

Enhancing Zero-Shot Chain-of-Thought Reasoning in Large Language Models through Logic

Xufeng Zhao, Mengdi Li, Wenhao Lu, Cornelius Weber, Jae Hee Lee, Kun Chu, and Stefan Wermter

University of Hamburg

Knowledge Technology Research Group

LREC-COLING  **2024**



Motivation

Chain-of-Thought Prompting (CoT)

- CoT to **improve** performance

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

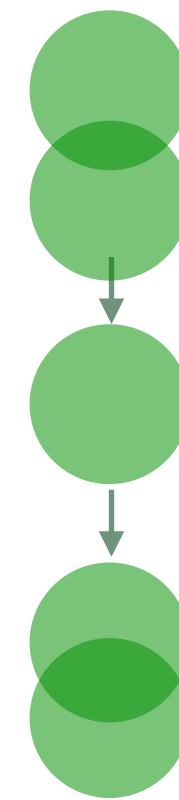
A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

[1] Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. "Chain-of-thought prompting elicits reasoning in large language models." *Advances in Neural Information Processing Systems* 35 (2022): 24824-24837.

Motivation

Chain-of-Thought Prompting (CoT)

- CoT **improves** performance
- **Zero-shot** CoT works



Mike plays ping pong for 40 minutes. In the first 20 minutes, he scores 4 points. In the second 20 minutes, he scores 25% more points. How many total points did he score?

Let's think step by step.

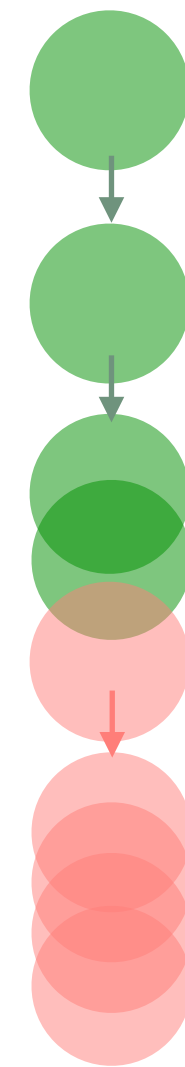
Answer:

- #1.** In the first 20 minutes, Mike scores 4 points.
- #2.** In the second 20 minutes, he scores 25% more points than in the first 20 minutes.
- #3.** To find 25% more points, we can multiply the number of points in the first 20 minutes by 1.25.
- #4.** 4 points * 1.25 = 5 points.
- #5.** In total, Mike scores 4 points + 5 points = $\langle\langle 4+5=9 \rangle\rangle$ 9 points. Answer: 9.

Motivation

Chain-of-Thought Prompting (CoT)

- CoT **improves** performance
- **Zero-shot** CoT works
- Fail if an intermediate step goes **wrong!**



An aeroplane flies along the four sides of a square at the speeds of 200, 400, 600 and 800km/hr. Find the average speed of the plane around the field? Options: OptA) 384 OptB) 562 OptC) 458 OptD) 156 OptE) 452

Let's think step by step.

Answer:

#1. To find the average speed, we need to find the total distance traveled and divide it by the total time taken.

#2. The distance traveled on each side of the square is equal to the speed ...

#...

