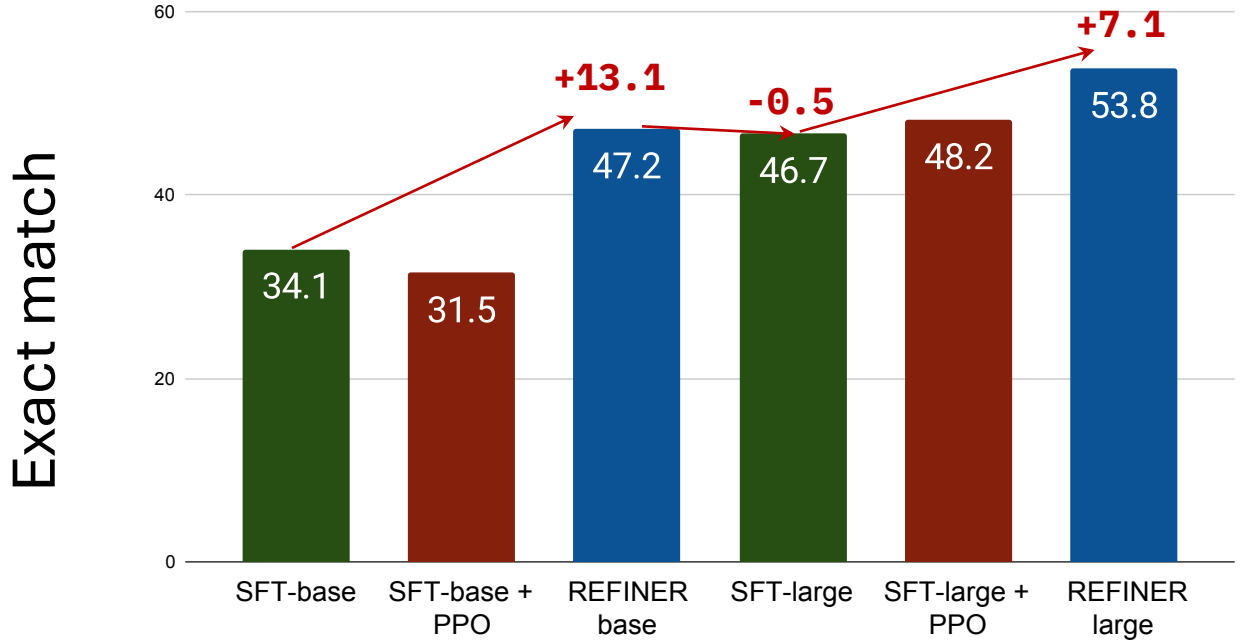
Math Word Problem

Dataset: SVAMP
Base Model : T5

Dataset: SVAMP

- We compare REFINER with SFT and PPO.
- Adding a critic in the loop improves +13.1 exact match scores over SFT.
- REFINER base model outperforms SFT-large model.

How well REFINER works on supervised models?



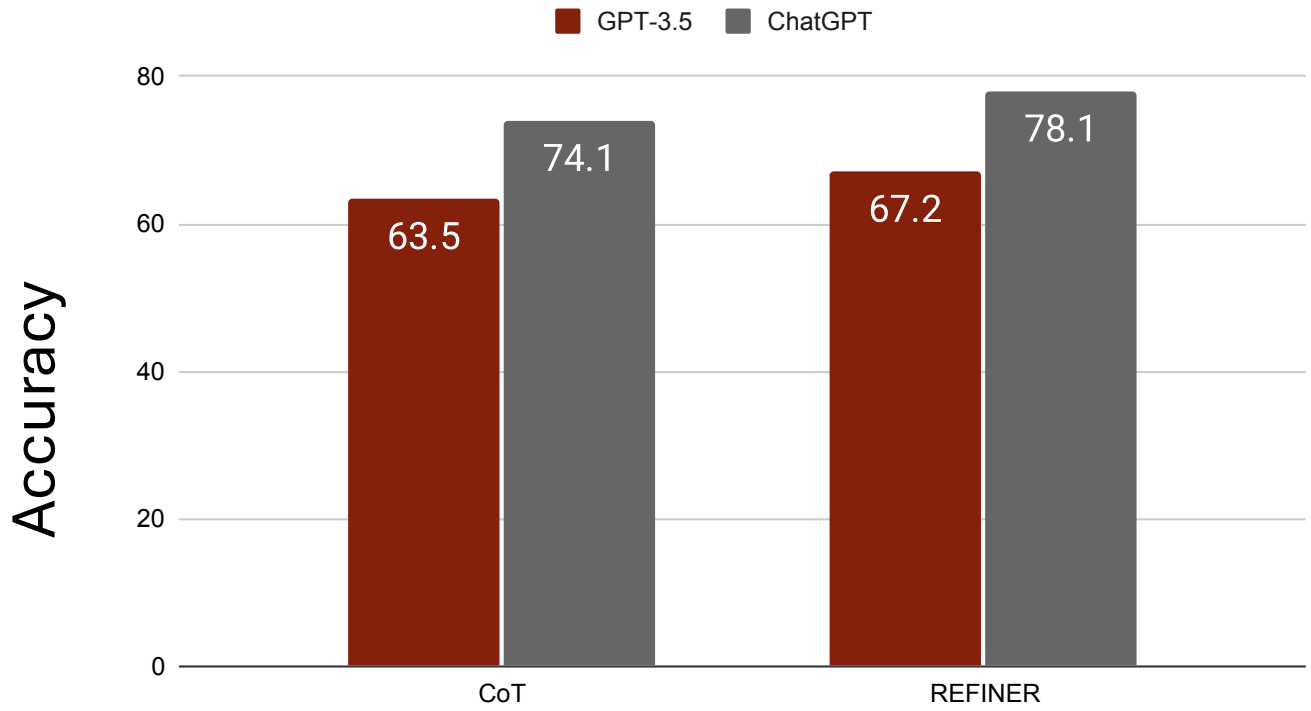
Structured and fine-grained natural language feedback surpasses scalar value feedback.

Math Word Problem

Dataset: SVAMP
Base Model : T5

- We compare REFINER with SFT and PPO.
- Adding a critic in the loop improves +13.1 exact match scores over SFT.
- REFINER base model outperforms SFT-large model.

Can we use critic model as a tool?



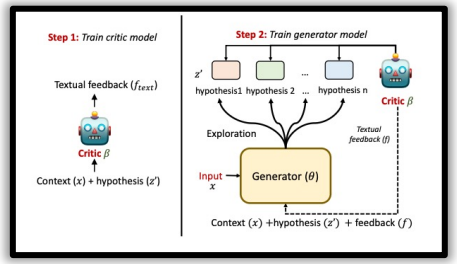
Trained critic can be used as a tool.

Math Word Problem

Dataset: GSM8K

- REFINER can improve CoT by +3.7 and +4.0.

Outline

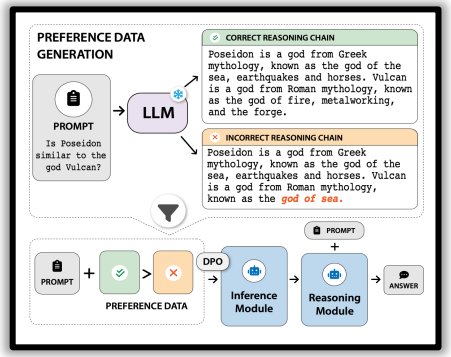


Refine knowledge using feedback

Paul et.al. 2024 (EACL)

Paul et.al. 2024 (EMNLP Findings)

Faithful Reasoning about knowledge



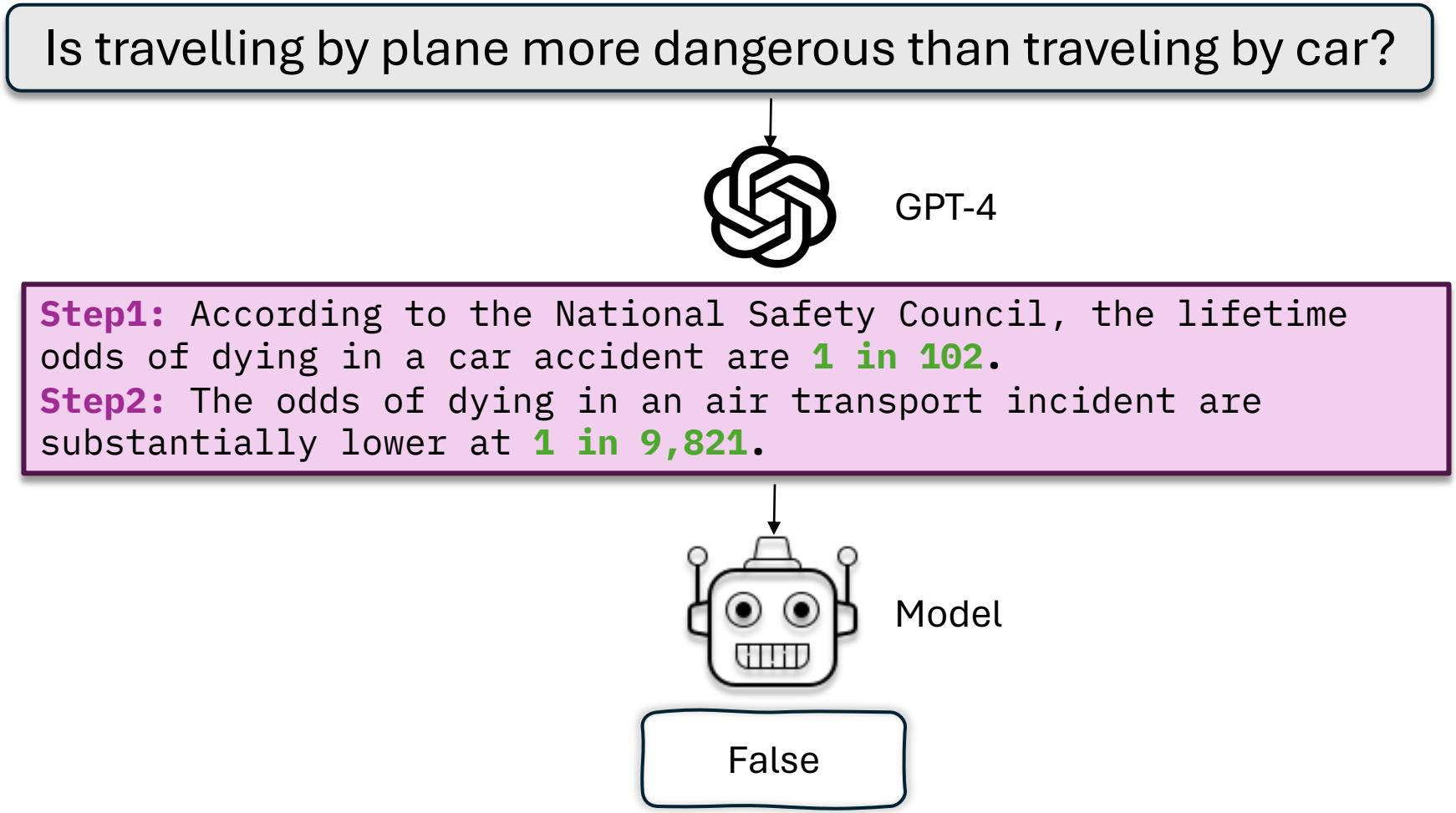
Making Reasoning Matter: Measuring and Improving the Faithfulness of Chain of Thoughts

EMNLP Findings 2024




Debjit Paul Robert West Antoine Bosselut Boi Faltings

Does LLMs reliabily use their own knowledge?

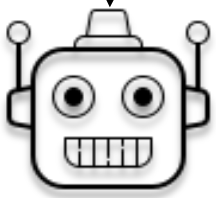


Does LLMs reliabily use their own knowledge?

Is travelling by plane more dangerous than traveling by car?

GPT-4

Step1: According to the National Safety Council, the lifetime odds of dying in a car accident are **1 in 9,821.**
Step2: The odds of dying in an air transport incident are substantially lower at **1 in 102.**

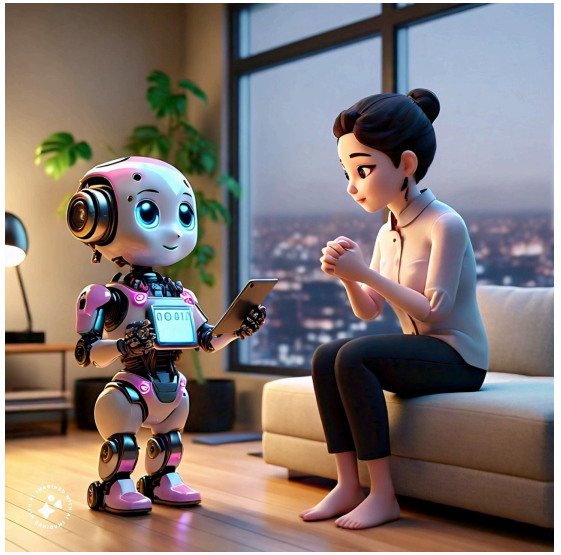


Model

True

We study how reliably LLMs use inference chains to arrive at a conclusion.

Research Question I

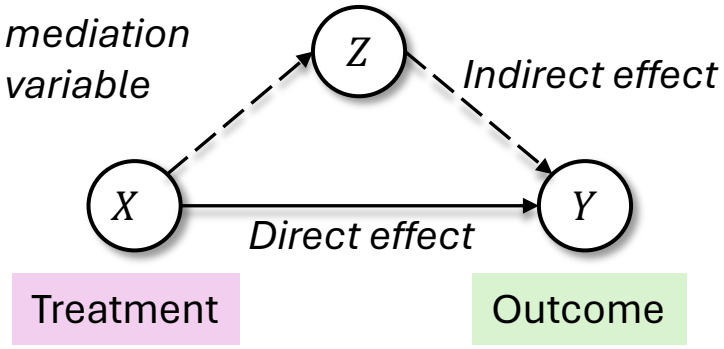


How can we effectively measure the faithfulness of LLMs when reasoning over reasoning chain?

Causal mediation analysis

- It is a method to measure how an independent variable (or treatment) affects a dependent variable (or outcome) mediated by intermediate variables

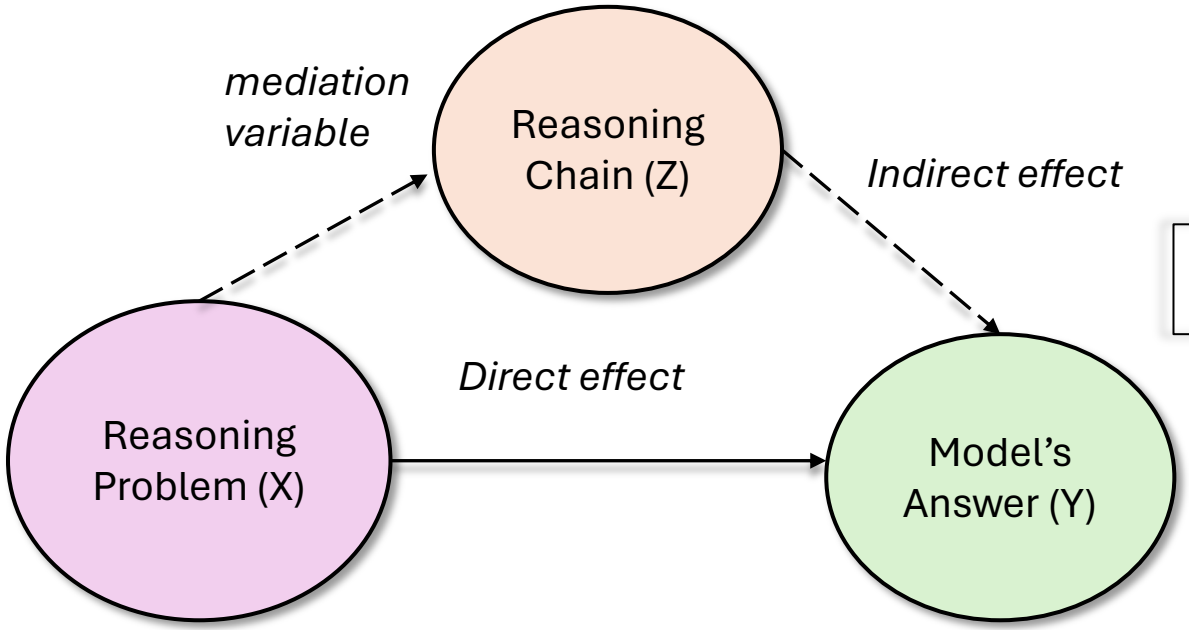
Pearl 2021



Judea Pearl

Causal mediation analysis

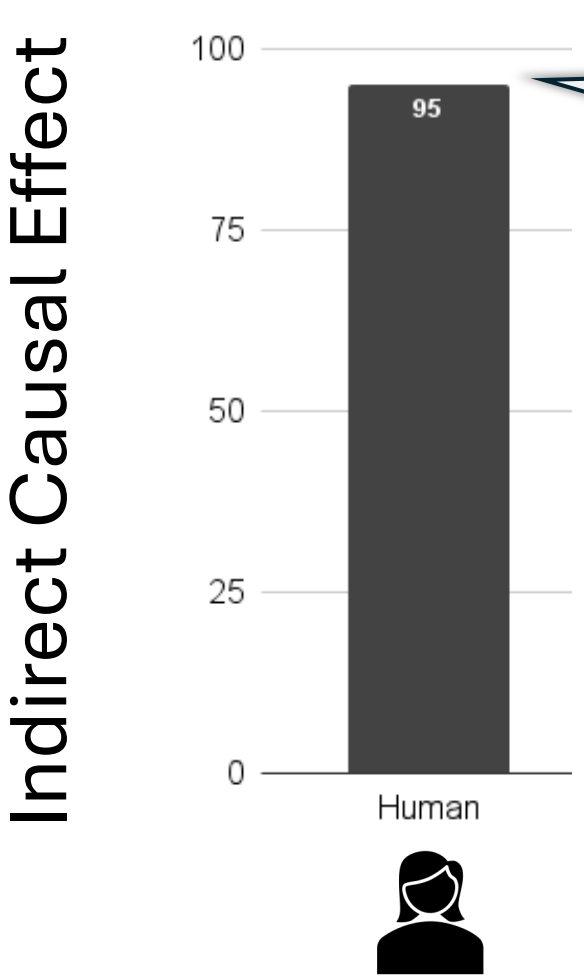
- It is a method to measure how an independent variable (or treatment) affects a dependent variable (or outcome) mediated by intermediate variables
- Measures the relationship between the reasoning chain (Z) and the model final answer (Y)



$$IE = E(Y \mid X, Z) - E(Y \mid X, Z')$$

Pearl, 2001

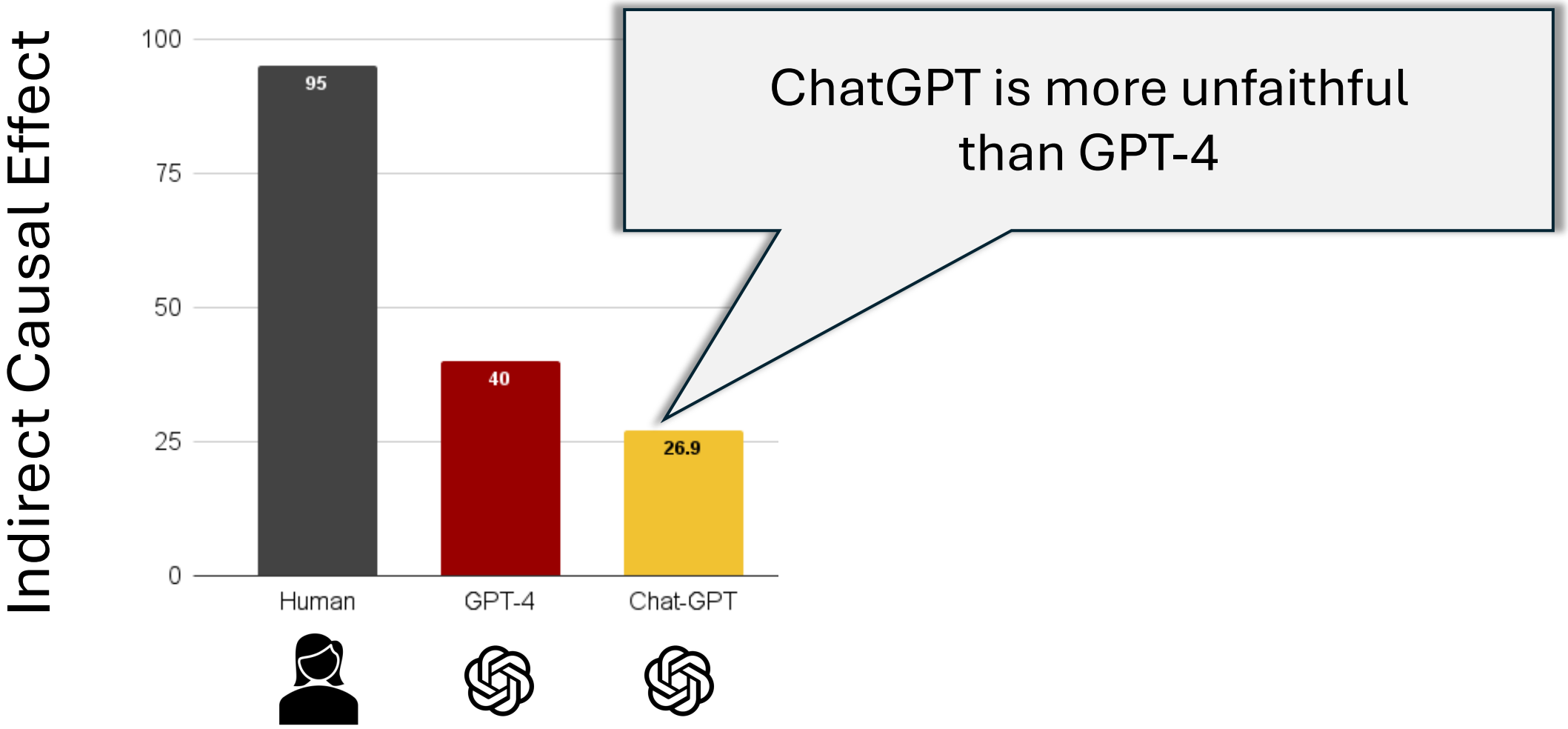
How well do humans perform?