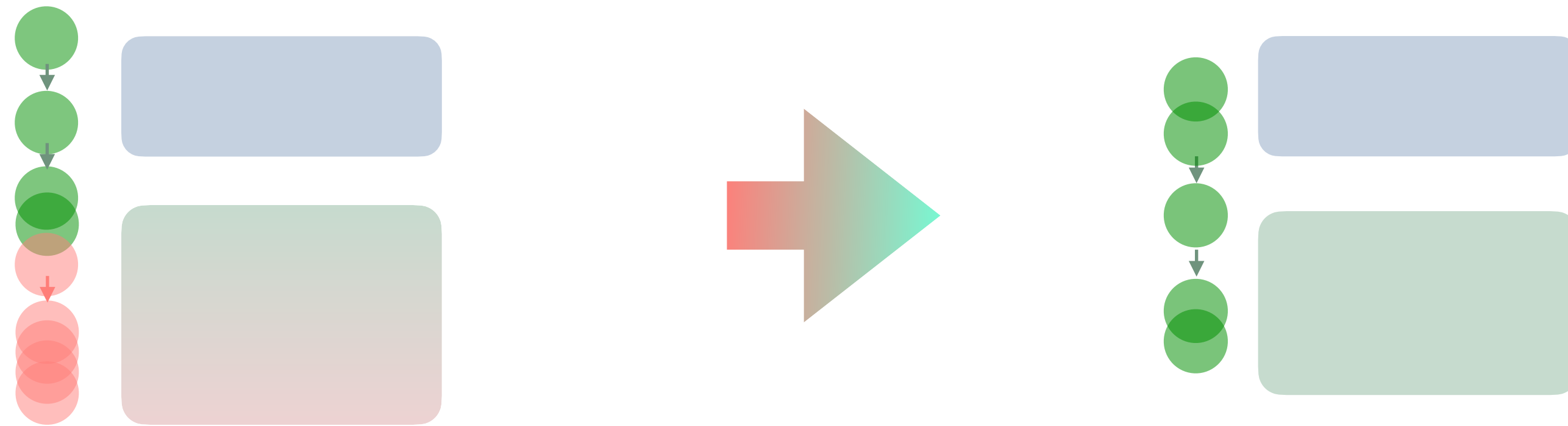
#7. Simplifying the expression, we get the average speed as $(4s) / ((3s + 2s + (4/3)s + (1/2)s) / 2400)$ km/hr.

#...

#17. None of the given options match the calculated average speed.

Motivation

Fail if an intermediate step goes **wrong!**



Can LLMs self-correct the reasoning chain?

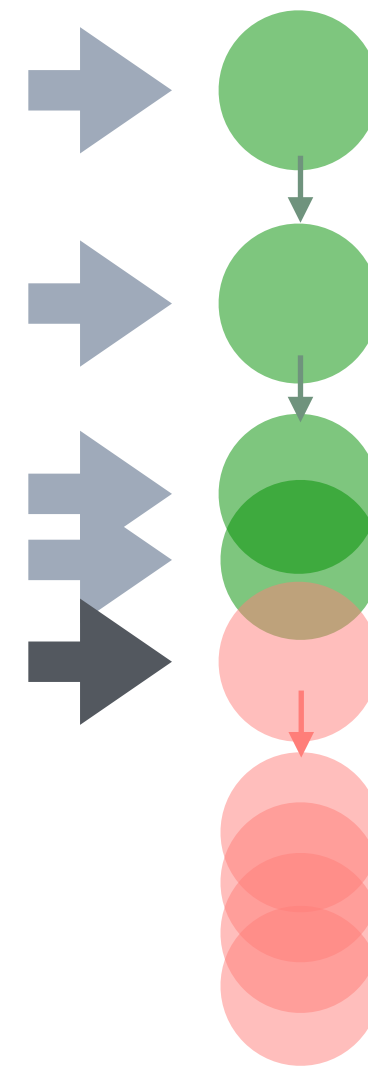
And how?

And how?

Motivation

Chain-of-Thought Prompting (CoT)

- **Verify** step by step
 - Prompt to self-check every step directly
 - Intuitive, biased, less-effective
 - Ground to logic
 - Guided, less-biased, interpretable
- **Revise** if necessary



An aeroplane flies along the four sides of a square at the speeds of 200, 400, 600 and 800km/hr. Find the average speed of the plane around the field? Options: OptA) 384 OptB) 562 OptC) 458 OptD) 156 OptE) 452

Let's think step by step.

Answer:

#1. To find the average speed, we need to find the total distance traveled and divide it by the total time taken.
#2. The distance traveled on each side of the square is equal to the speed ...

#...

#7. Simplifying the expression, we get the average speed as $(4s) / ((3s + 2s + (4/3)s + (1/2)s) / 2400)$ km/hr.

#...

#17. None of the given options match the calculated average speed.

Example

Guided reasoning with Logic

If Tom plays football outside, then John will also join to play; if John plays football, then Mary won't go outside. Knowing that Mary is outside, is Tom playing football?

XU

You

If Tom plays football outside, then John will also join to play; if John plays football, then Mary won't go outside. Knowing that Mary is outside, is Tom playing football?