A high indirect effect
means the model do emphasis on the reasoning steps

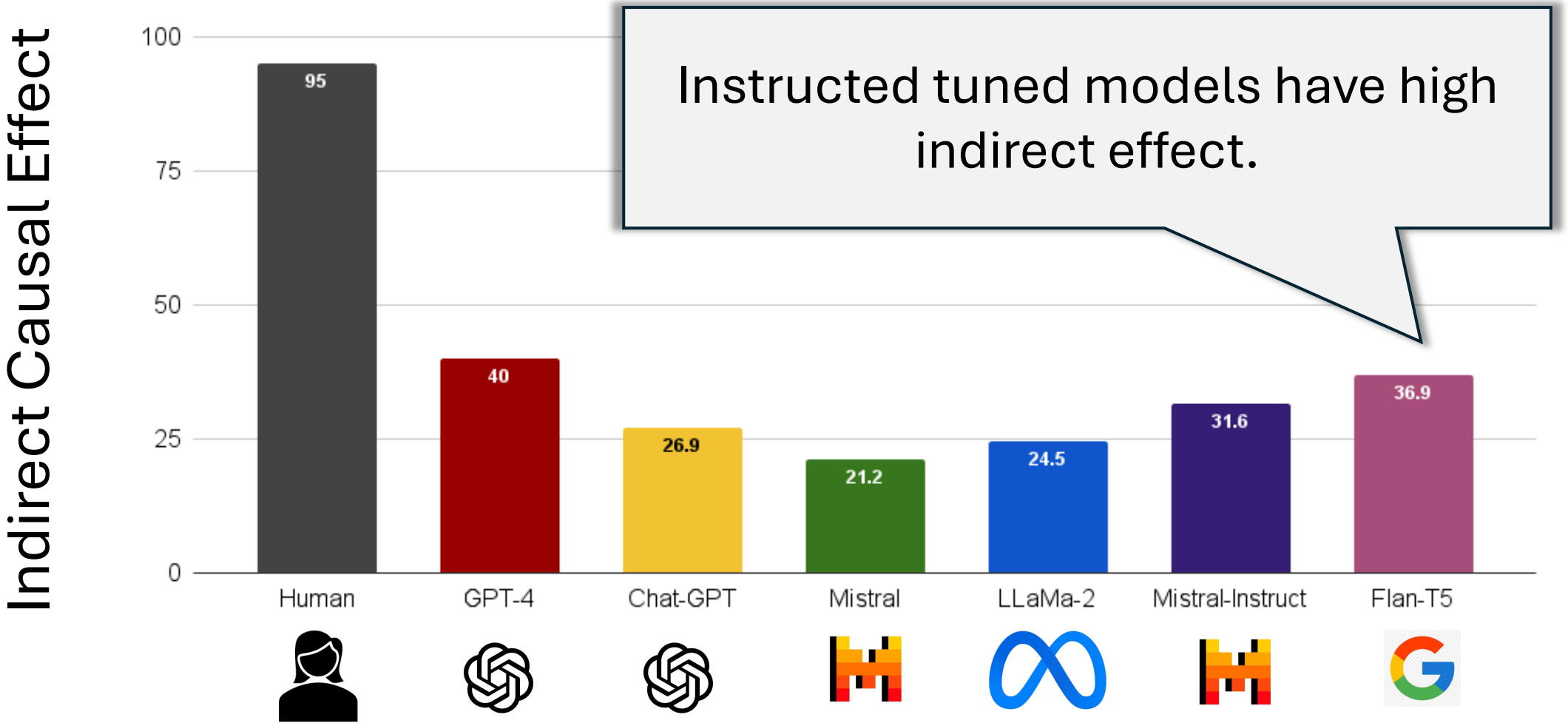
StrategyQA dataset

How well closed-models perform?



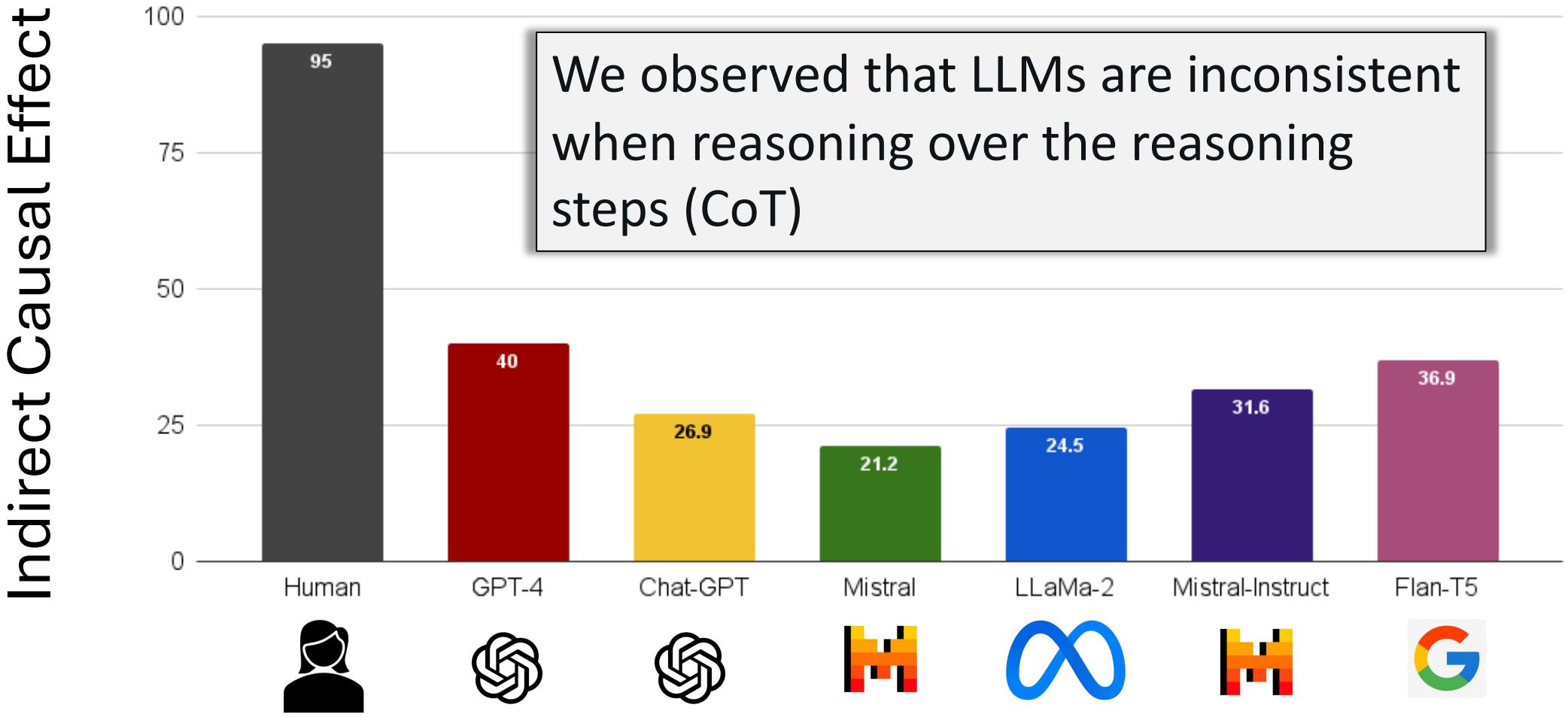
StrategyQA dataset

How well open-sourced-models perform?



StrategyQA dataset

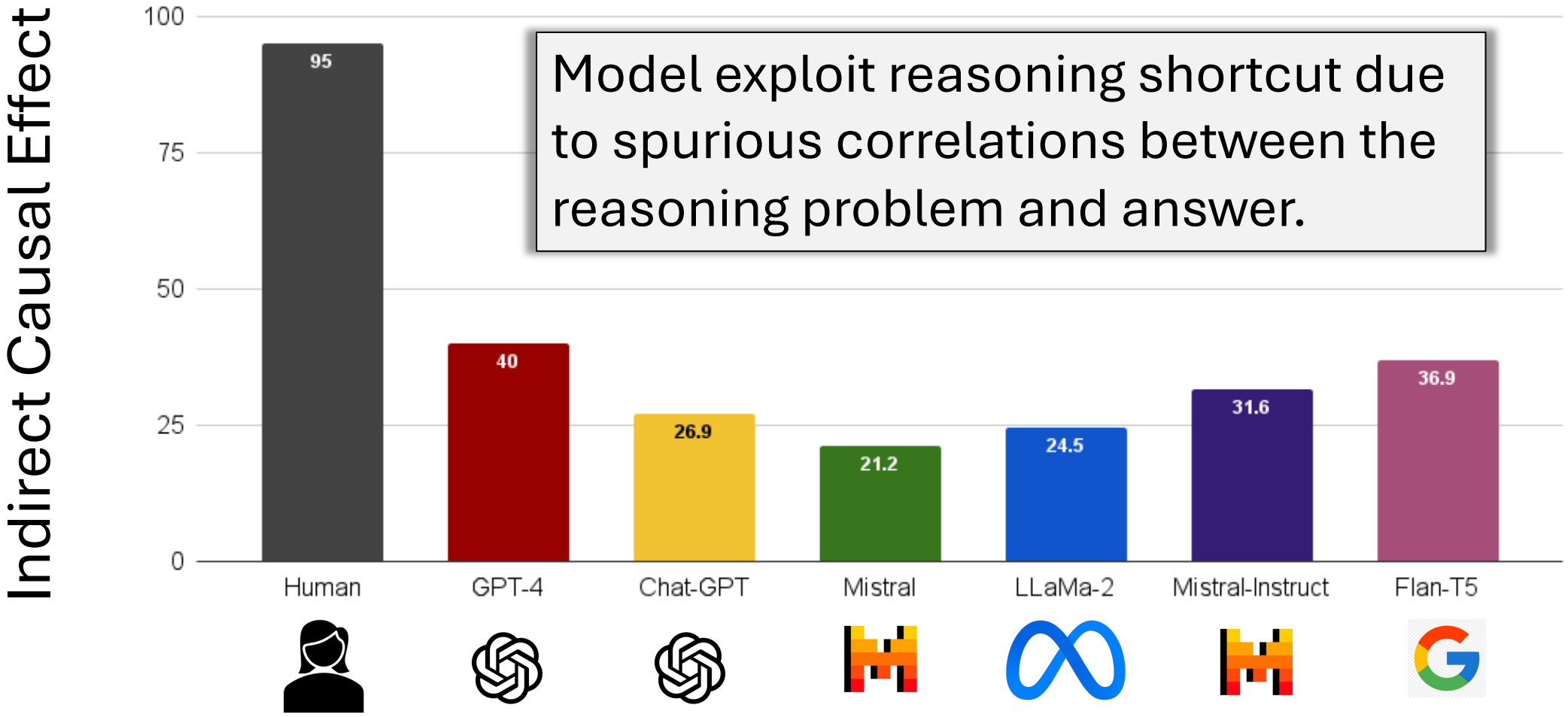
How well open-sourced-models perform?



We observed that LLMs are inconsistent when reasoning over the reasoning steps (CoT)

StrategyQA dataset

How well open-sourced-models perform?



Model exploit reasoning shortcut due to spurious correlations between the reasoning problem and answer.

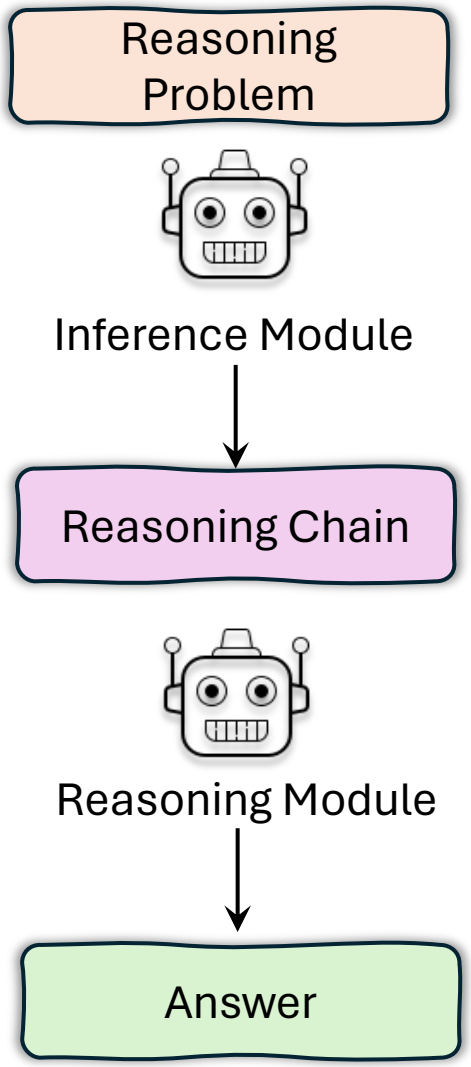
StrategyQA dataset

Research Question II

How do we build a model that can faithfuly reason over the inference chain and arrive to a correct answer?

FRODO

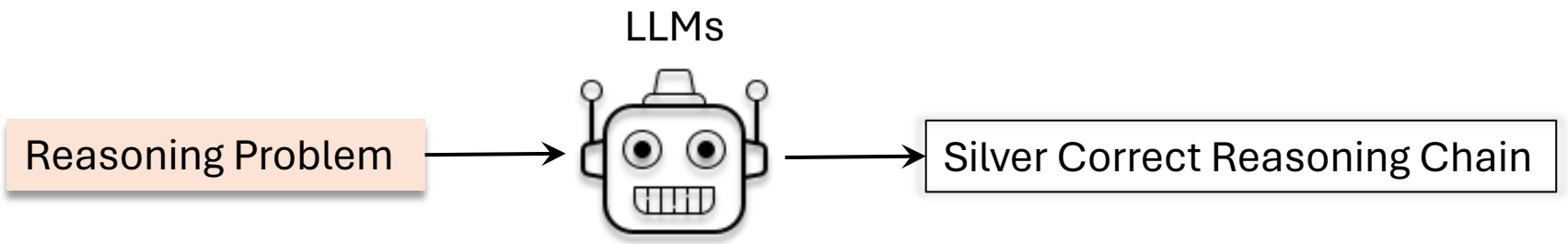
- FRODO comprises of two modules:
  - Inference module
    - generate correct reasoning chain
  - Reasoning module
    - takes the reasoning chains as input and faithfully reasons over them to arrive at the correct answer



Inference Module

- Obtaining human-crafted golden reasoning chain is both time-consuming and costly.
- Automatically obtain the silver rationale from LLM (ChatGPT) using in-context learning.

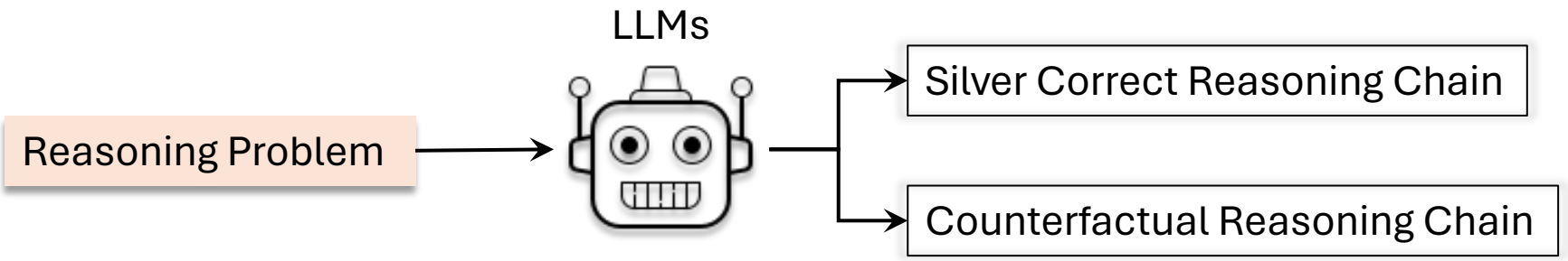
No gold reasoning chain



Inference Module

- Automatically obtain the silver rationale from LLM (ChatGPT) using in-context learning.

- **Hypothesis:** Learning to prefer correct reasoning chains over counterfactual chains can make models more robust and enhance generalization.



Example of Reasoning Chain

Reasoning Problem

Is Poseidon similar to the gold Vulcan?

ChatGPT

Silver Correct Reasoning Chain

Poseidon is a god from Greek mythology, known as the god of the sea, storms, and earthquakes.

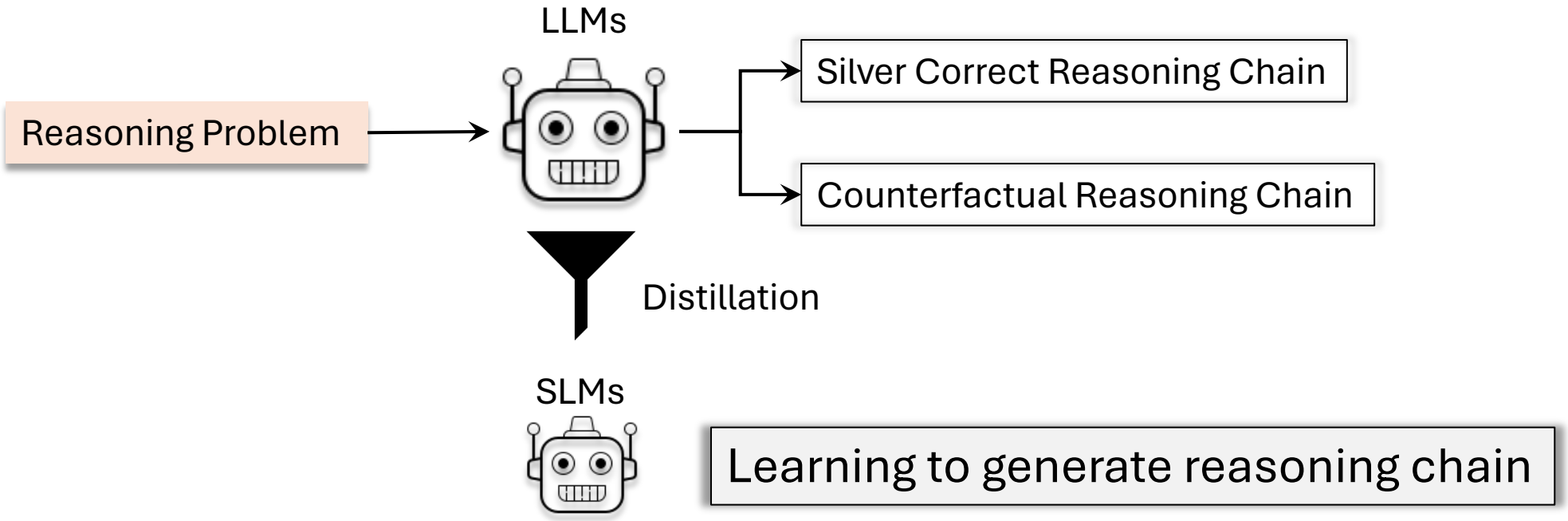
Vulcan is a god from Roman mythology, who is also known as the **god of the fire**.

Poseidon is a god from Greek mythology, known as the god of the sea, storms, and earthquakes.

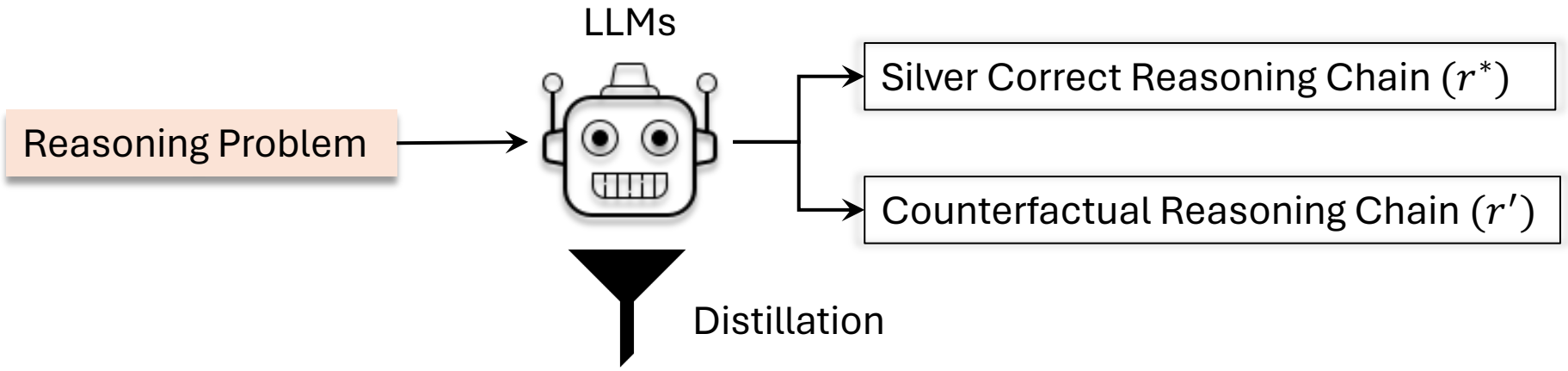
Vulcan is a god from Roman mythology, who is also known as the **god of the sea**.

Counterfactual Reasoning Chain

Inference Module



Inference Module



Training Phase

Imitation Phase

Training

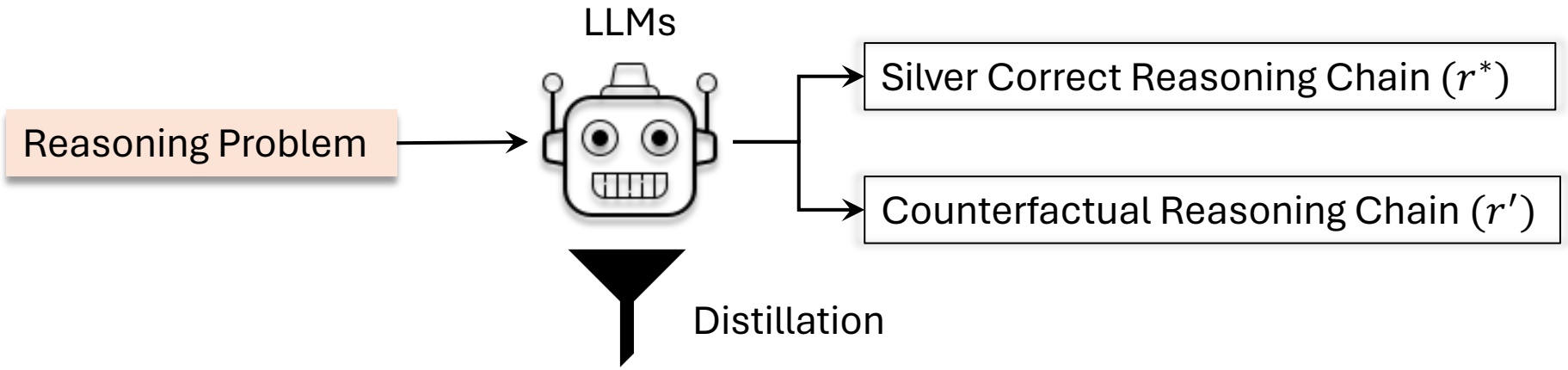
Simple Fine-Tuning

Learning

Generate correct reasoning chain

$$\text{Loss} = -\sum_{(x,r)} logP(r^*|x;\theta)$$

Inference Module



Training Phase

Imitation Phase

Training

Simple Fine-Tuning

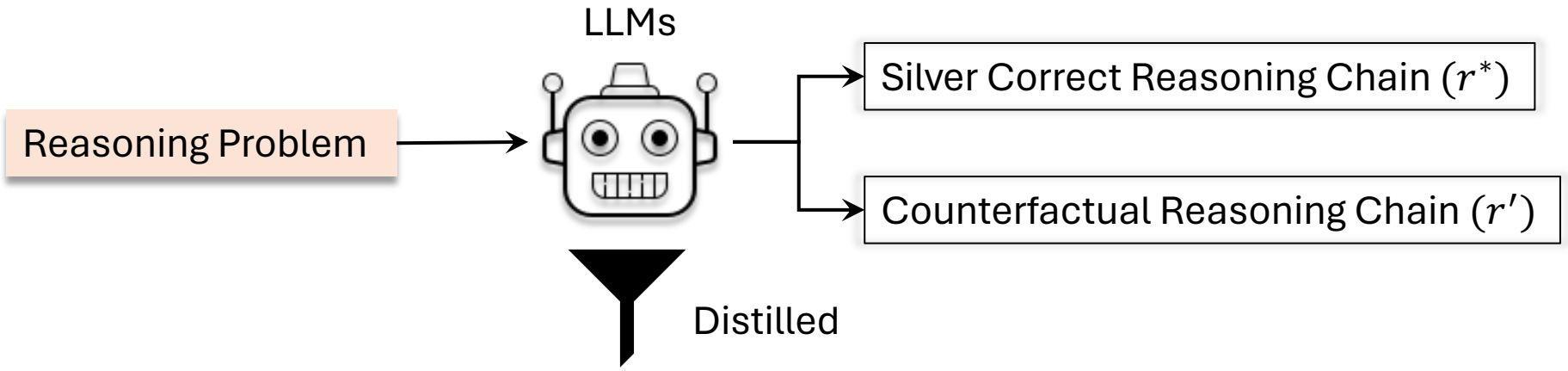
Learning

Generate correct reasoning chain

$$\text{Loss} = -\sum_{(x,r)} logP(r^*|x;\theta)$$

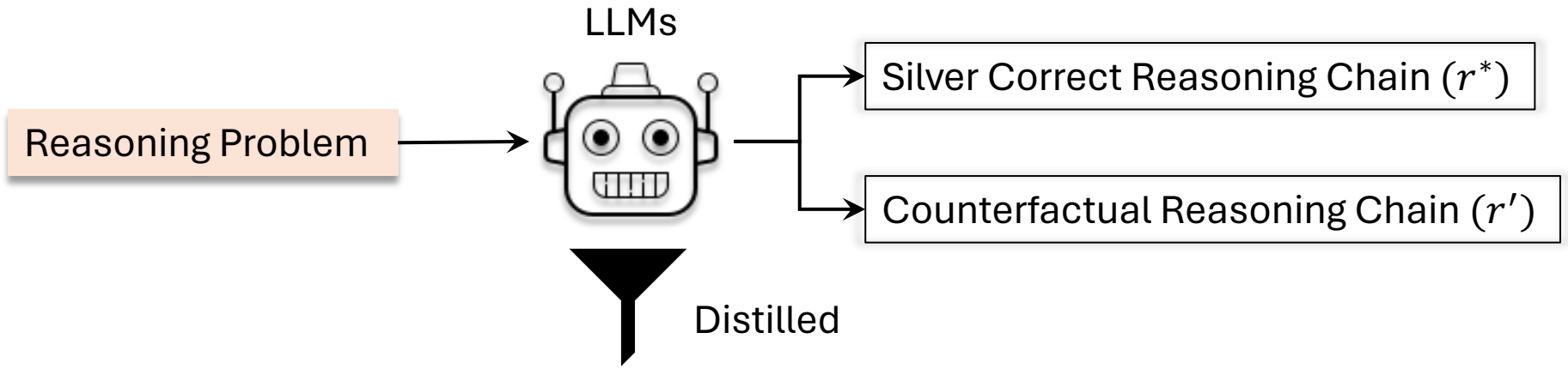
SFT can lead to learning shortcuts and degrade performance (Shridhar et.al. 2023).

Inference Module



Training Phase	SLMs	Training	Learning
Imitation Phase		Simple Fine-Tuning	Generate correct reasoning chain
RL Phase		Direct Preference Optimization (DPO)	Rank the correct reasoning chain over counterfactual chain

Inference Module



Training Phase

Imitation Phase

Training

Simple Fine-Tuning

Learning

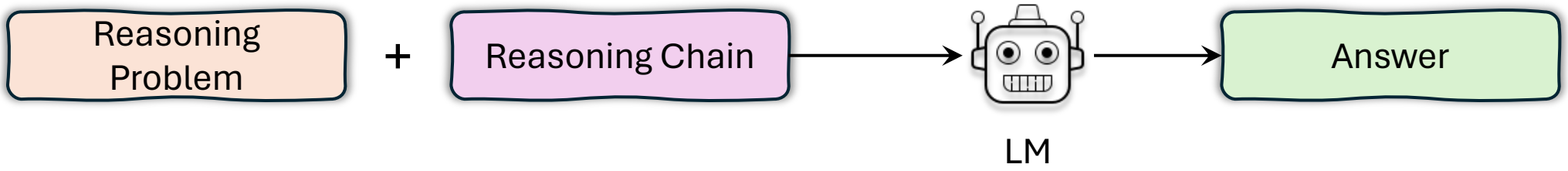
Generate correct reasoning chain

RL Phase

Direct Preference Optimization (DPO)

$$\text{Loss} = -\mathbb{E}_{\{x,r^*,r'\}} log\sigma(f_\theta(r^*,x) - f_\theta(r',x))$$

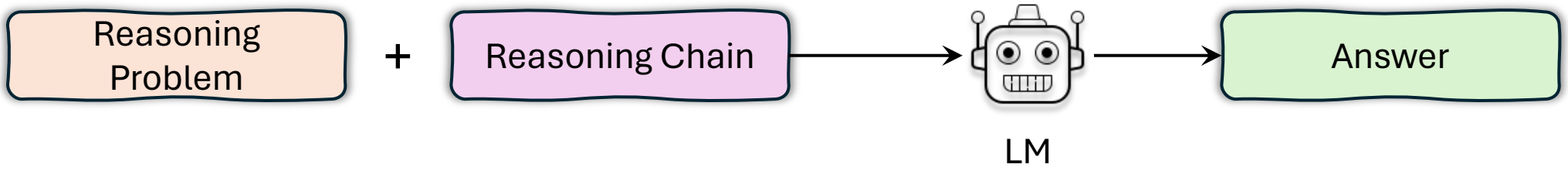
Reasoning Module



- **Goal:** Improve the synergy between the reasoning chain and the final answer.
- To train the reasoning module we used a linear combination of three losses:

$$\text{Loss} = -\alpha \sum_i logP(a^*|x, r^*) - \beta \sum_i logP(a'|x, r') + \gamma \max(0, \text{t} * \text{h}(\text{x}, r^*, a^*) - \text{h}(\text{x}, r', a') + m)$$

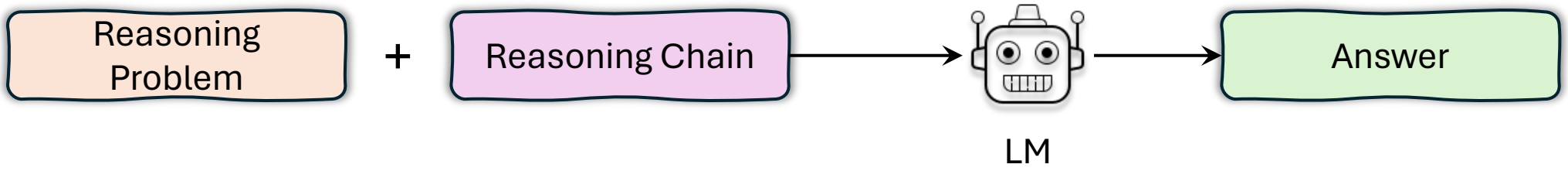
Reasoning Module - Correctness



- **Goal:** Improve the synergy between the reasoning chain and the final answer.
- To train the reasoning module we used a linear combination of three losses:

$$\text{Loss} = -\alpha \underbrace{\sum_i logP(a^*|x, r^*)}_{\text{language model loss}} - \beta \sum_i logP(a'|x, r') + \gamma \max(0, \text{t} * \text{h}(\text{x}, r^*, a^*) - \text{h}(\text{x}, r', a') + m)$$

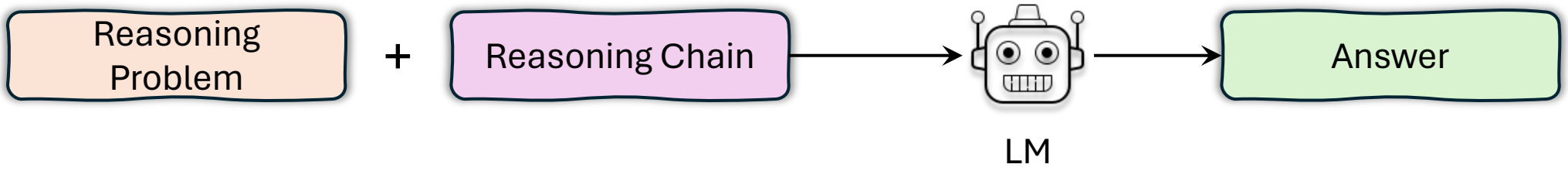
Reasoning Module - Faithfulness



- **Goal:** Improve the synergy between the reasoning chain and the final answer.
- To encourage our reasoner module to reason faithfully over the reasoning steps, we use counterfactual loss.

$$\text{Loss} = -\alpha \underbrace{\sum_i logP(a^*|x,r^*)}_{\text{language model loss}} - \beta \underbrace{\sum_i logP(a'|x,r')}_{\text{counterfactual loss}} + \gamma \max(0, \mathrm{t} * \mathrm{h}(\mathrm{x}, r^*, a^*) - \mathrm{h}(\mathrm{x}, r', a') + m)$$

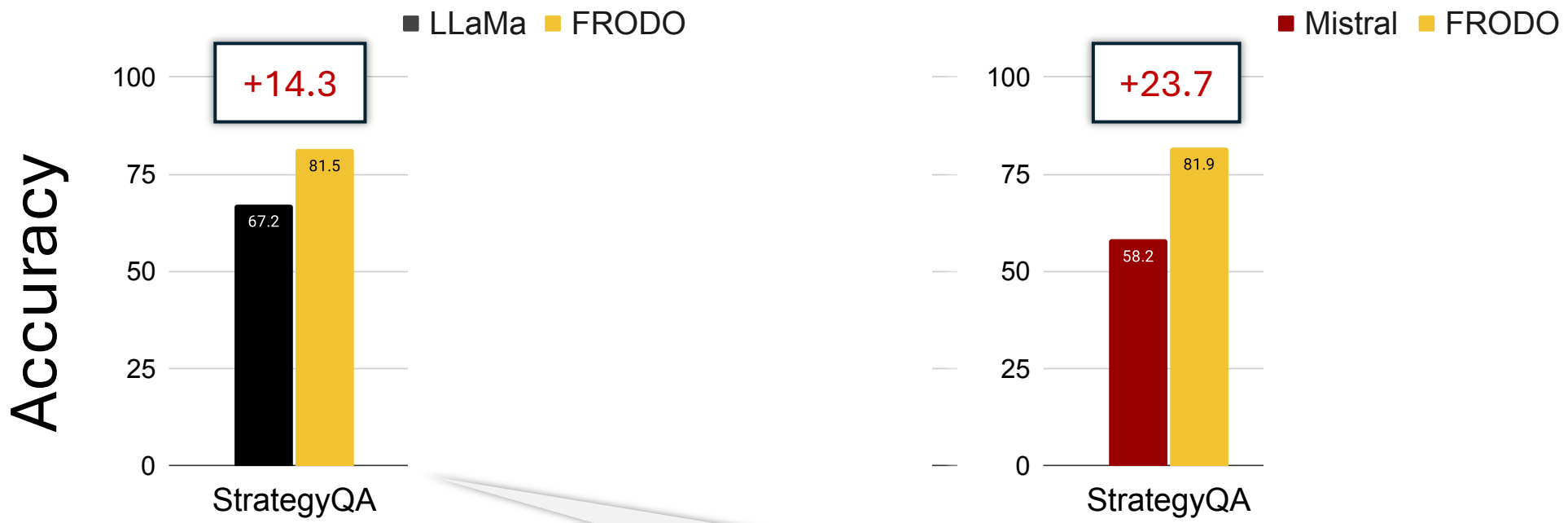
Reasoning Module – Robustness



- **Goal:** Improve the synergy between the reasoning chain and the final answer.
- To encourage our reasoner module to improve model robustness and generalization against input variation.

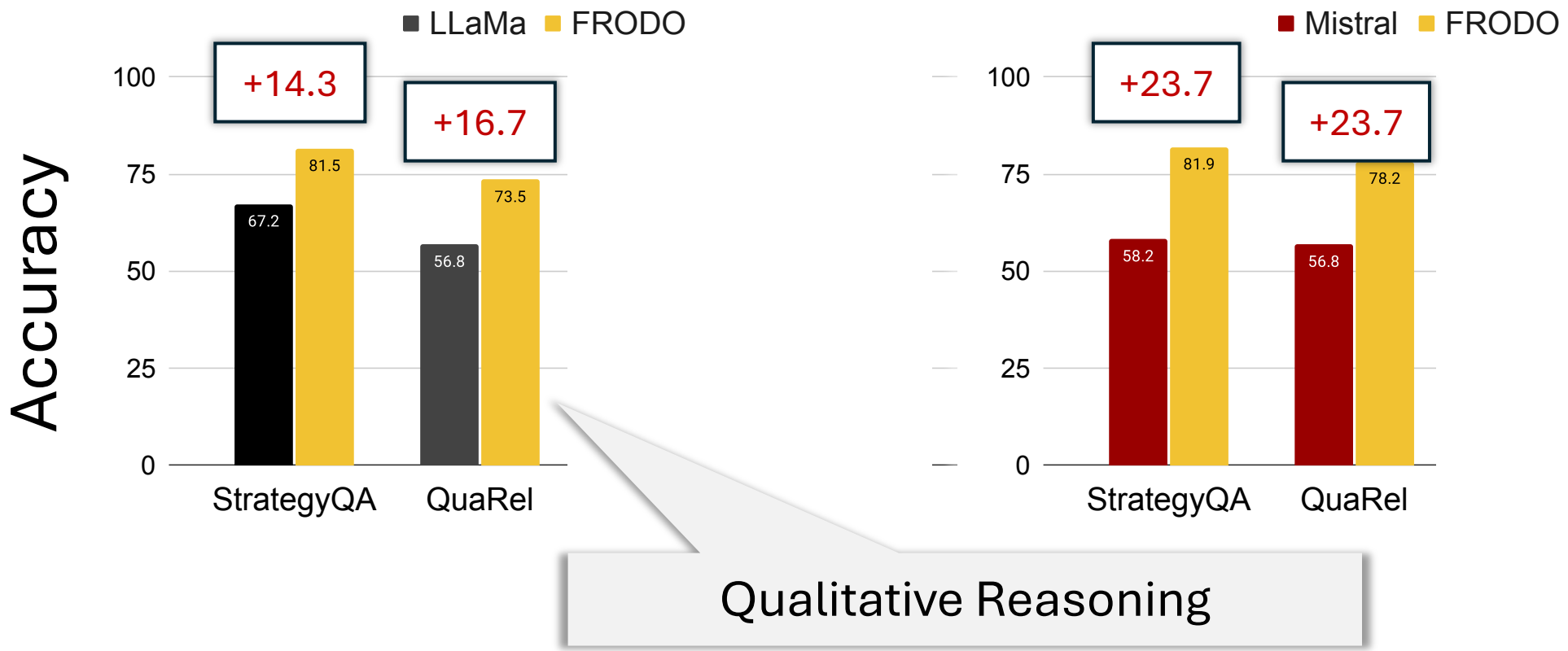
$$\text{Loss} = -\alpha \underbrace{\sum_i logP(a^*|x,r^*)}_{\text{language model loss}} - \beta \underbrace{\sum_i logP(a'|x,r')}_{\text{counterfactual loss}} + \gamma \underbrace{\max(0, \mathrm{t} * \mathrm{h}(\mathrm{x}, r^*, a^*) - \mathrm{h}(\mathrm{x}, r', a') + m)}_{\text{margin ranking loss}}$$

How well does FRODO work?



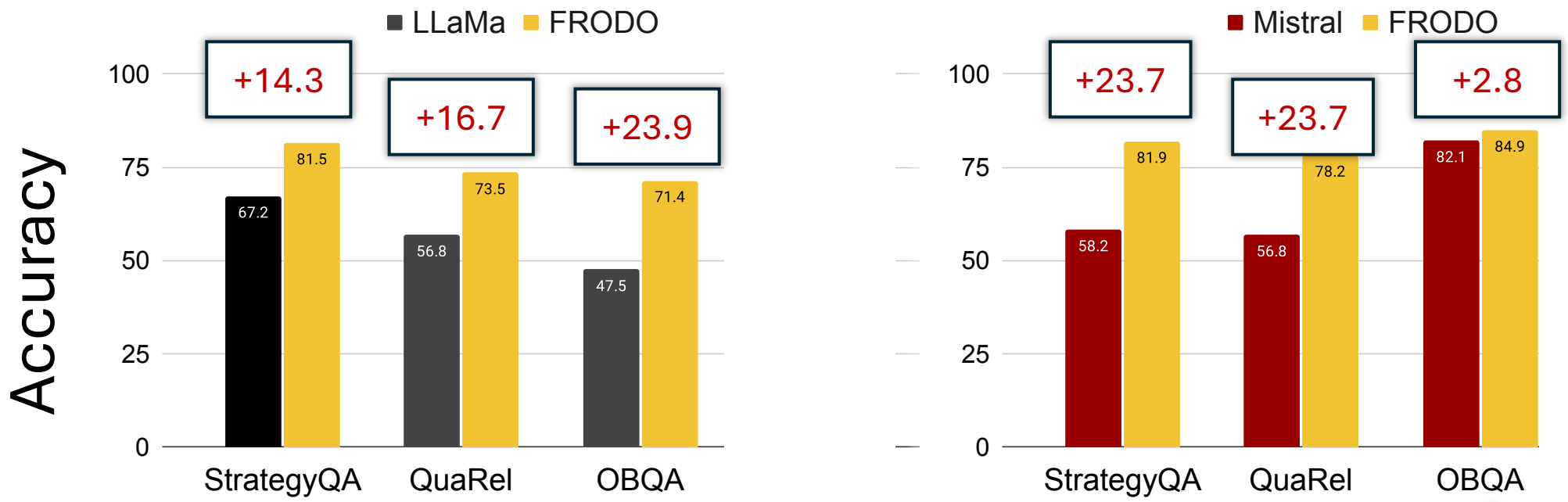
Commonsense and Factual reasoning

How well does FRODO work?