ChatGPT

Based on the information provided:

1. If Tom plays football outside, then John will also join to play.
2. If John plays football, then Mary won't go outside.
3. Mary is outside.

From the given information, we cannot conclusively determine whether Tom is playing football or not. The information about John playing football does not provide a direct link to Tom's activities, and Mary being outside doesn't provide information about Tom's specific actions.

Example

Guided reasoning with Logic

If Tom plays football outside, then John will also join to play; if John plays football, then Mary won't go outside. Knowing that Mary is outside, is Tom playing football?

XU

You

use contrapositive to deduce again



ChatGPT

Certainly! Let's reevaluate the information using the contrapositive of the given statements:

1. If Tom plays football outside, then John will also join to play. (Contrapositive: If John doesn't join to play, then Tom didn't play football outside.)
2. If John plays football, then Mary won't go outside. (Contrapositive: If Mary goes outside, then John didn't play football.)

Now, given that Mary is outside, we can use the contrapositive of statement 2 to conclude that John didn't play football. Therefore, from the contrapositive of statement 1, we can further deduce that Tom didn't play football outside either.

So, with the information that Mary is outside, we can deduce that Tom is not playing football outside.

LoT: Logical Chain of Thoughts

Verify step by step

- Chain of Thought Formulation

$$\{P, T_1, T_2, \dots, T_N\}$$

- Verify the i -th step

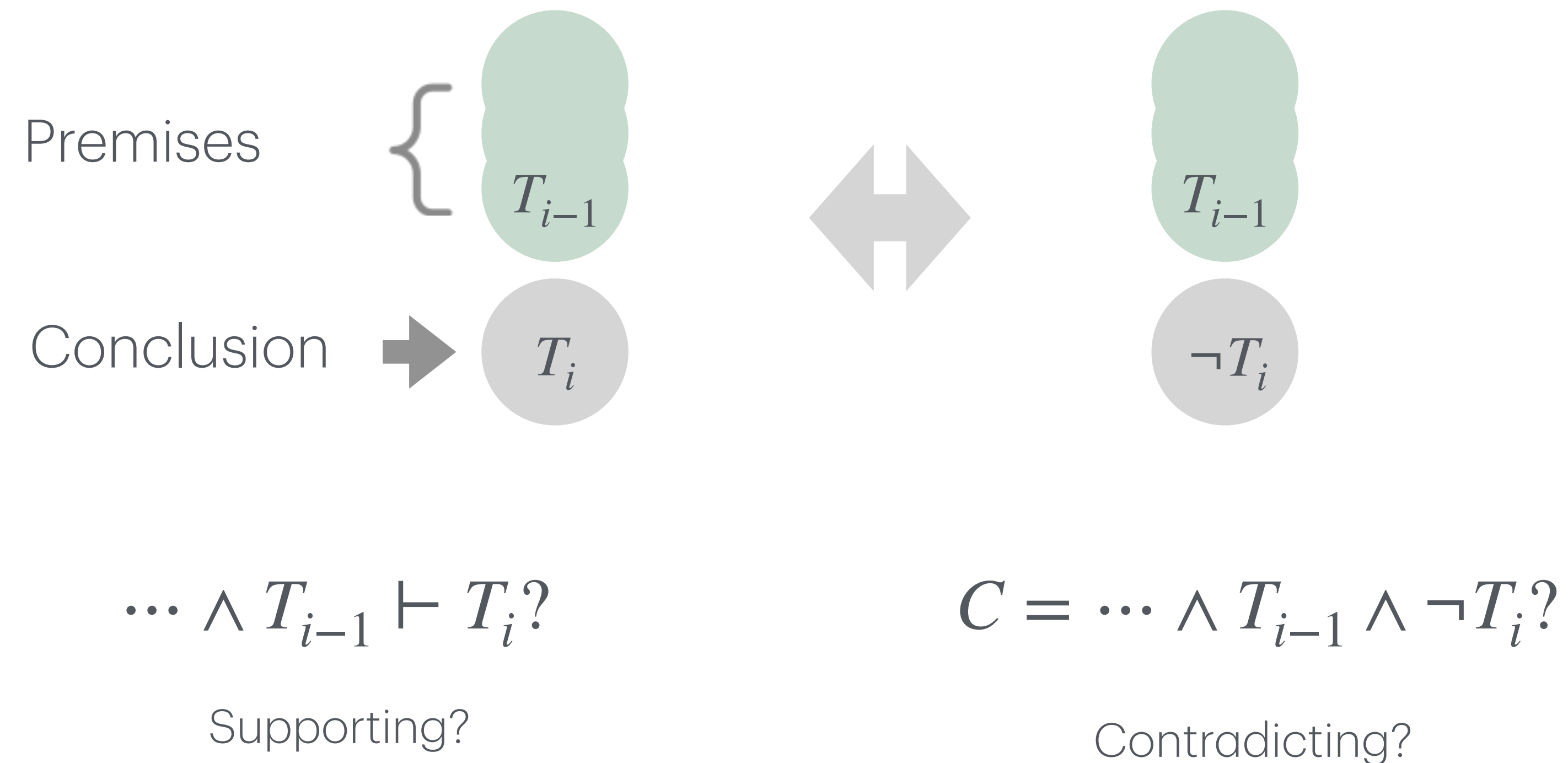
$$P, \dots, T_{i-1} \vdash T_i?$$



LoT: Logical Chain of Thoughts

Verify step by step

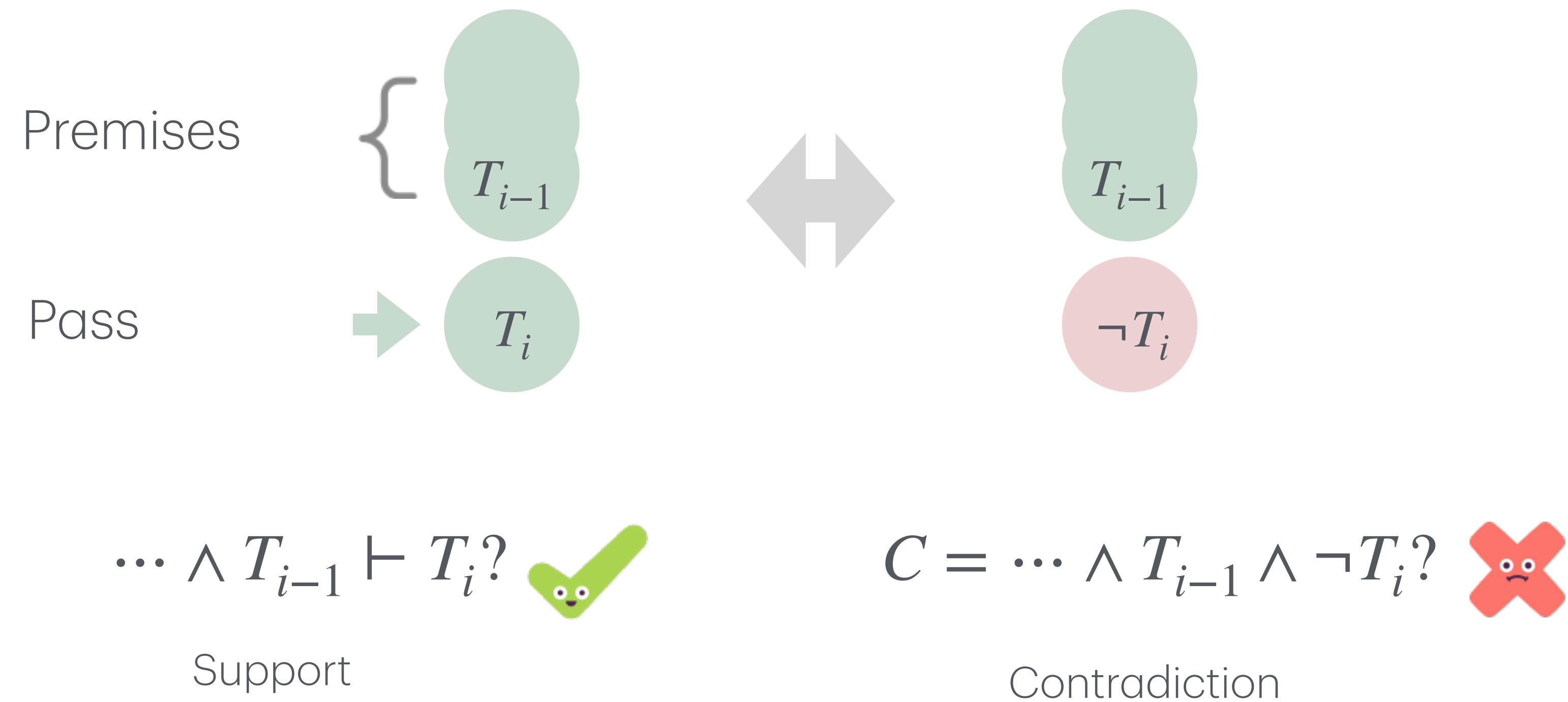
- Reductio ad Absurdum (reduction to absurdity)