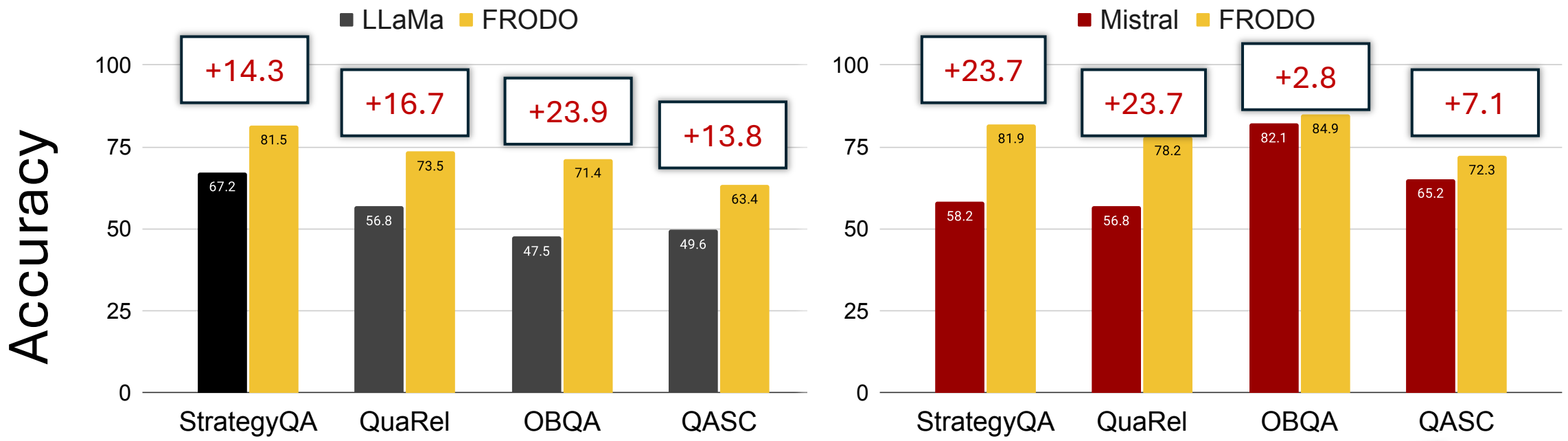
Qualitative Reasoning

How well does FRODO work?



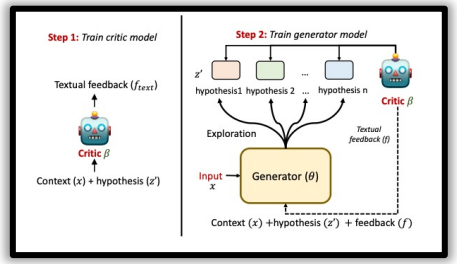
Commonsense and Factual reasoning

How well does FRODO work?



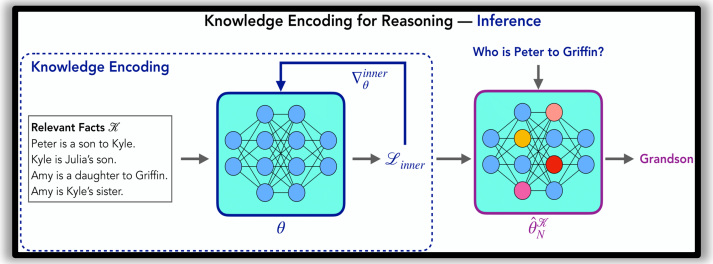
Reasoning over Science Facts

Outline



Refine knowledge using feedback

Paul et.al. 2024 (EACL)

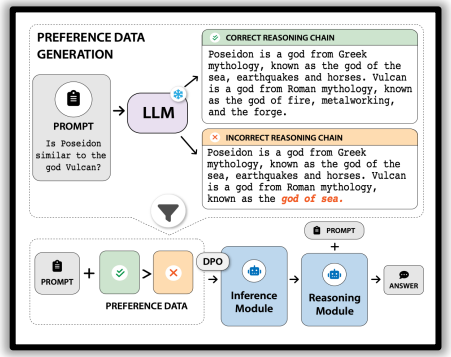


Robust Reasoning by updating world models

Chen et.al. 2023 (Neurips)

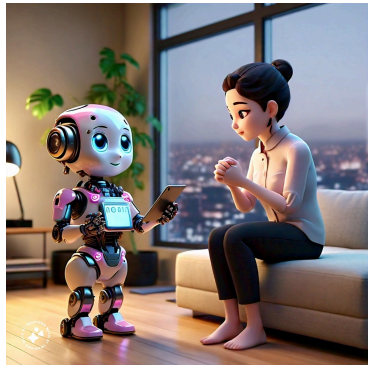
Paul et.al. 2024 (EMNLP Findings)

Faithful Reasoning about knowledge

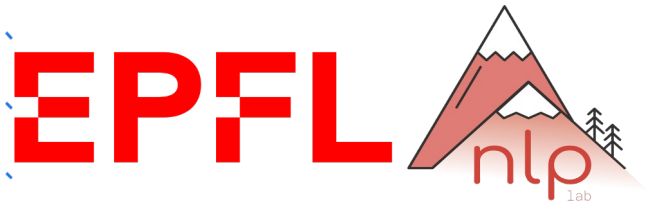


Recap

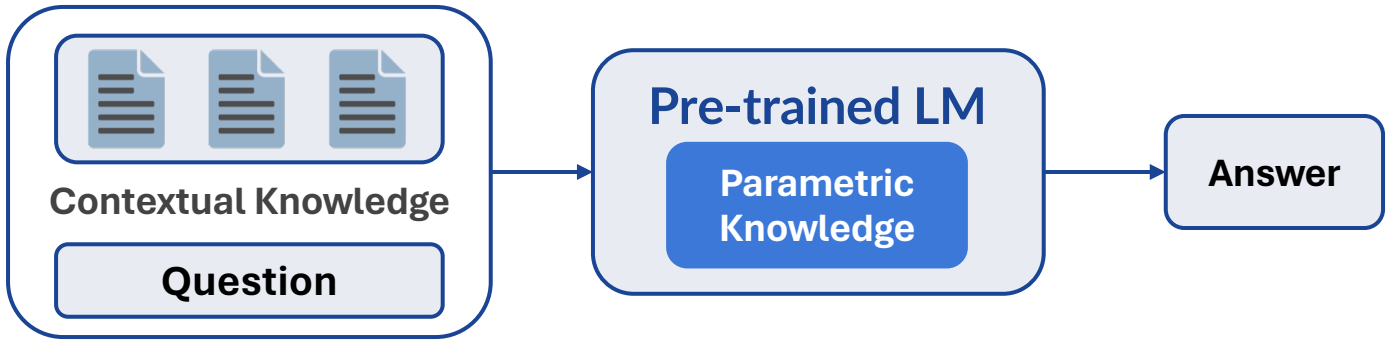
- Interactive method improves the quality of inference generation
- Causal mediation analysis show that LLMs such as GPT-4, ChatGPT etc. are inconsistent when reasoning over their own generated reasoning traces (CoT).
- Counterfactual reasoning can improve the faithfulness and performance of LMs.



RECKONING: Reasoning through Dynamic Knowledge Encoding



Background: PLM In-context Reasoning



(Input Facts:) Alan is blue. Alan is rough. Alan is young.
Bob is big. Bob is round.
Charlie is big. Charlie is blue. Charlie is green.
Dave is green. Dave is rough.

(Input Rules:) Big people are rough.
If someone is young and round, then they are kind.
If someone is round and big, then they are blue.
All rough people are green.

Q1: Bob is green. True/false? **[Answer: T]**
Q2: Bob is kind. True/false? **[Answer: F]**
Q3: Dave is blue. True/false? **[Answer: F]**

Challenge

Real-world context contains **noisy information**

PLMs are **sensitive** to **irrelevant** information in the context

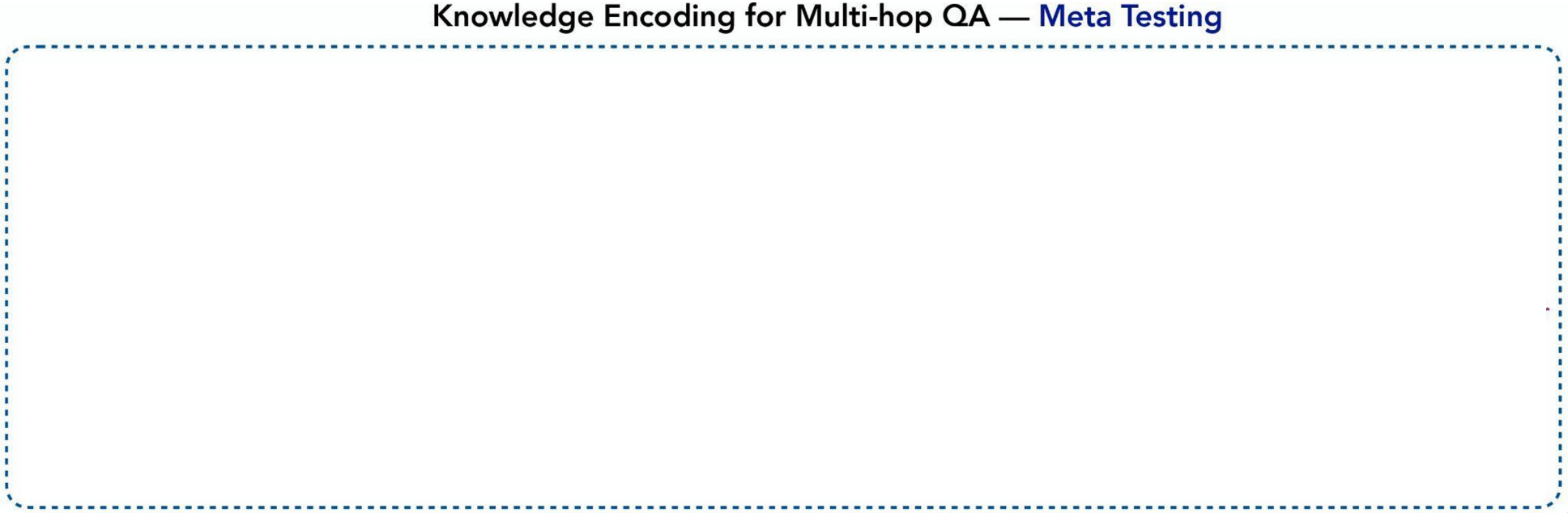
Challenge

Language models with different parameter sizes are sensitive to distractors in the context.

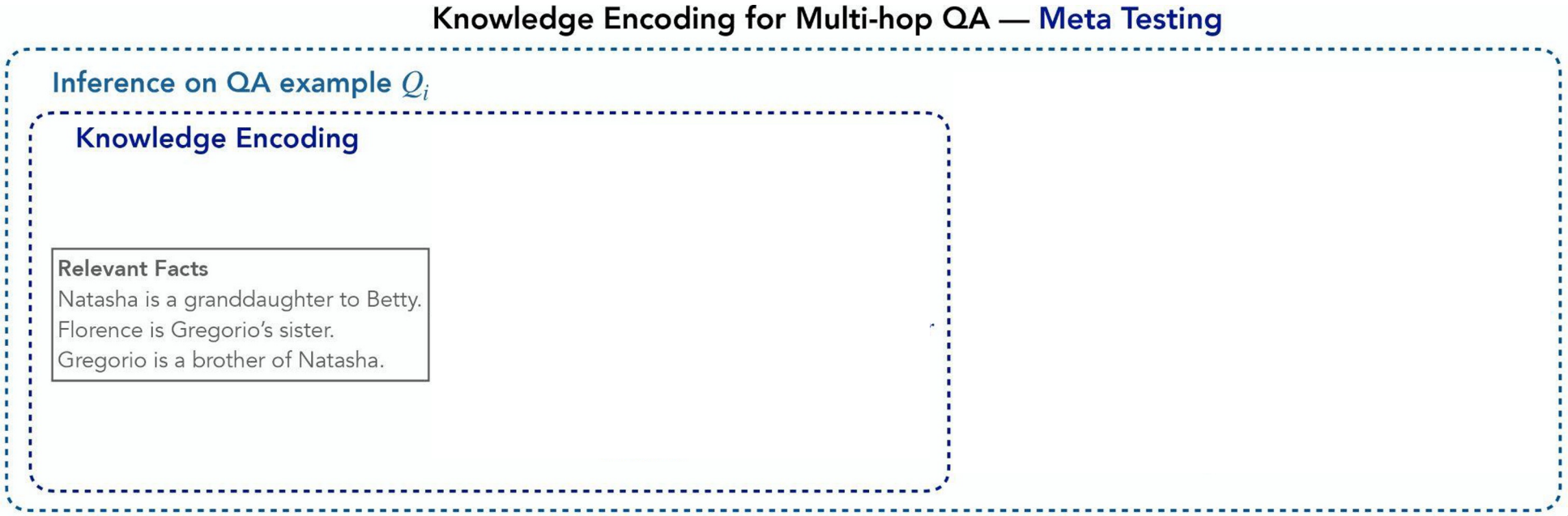
Research Question

Can we mitigate noisy information by turning contextual knowledge into parametric knowledge through backpropagation?

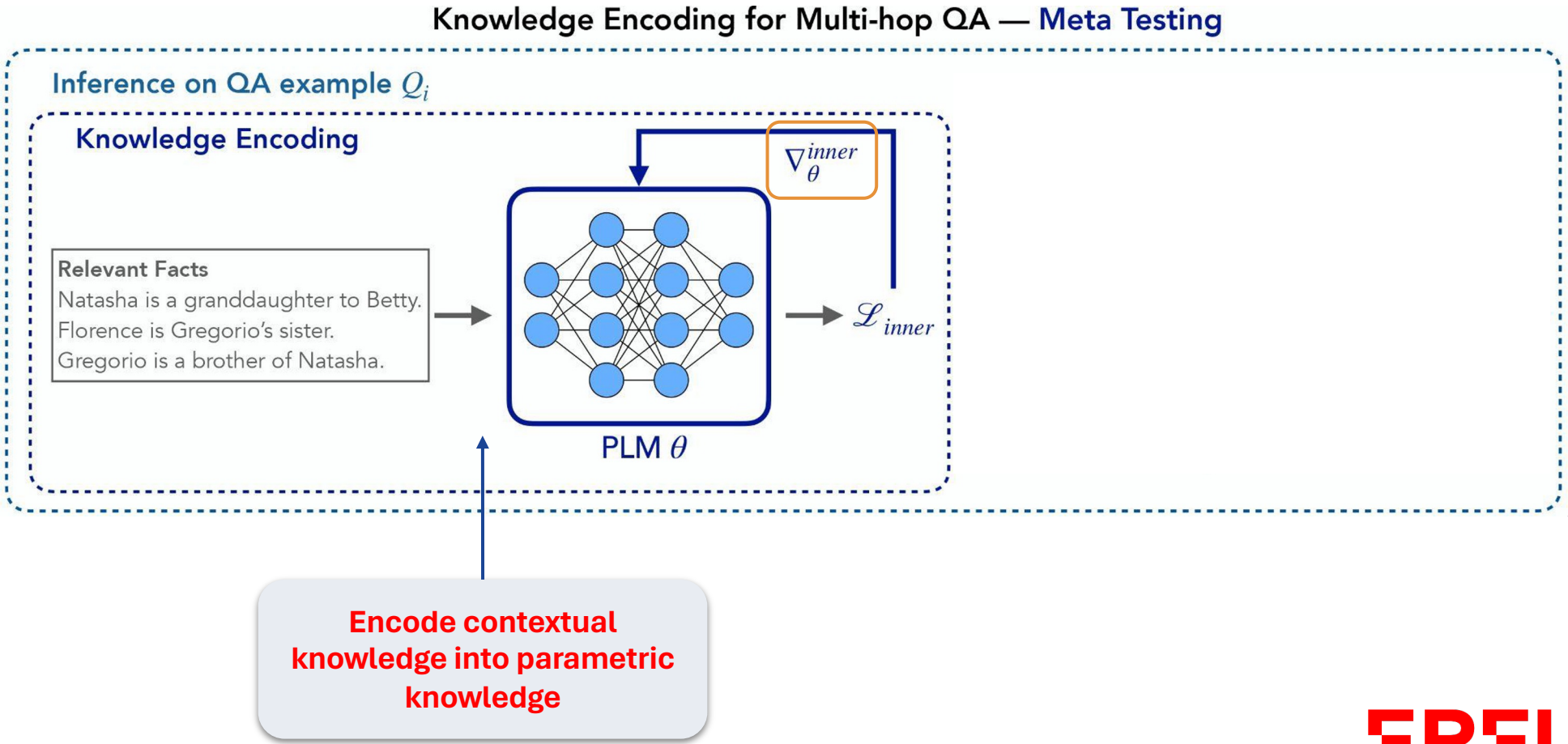
RECKONING: Knowledge Encoding for Reasoning



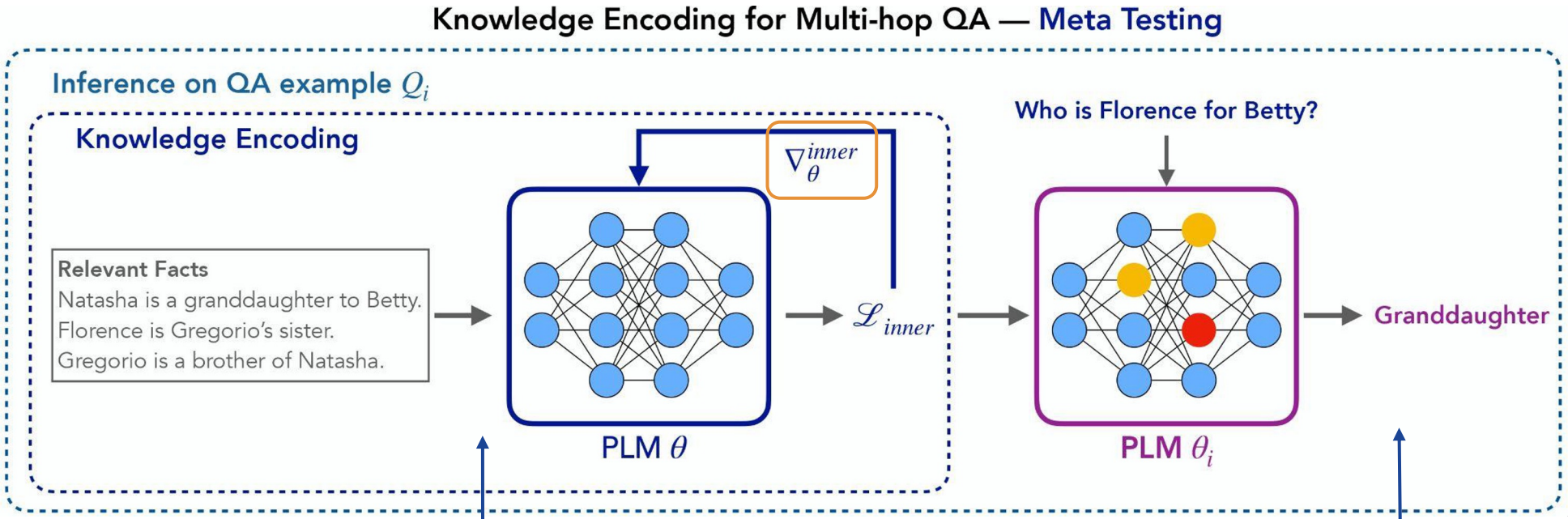
RECKONING: Knowledge Encoding for Reasoning



RECKONING: Knowledge Encoding for Reasoning



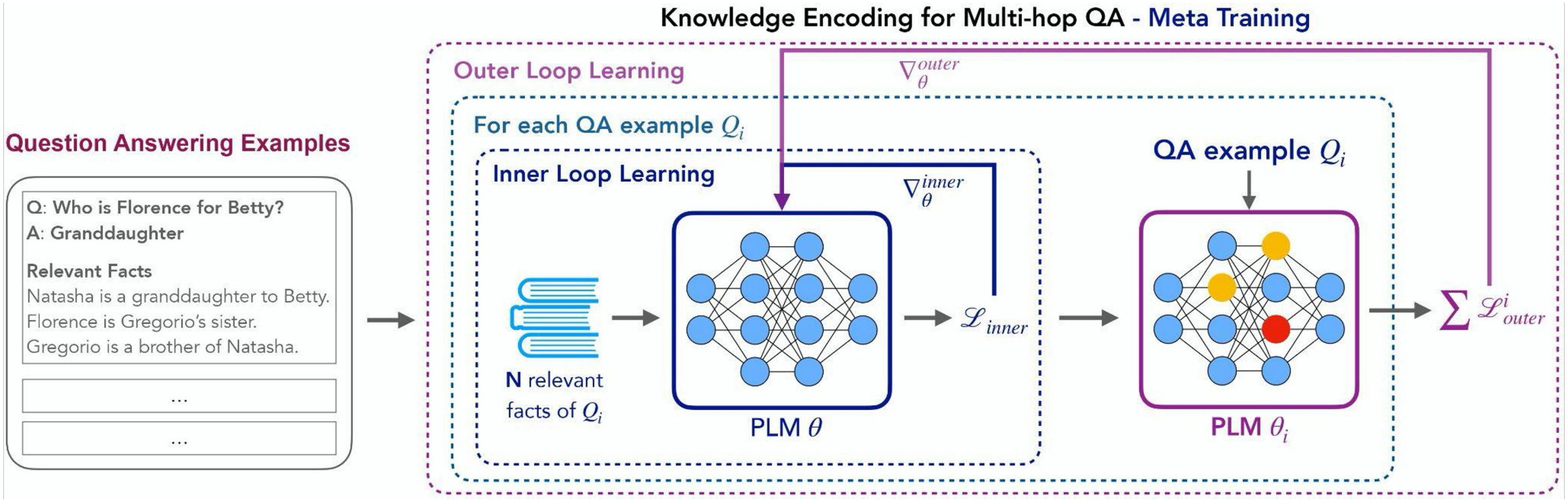
RECKONING: Knowledge Encoding for Reasoning



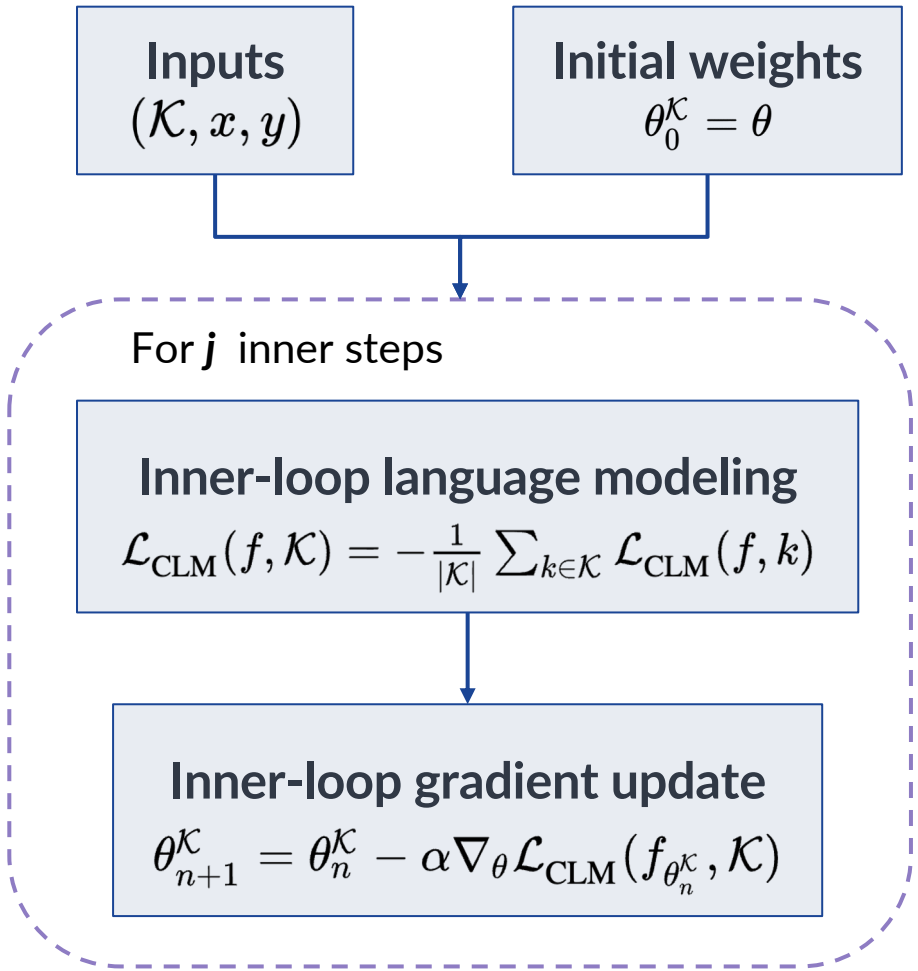
Encode contextual knowledge into parametric knowledge

Access and reason over parametric knowledge

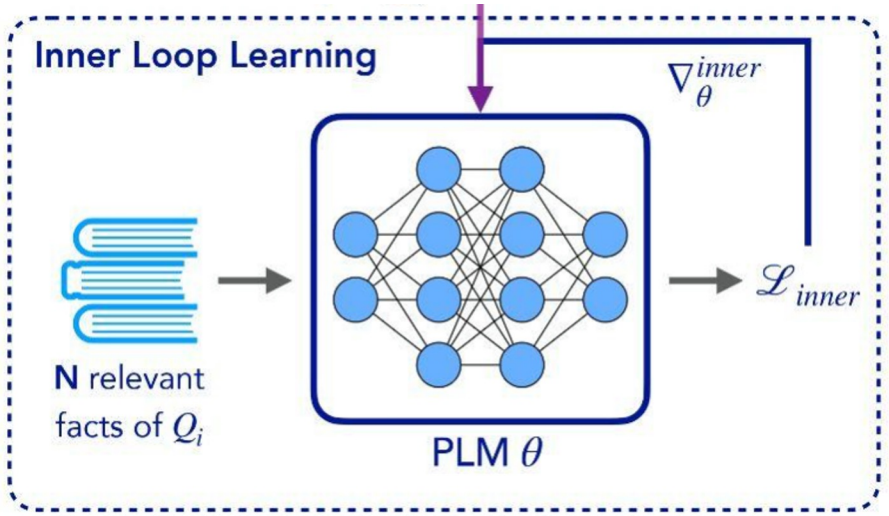
Bi-level Optimization



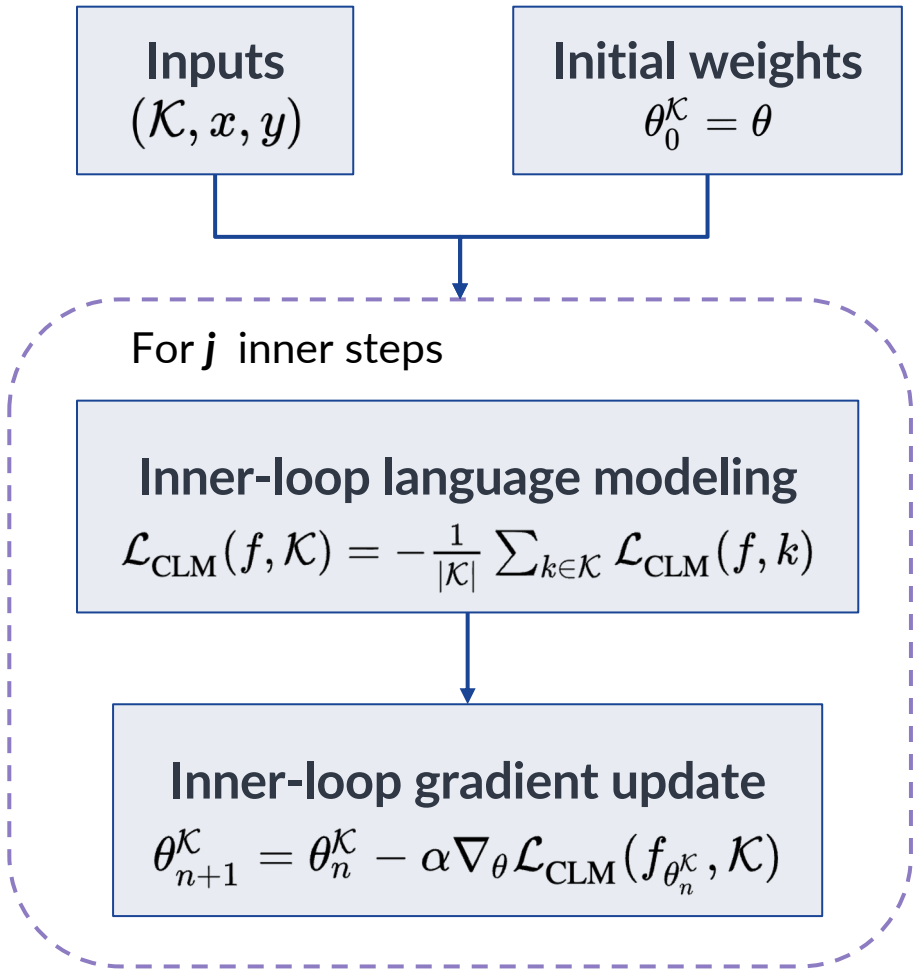
Inner-loop Learning



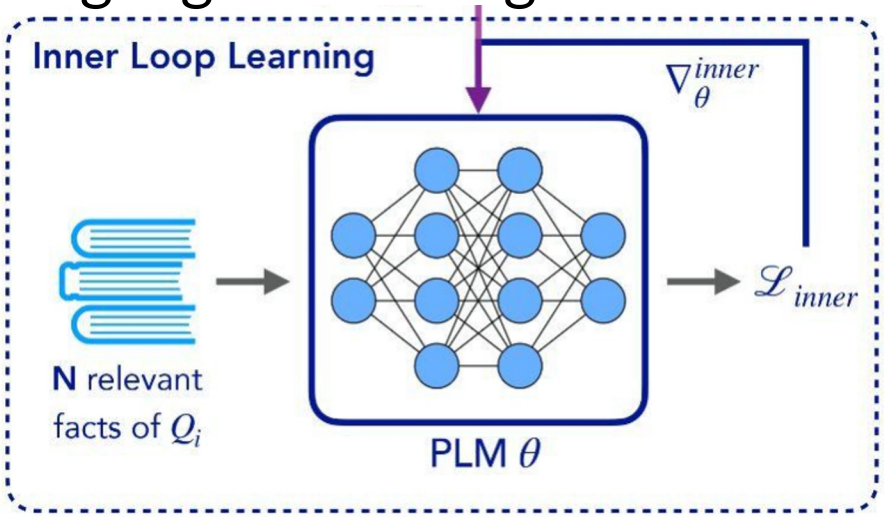
- **Language modeling** on natural language knowledge