LoT: Logical Chain of Thoughts

Revise on demand

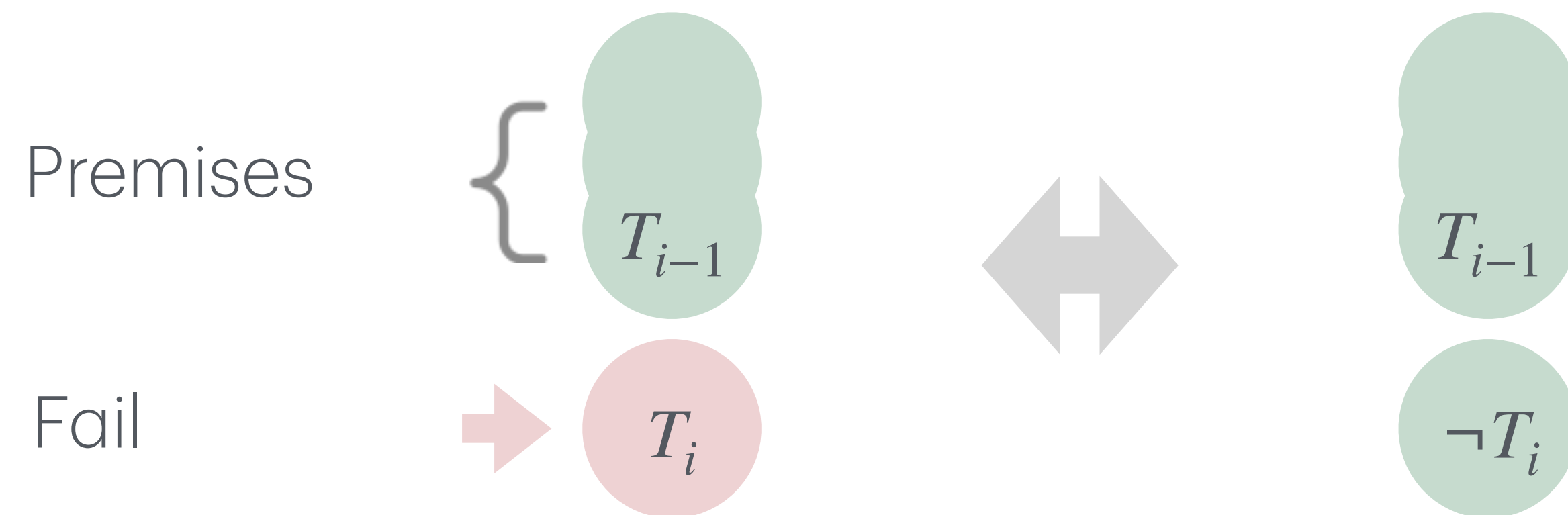
- Continue



LoT: Logical Chain of Thoughts

Revise on demand

- Continue
- Revise



$$\dots \wedge T_{i-1} \vdash T_i? \quad \text{✗}$$