Inner-loop Learning

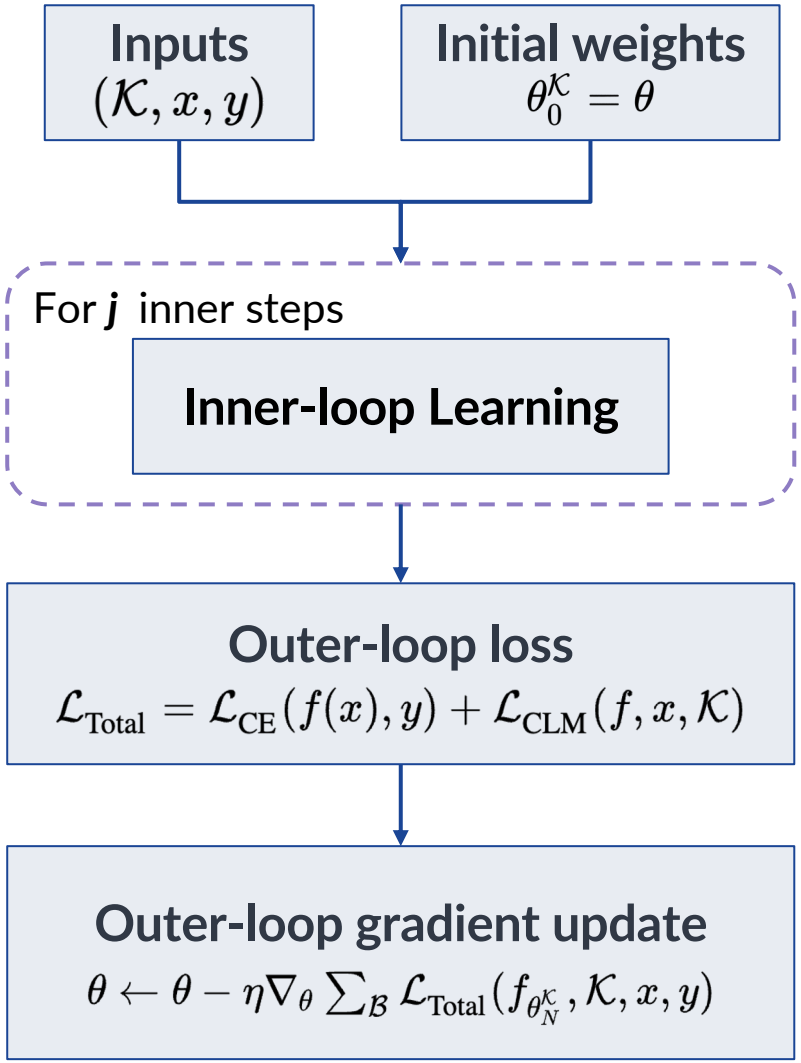


- **Language modeling** on natural language knowledge

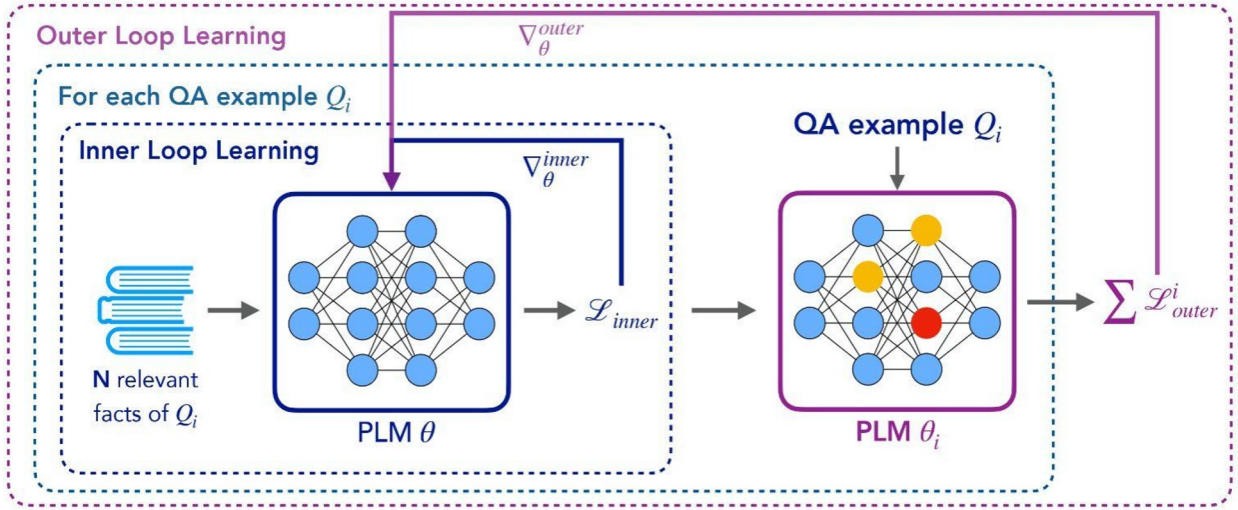


Knowledge encoding through backpropagation

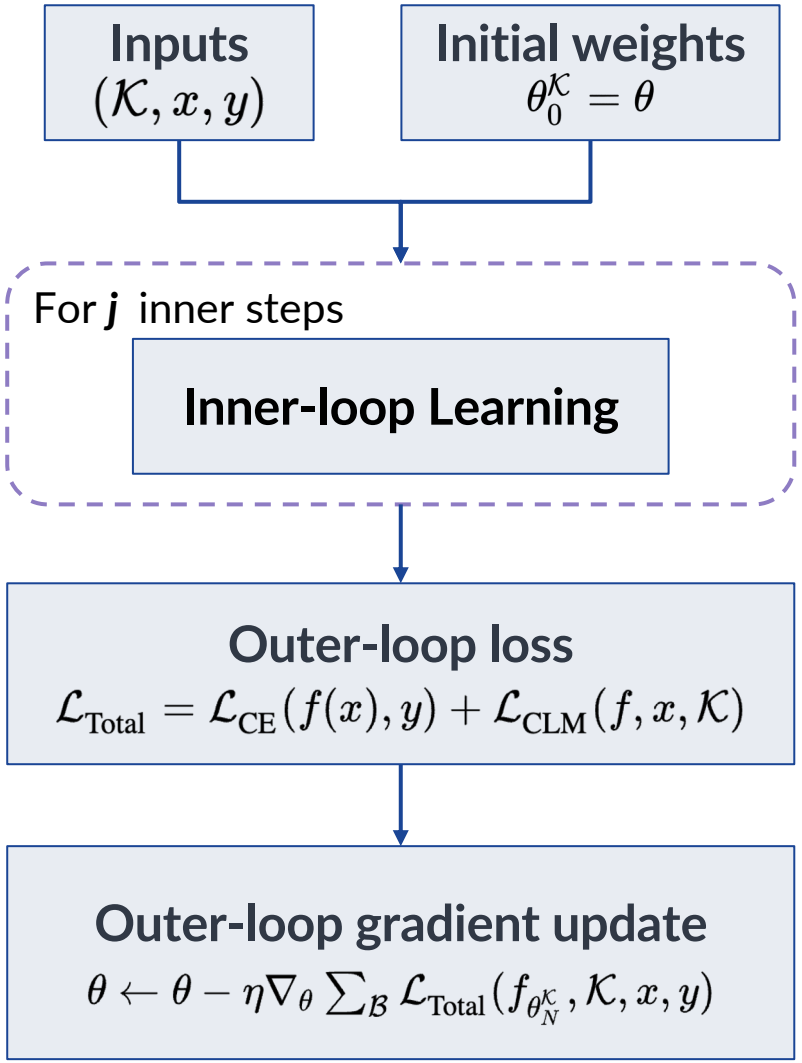
Outer-loop Learning



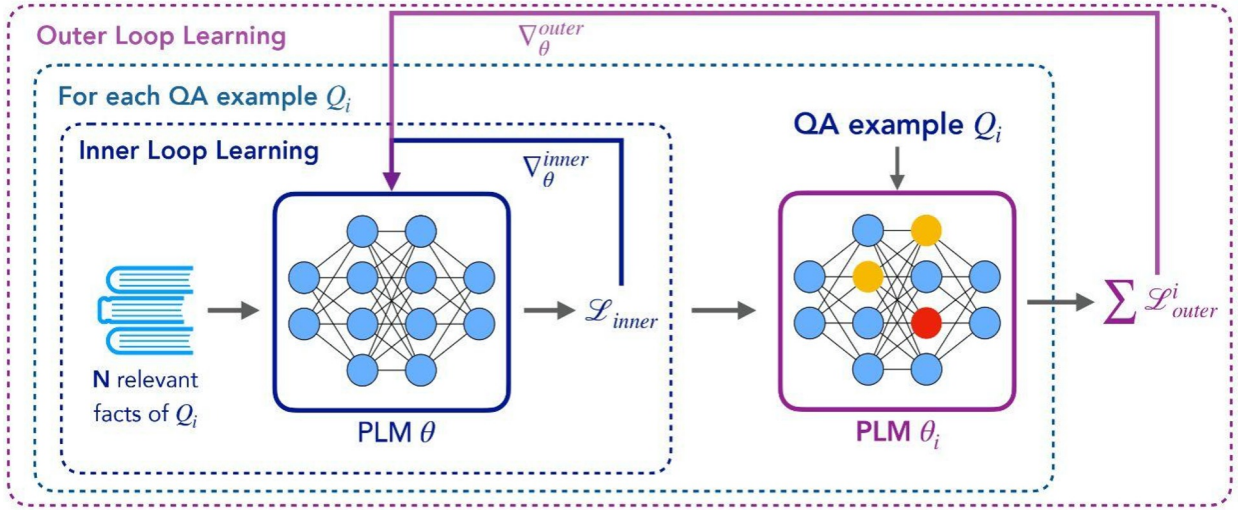
- **Closed-Book** question answering



Outer-loop Learning

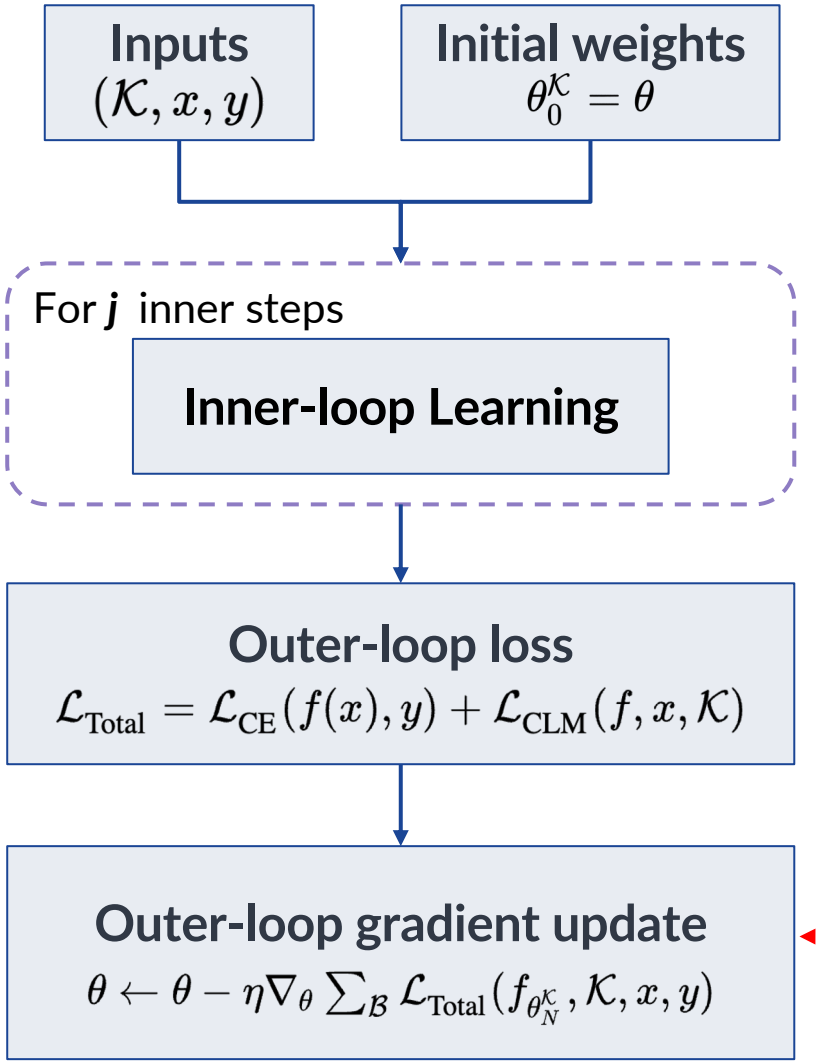


- **Closed-Book** question answering

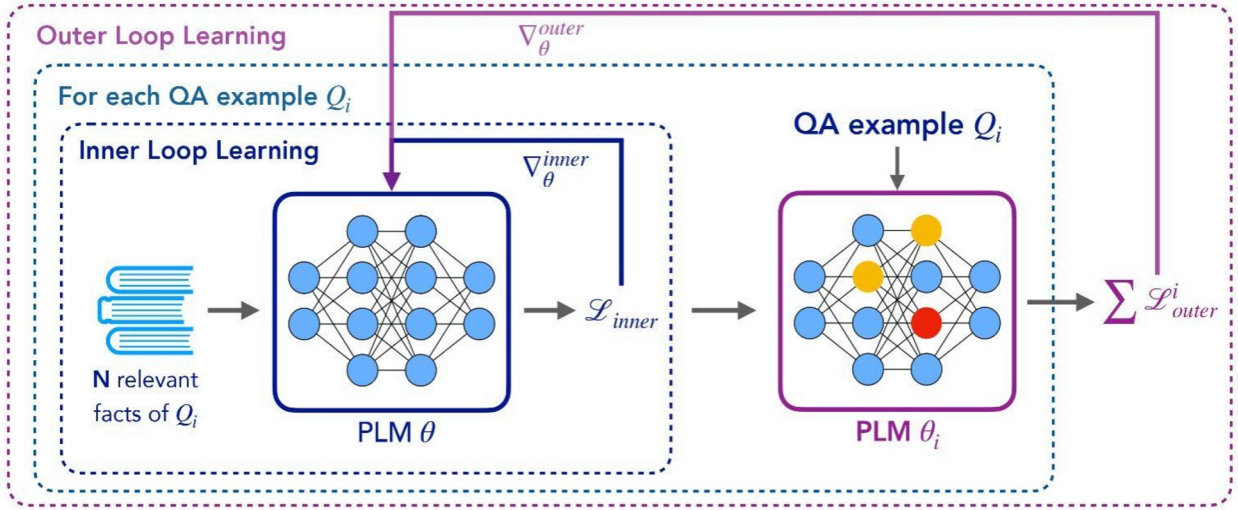


Learning to reason and generate relevant knowledge

Outer-loop Learning

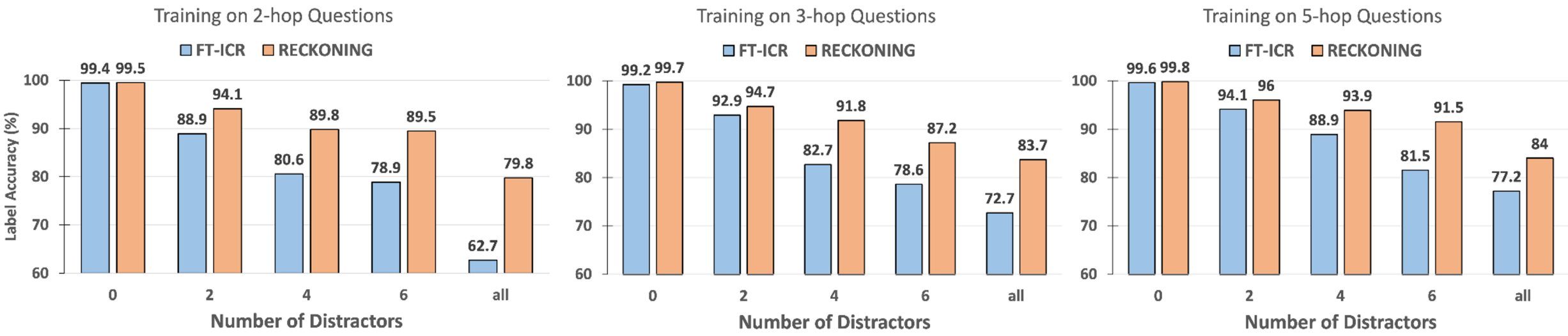


- **Closed-Book** question answering



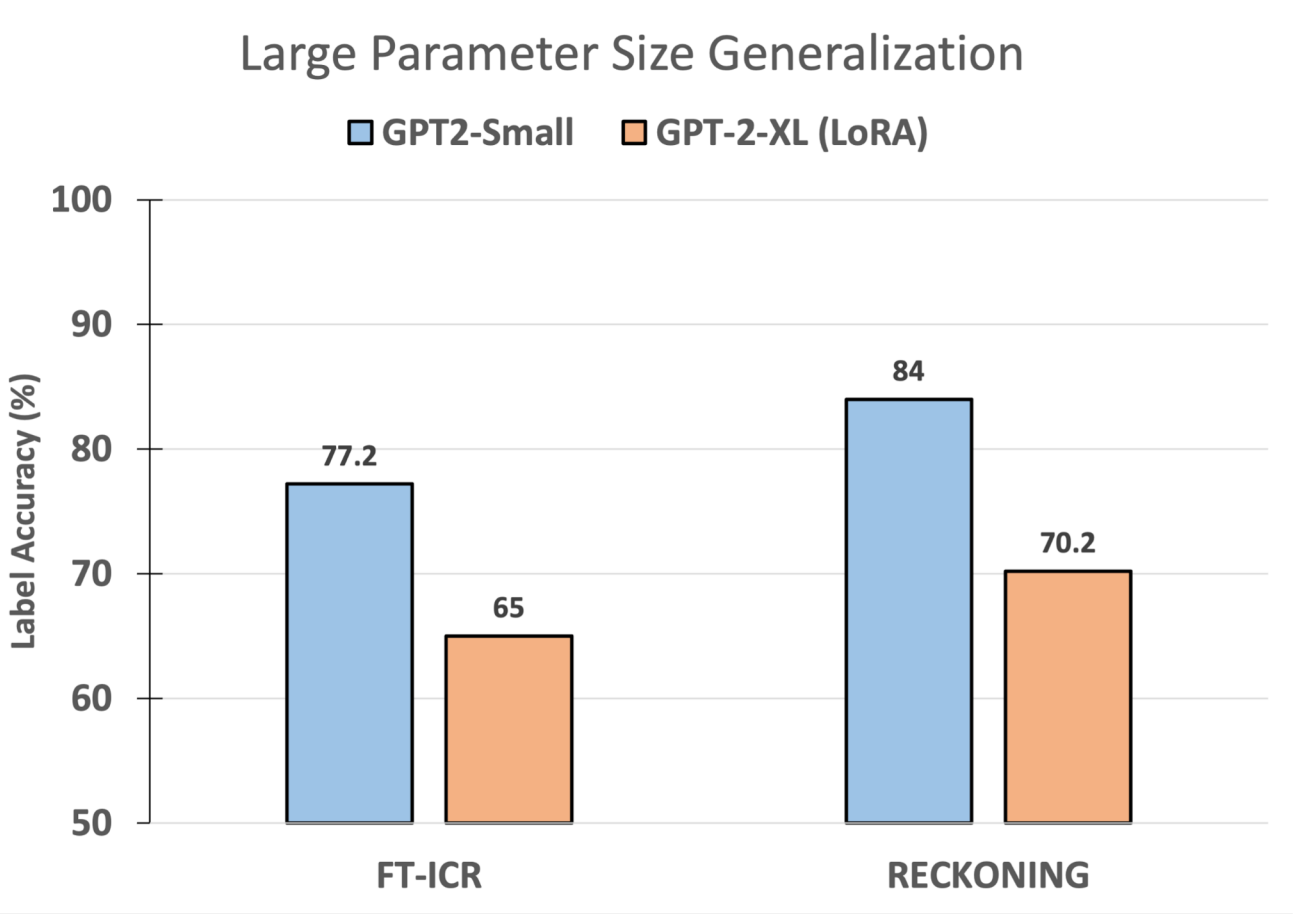
Learning the optimal meta-parameters

Robustness to Distractors



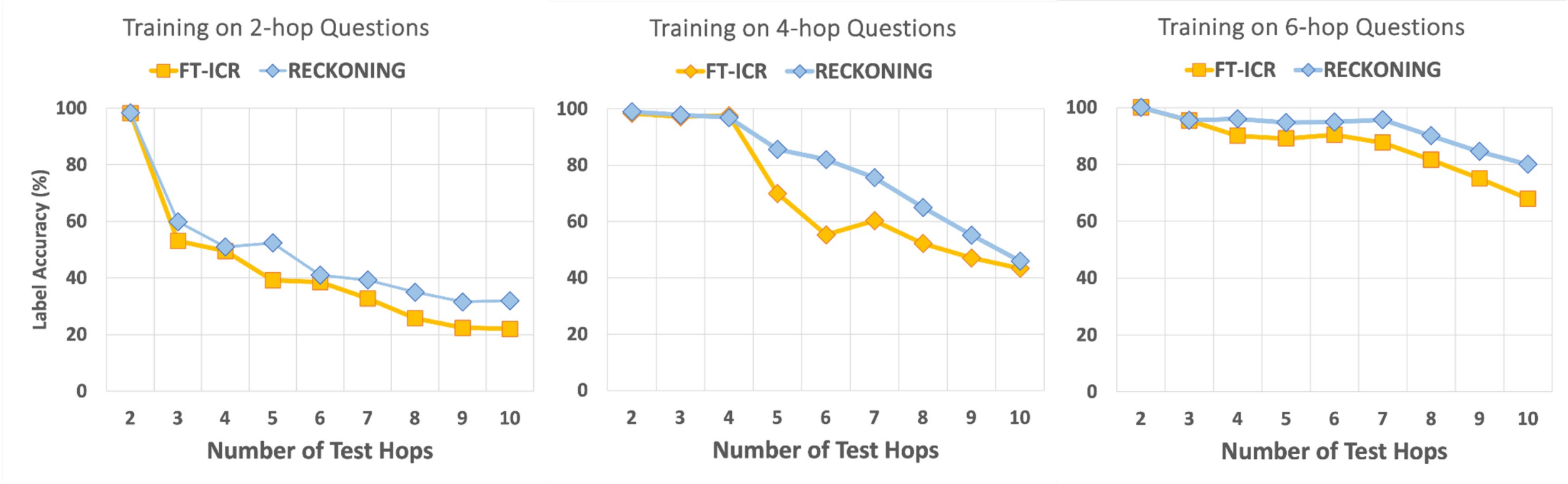
RECKONING consistently outperforms the baseline when there are irrelevant distractors in the context.

Generalizing to Large Models



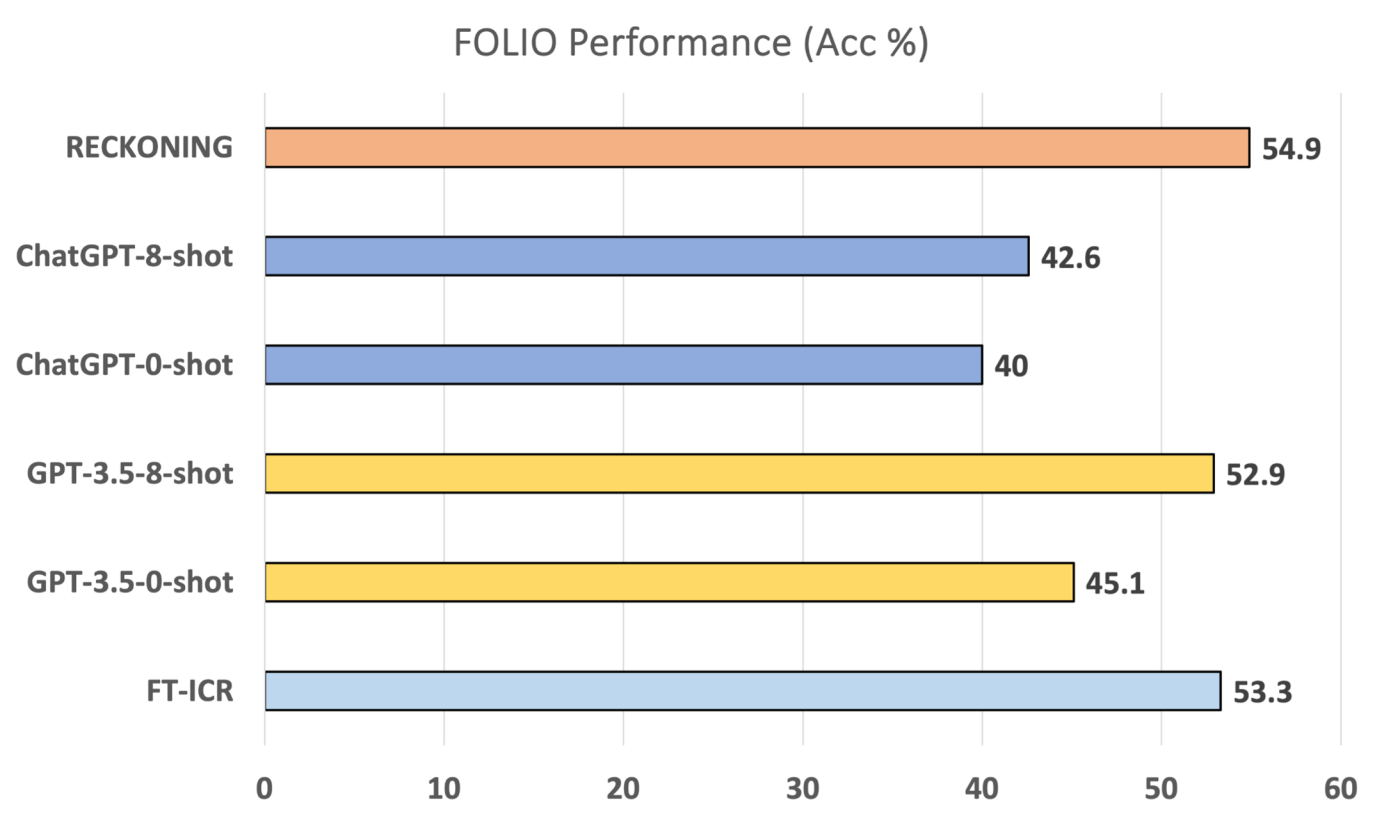
RECKONING's benefit on robustness to distractors generalize to larger models

Longer Reasoning Chain Generalization



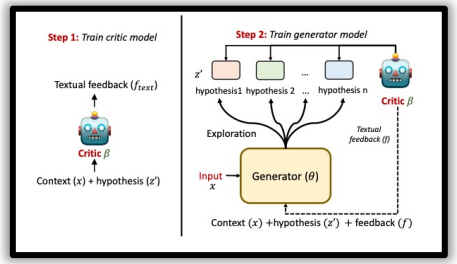
RECKONING shows better generalization to problems requiring longer reasoning chains than training examples.

Reasoning on Real-World Knowledge



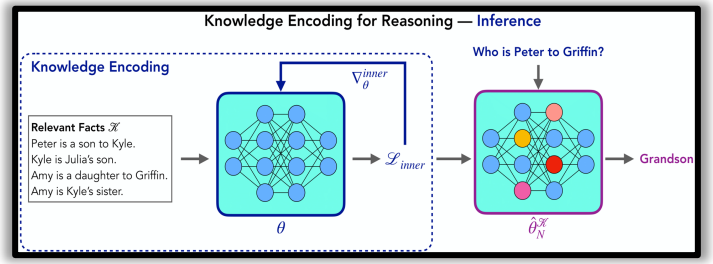
The performance gain of RECKONING generalizes to real-world knowledge and reasoning

Outline



Refine knowledge using feedback

Paul et.al. 2024 (EACL)

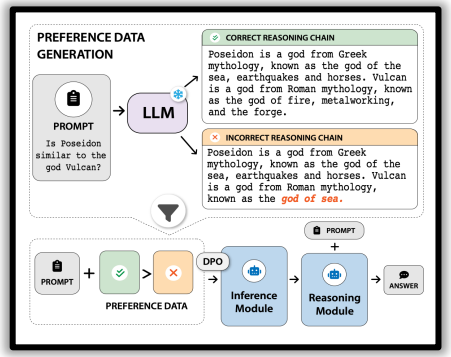


Robust Reasoning by updating world models

Chen et.al. 2023 (Neurips)

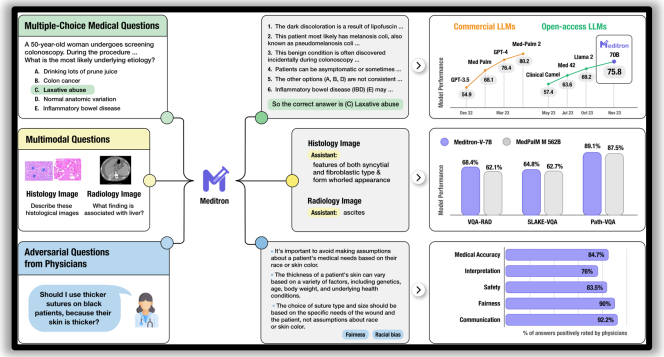
Paul et.al. 2024 (EMNLP Findings)