Non-support

$$C = \dots \wedge T_{i-1} \wedge \neg T_i? \quad \text{✓}$$

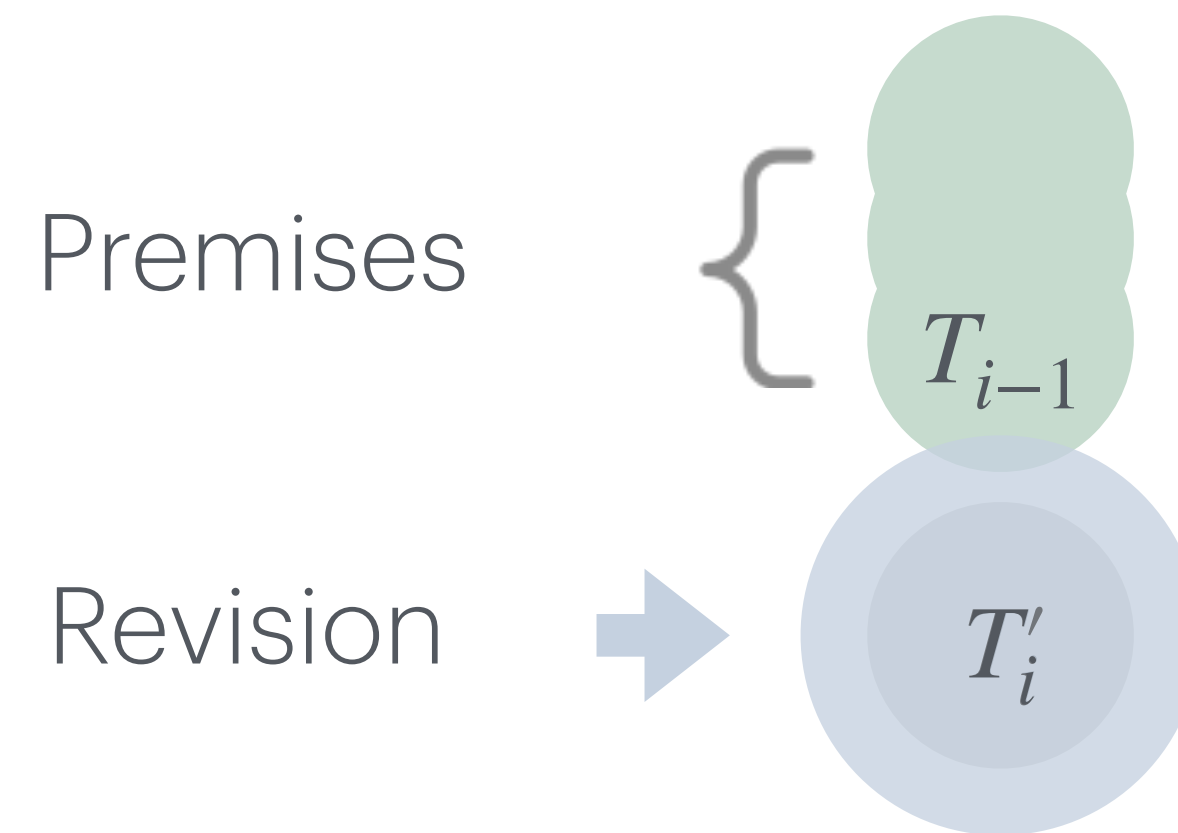
Tautology

LoT: Logical Chain of Thoughts

Revise on demand

- Continue
- Revise & re-deduce

CoT

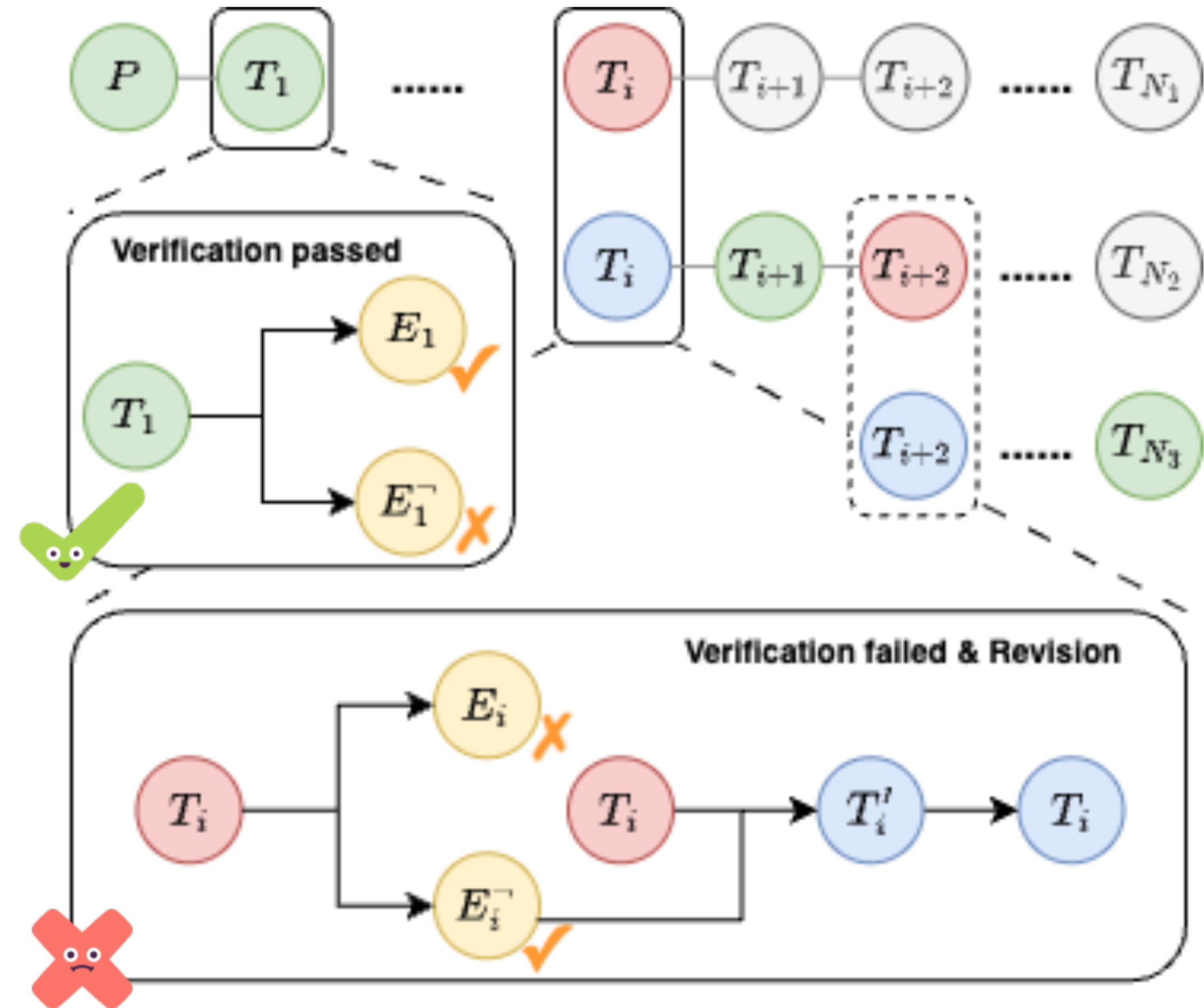


LoT

LoT: Logical Chain of Thoughts

Practical Implementation

- E_i post-hoc explanation of T_i
 - “Step i is **true** because ___”
- E_i^{-1} post-hoc explanation of $\neg T_i$
 - “Step i is **false** because ___”