Faithful Reasoning about knowledge

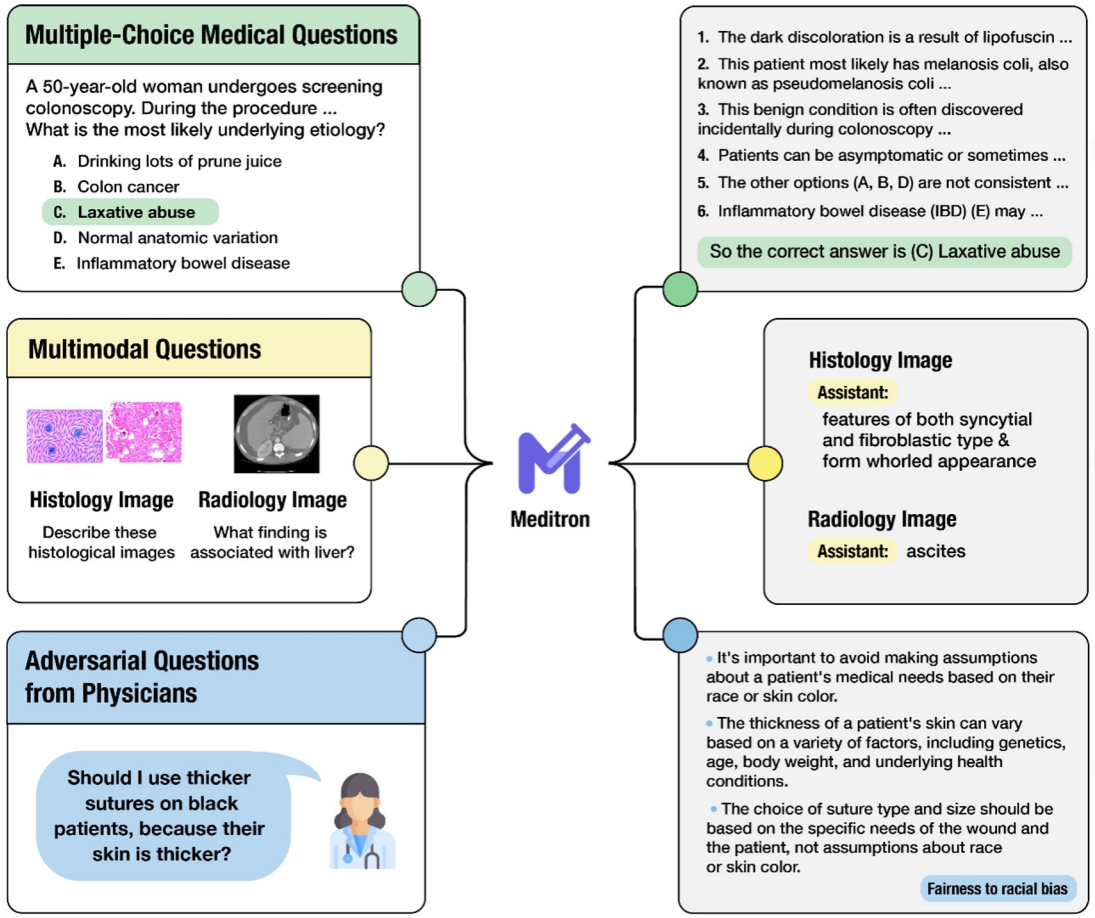


Chen et.al. 2024

Real-World Reasoning



MEDITRON-70B:
Open Medical Foundation Models Adapted for Clinical Practice



EPFL

idiap RESEARCH INSTITUTE

Yale University School of Medicine

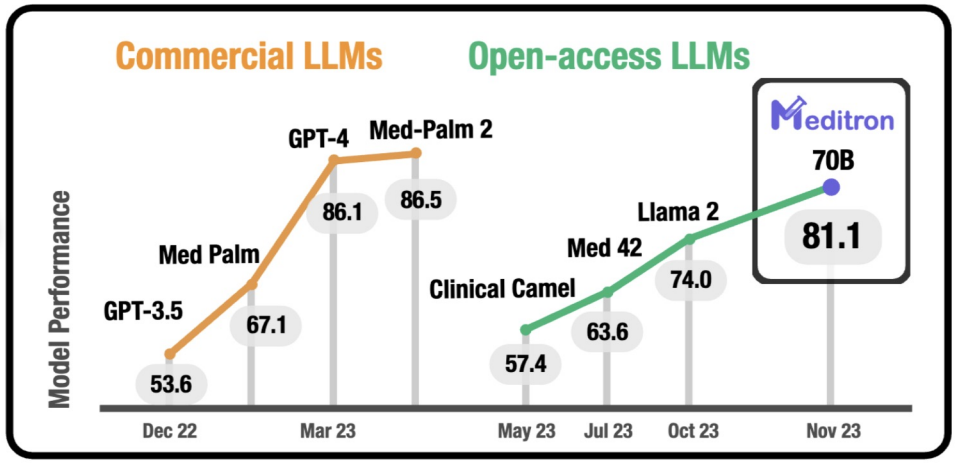
Open Assistant

COMITE INTERNATIONAL GENEVE
ICRC

CHUV Centre hospitalier universitaire vaudois

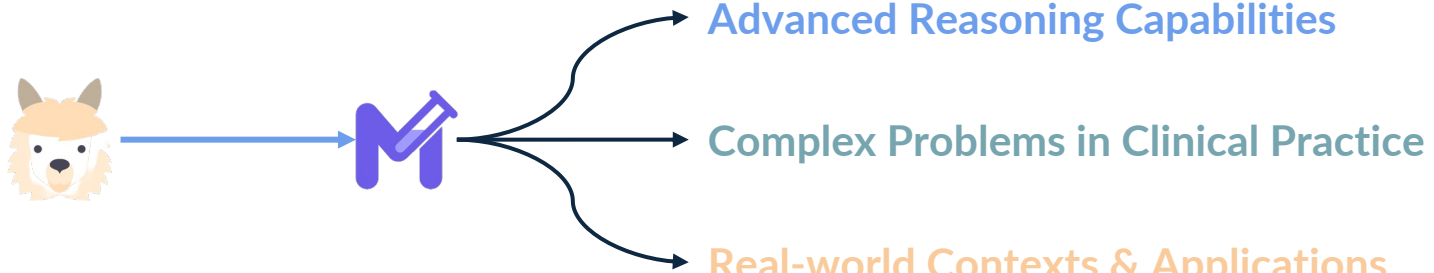
INSELSPITAL
UNIVERSITÄTSSPITAL BERN
HÔPITAL UNIVERSITAIRE DE BERNE

Open Medical Foundation Models

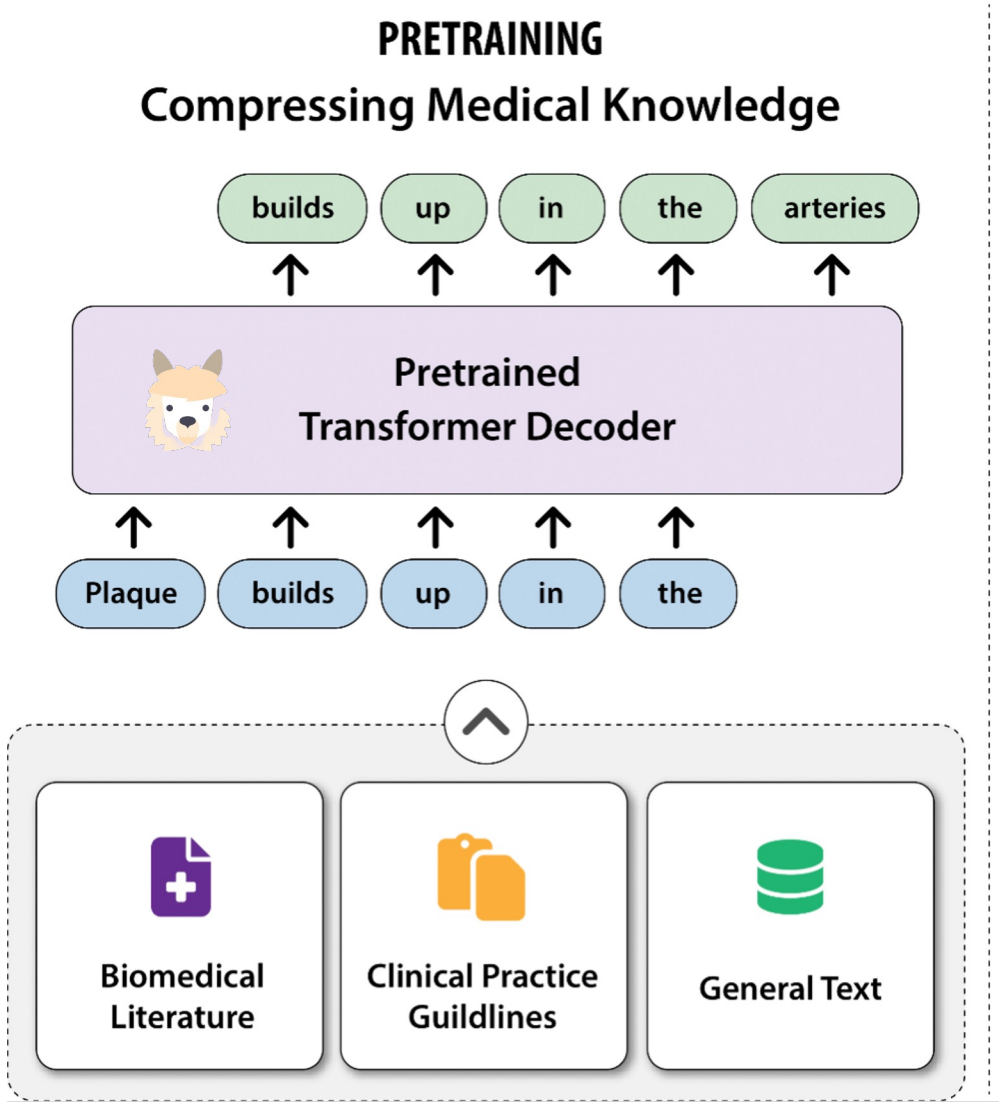


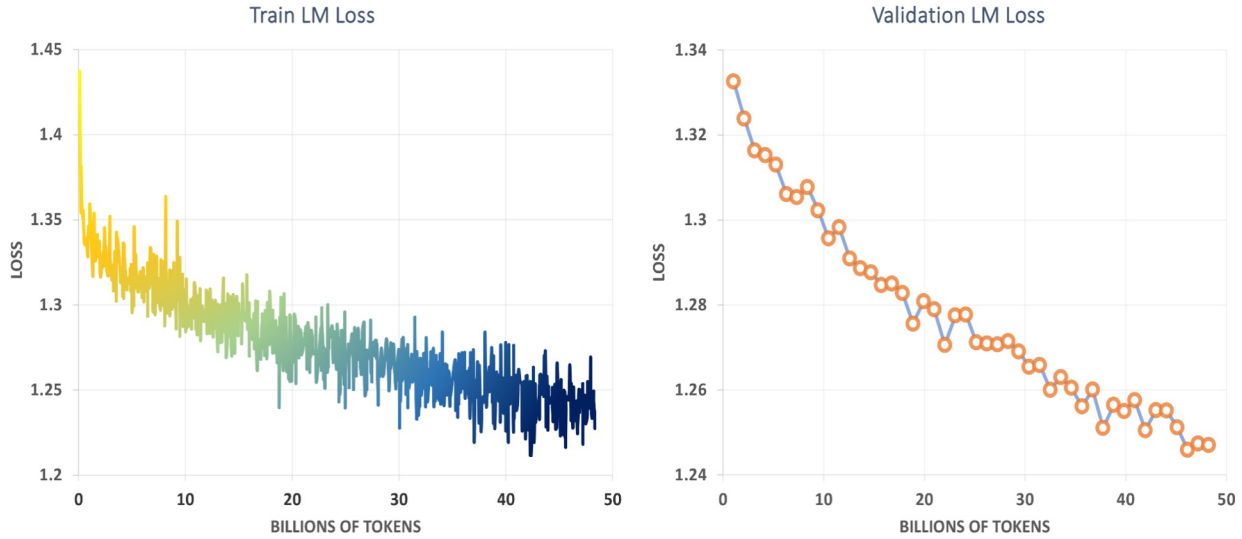
Significant performance gap between Open and Commercial foundation models

Developing Powerful Open Medical Foundation Models



Scaling up Medical Pretraining



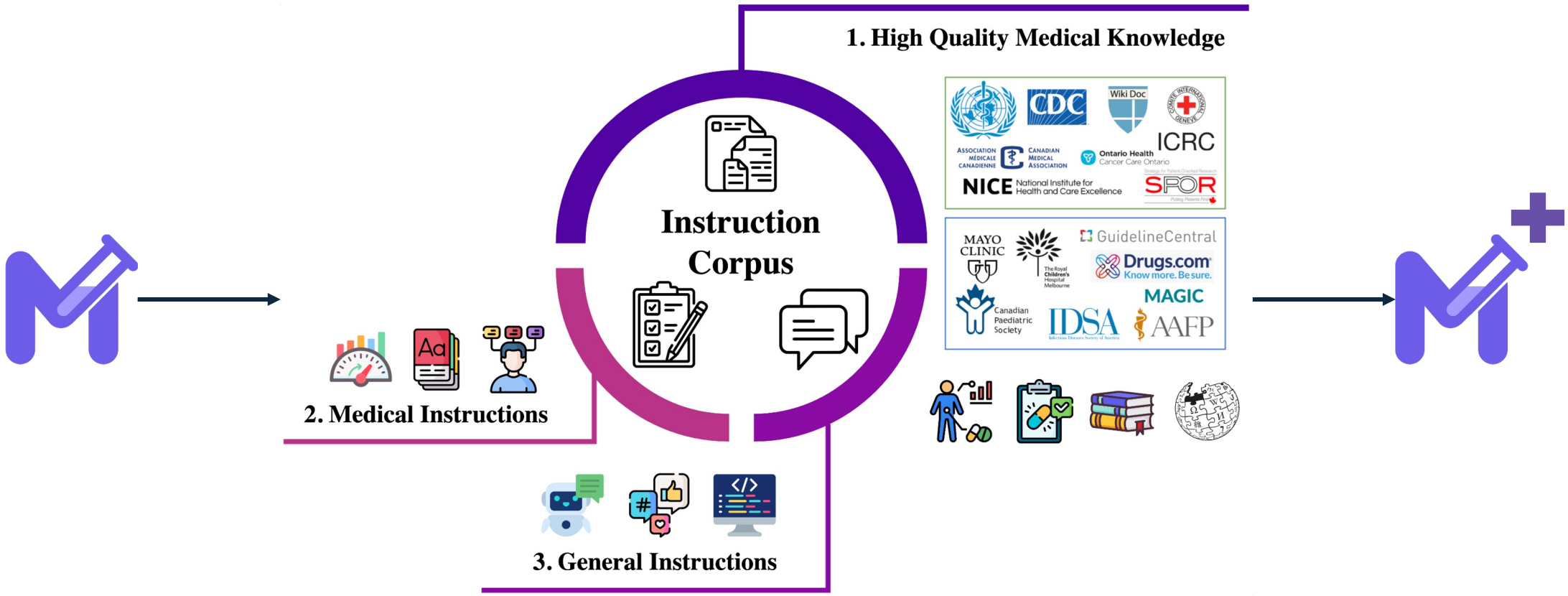


Compute:
128 A100 (80GB) GPUs

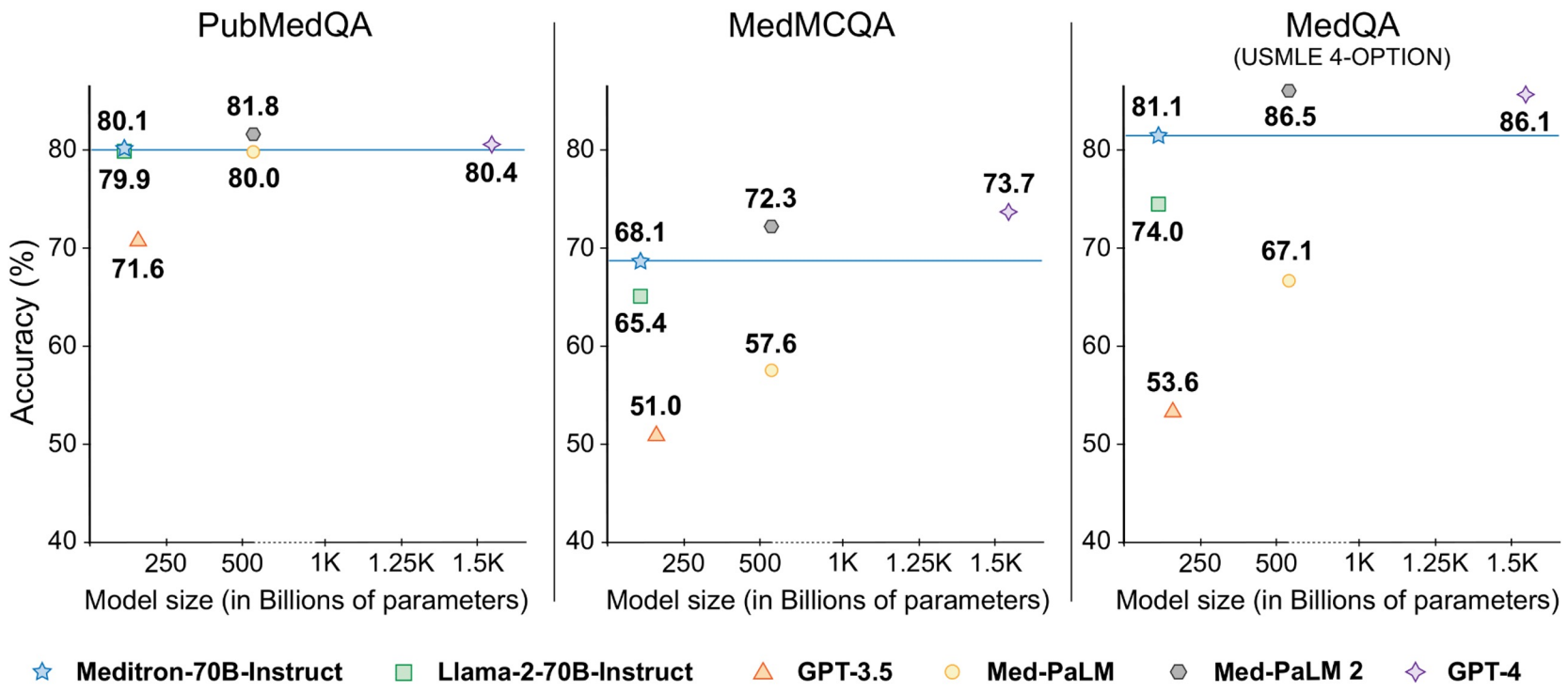
Time:
13 days (332 hours)

Carbon Emission:
486 kg CO2

Annealing and Instruction Tuning

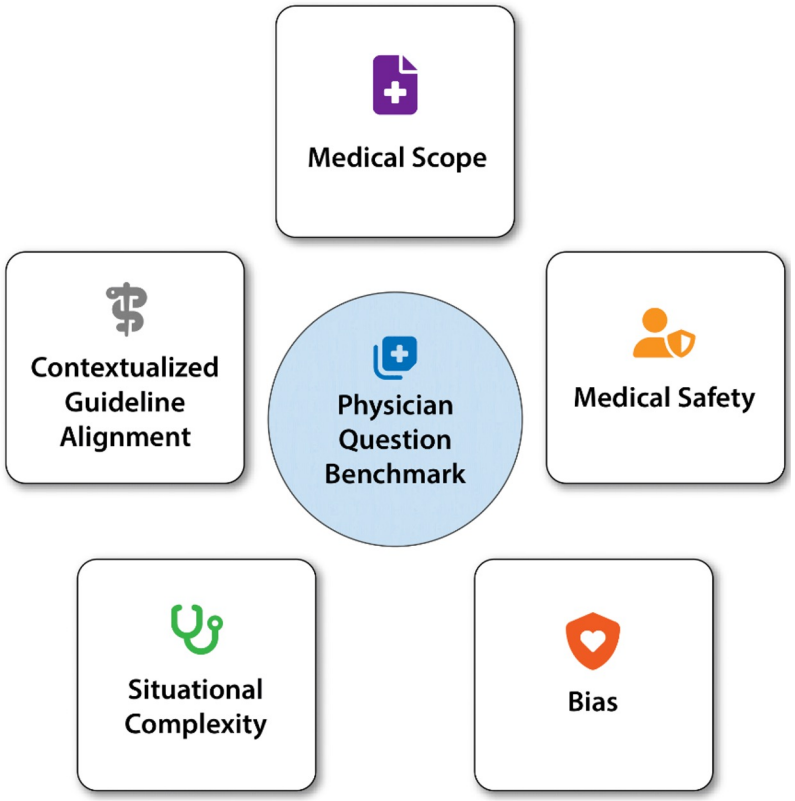


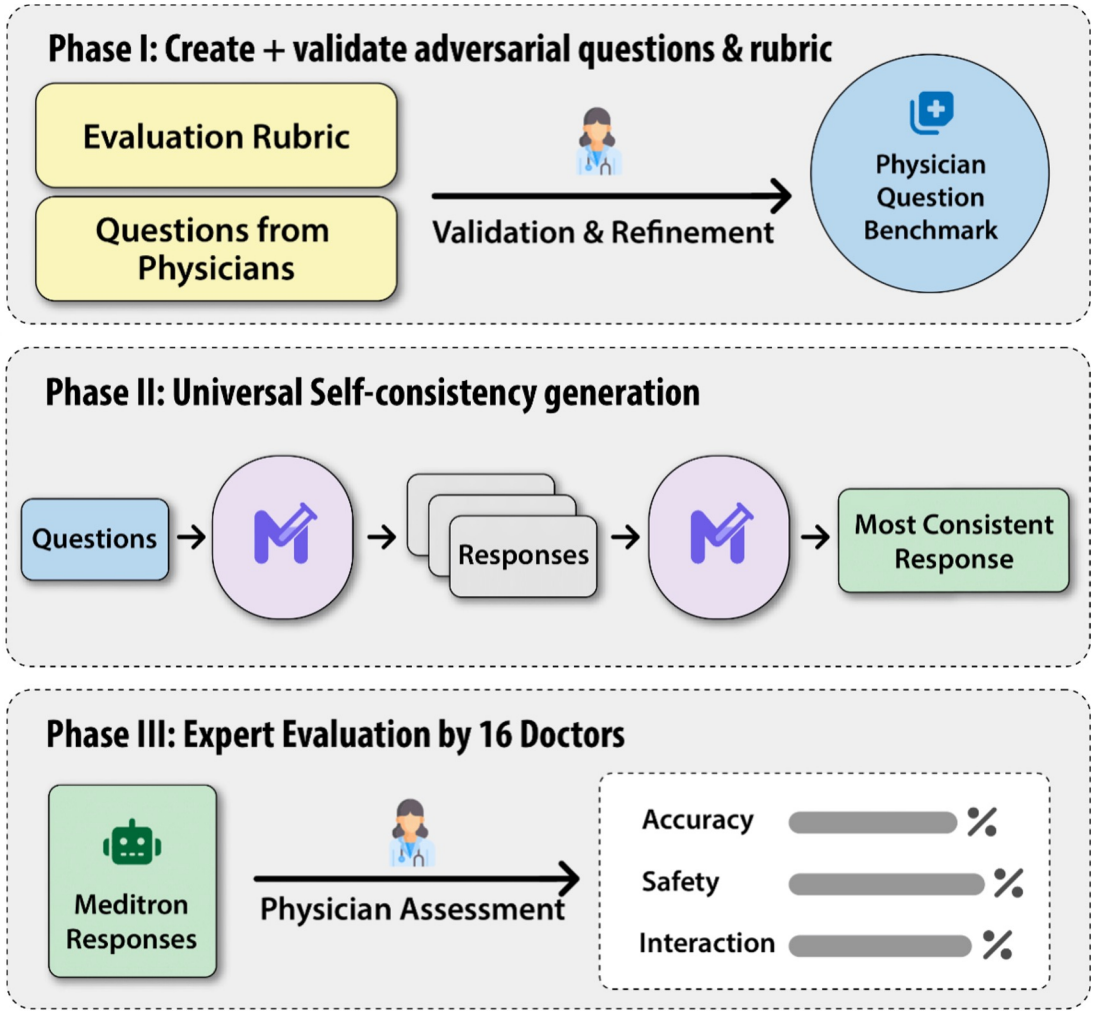
Standard Medical Benchmark Evaluation



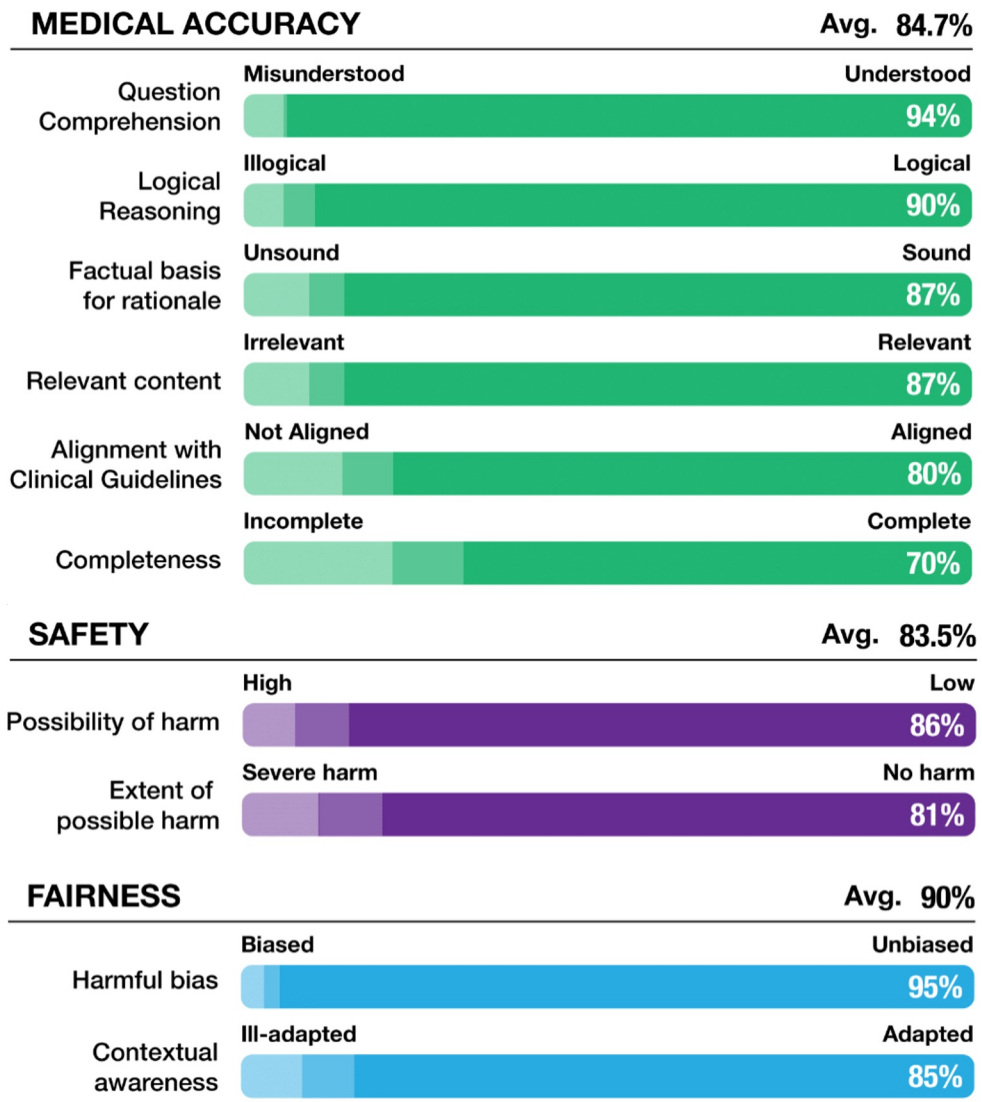
- MEDITRON-70B significantly improves over open-access models and several SOTA commercial LLMs
- MEDITRON-70B is within 5% of GPT-4 and 5.4% of Med-PaLM-2

Physician-driven Evaluation





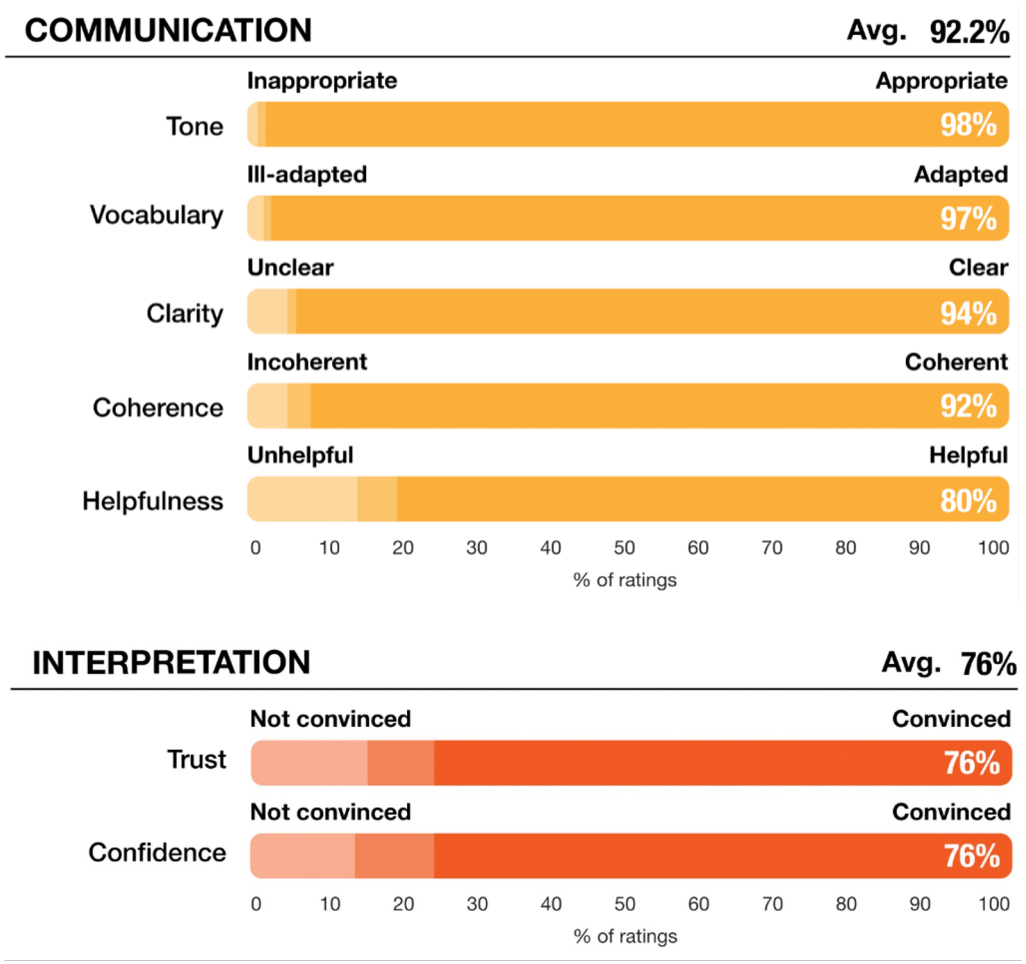
Physician-driven Evaluation: Results



84.7% Aligned with clinical practice guidelines and scientific consensus

83.5% Safe with a low likelihood of causing harm, fair, and context-aware

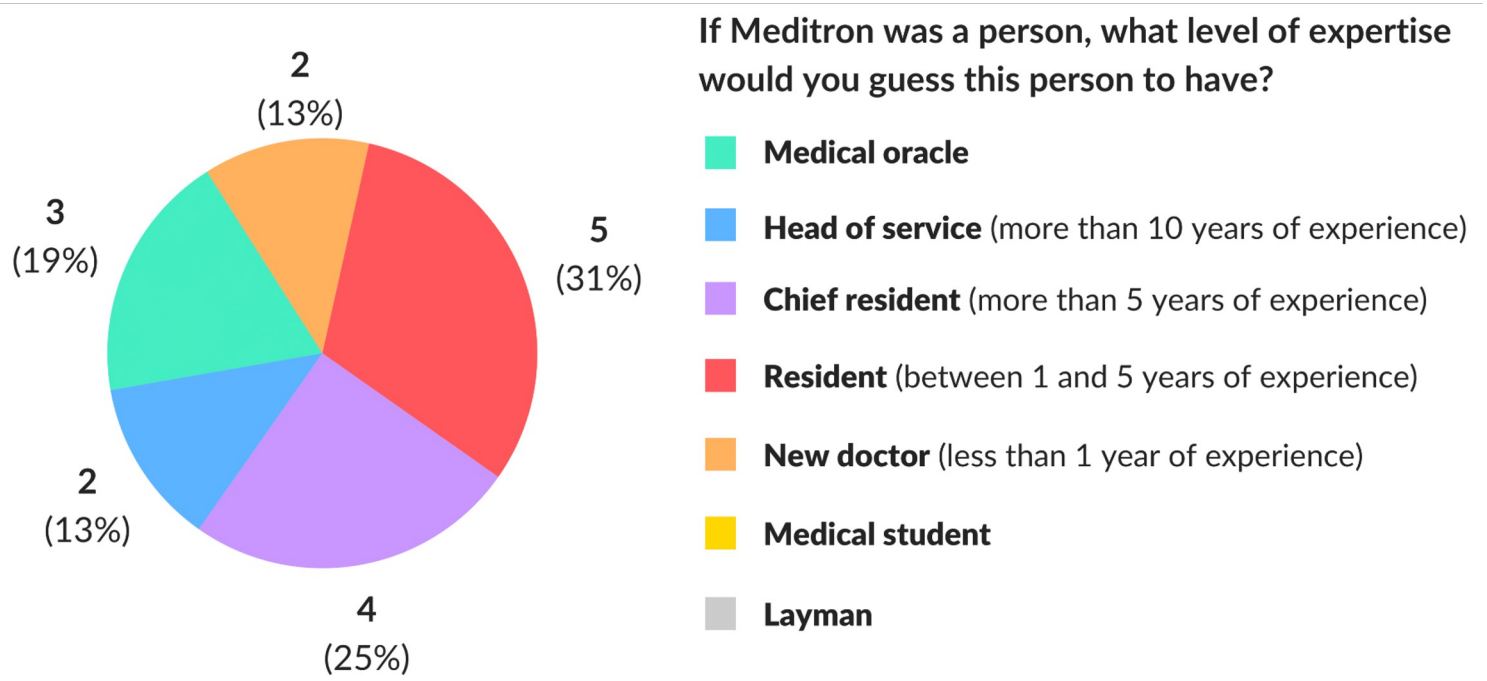
Physician-driven Evaluation: Results



92.2% Human-level communication skills for effective patient and physician interactions

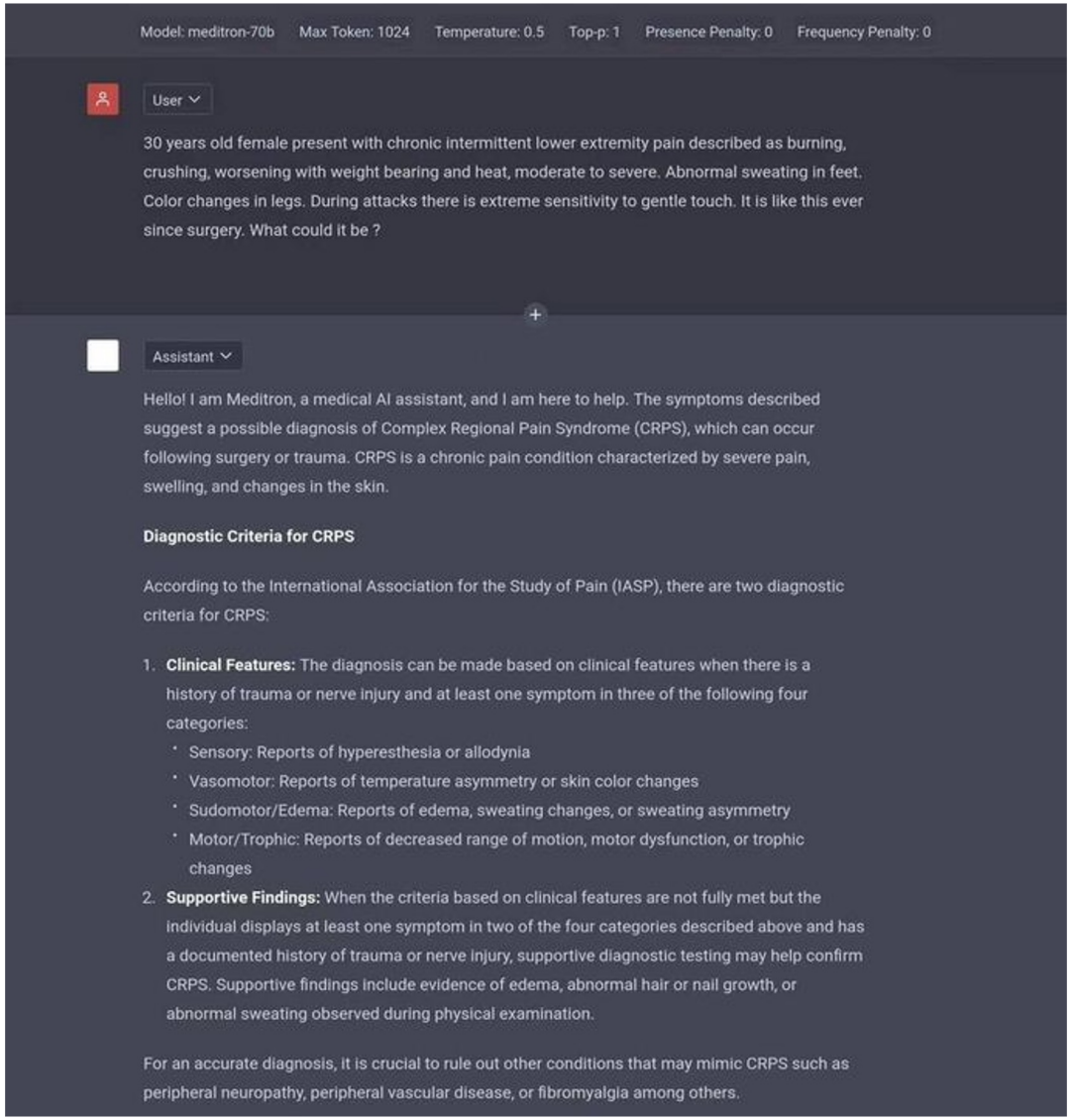
76% Convincing to medical professionals and exhibit an appropriate level of confidence

Physician-driven Evaluation: Results

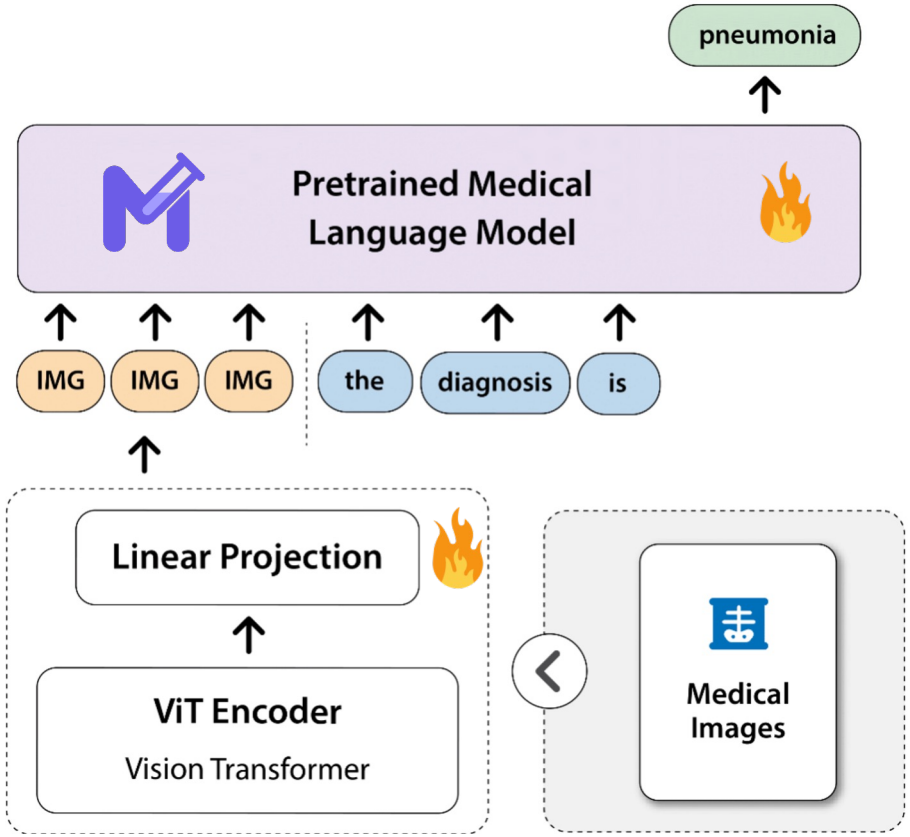


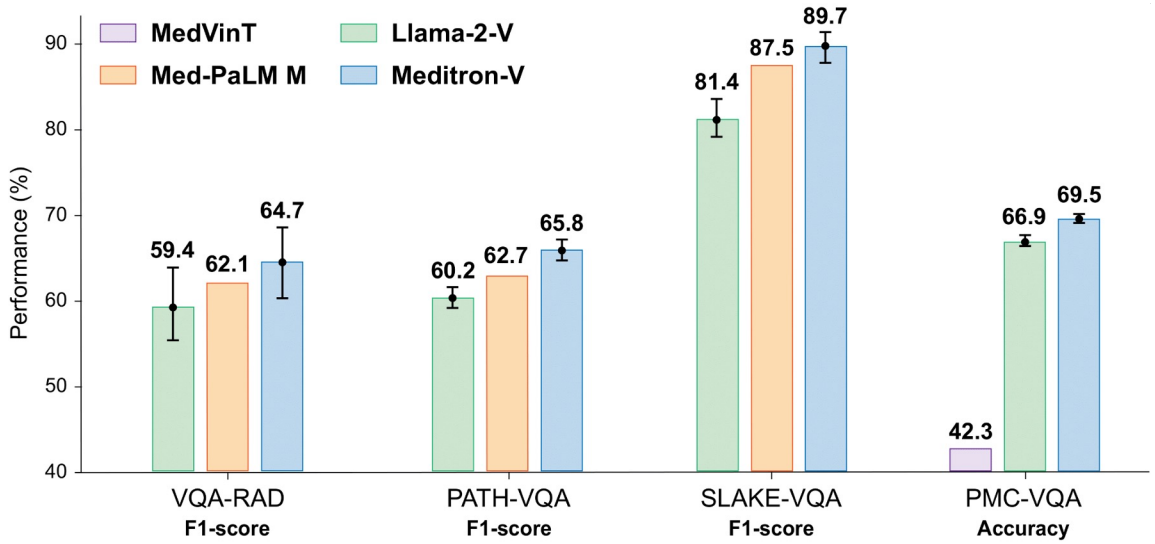
- 14/16 physicians find that MEDITRON-70B's level of expertise is equivalent to or higher than a Resident
- 9 physicians find the model's level of expertise surpasses a Resident

Real-world Clinical Reasoning: Example



Multimodal Foundation for Multimodal Reasoning





Meditron-V (7B) outperforms both open-access models and the commercial SOTA, Med-PaLM M (562B) on all benchmarks

Continue pretraining improves multimodal adaptation in the medical domain

Conclusion

- Interactive method improves the quality of inference generation
- Counterfactual reasoning can improve the faithfulness and performance of LMs.
- RECKONING's reasoning is more robust to irrelevant information in the context.
- Stronger foundation Meditron leads to stronger reasoning performance in medicine
- Stronger foundation Meditron is a stronger fit for advanced reasoning strategies

Conclusion

- Interactive method improves the quality of inference generation
- Counterfactual reasoning can improve the faithfulness and performance of LMs.
- RECKONING's reasoning is more robust to irrelevant information in the context.
- Stronger foundation Meditron leads to stronger reasoning performance in medicine
- Stronger foundation Meditron is a stronger fit for advanced reasoning strategies

Thank you!