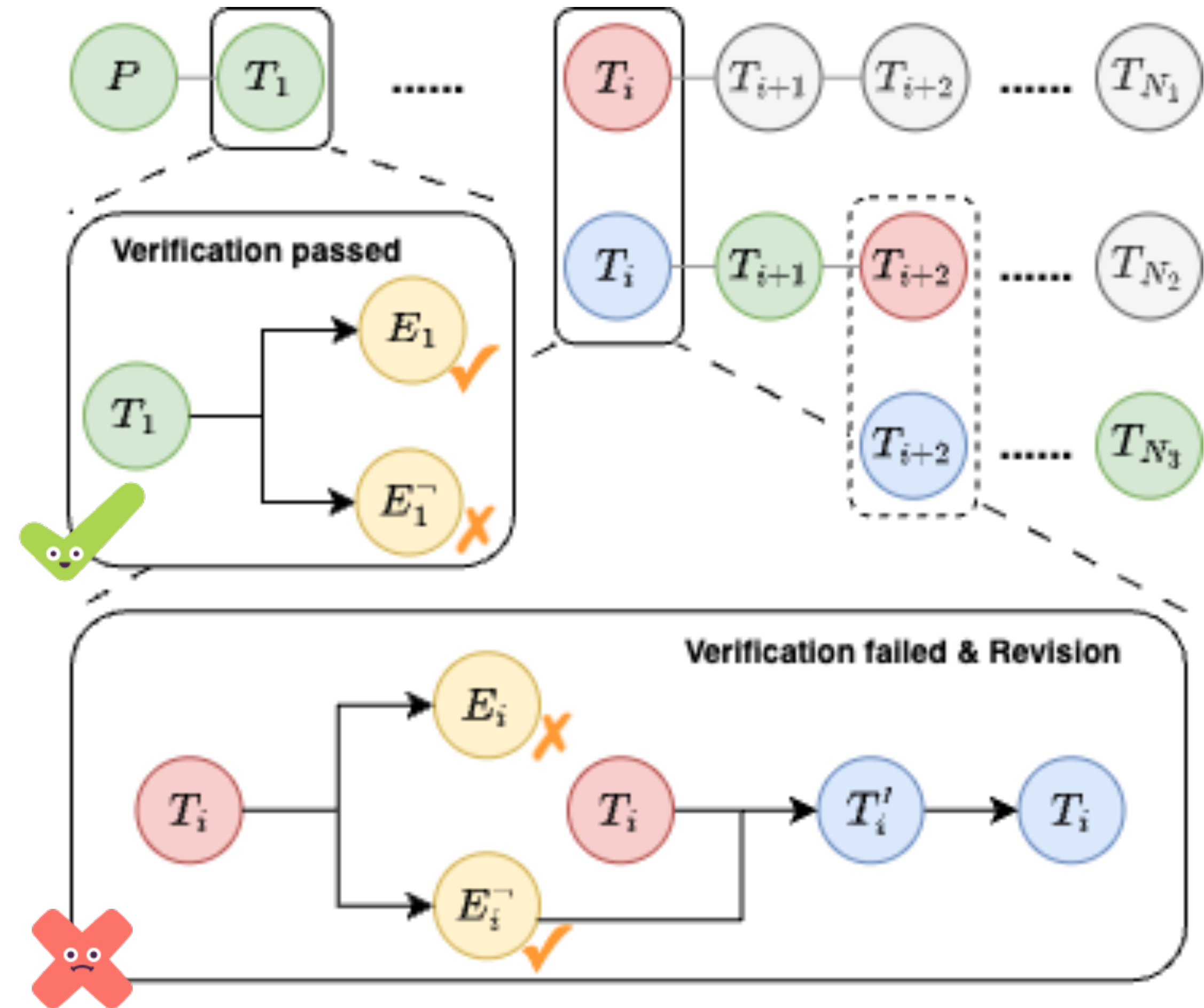
LoT: Logical Chain of Thoughts

Practical Implementation

- E_i post-hoc explanation of T_i
 - “Step i is **true** because ___”
- E_i^{-1} post-hoc explanation of $\neg T_i$
 - “Step i is **false** because ___”



*compulsory
error-finding*



LoT: Logical Chain of Thoughts

Practical Implementation

- E_i post-hoc explanation of T_i
 - “Step i is **true** because ___”
- E_i^\neg post-hoc explanation of $\neg T_i$
 - “Step i is **false** because ___”



*compulsory
error-finding*

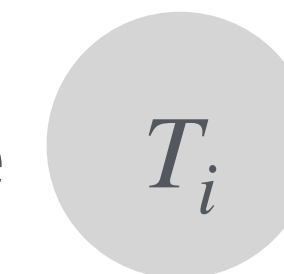
Generative self-verification



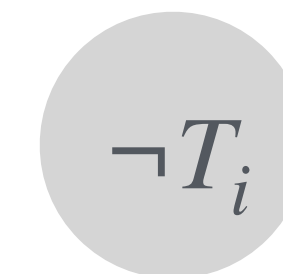
?



Discriminative preference



/



?

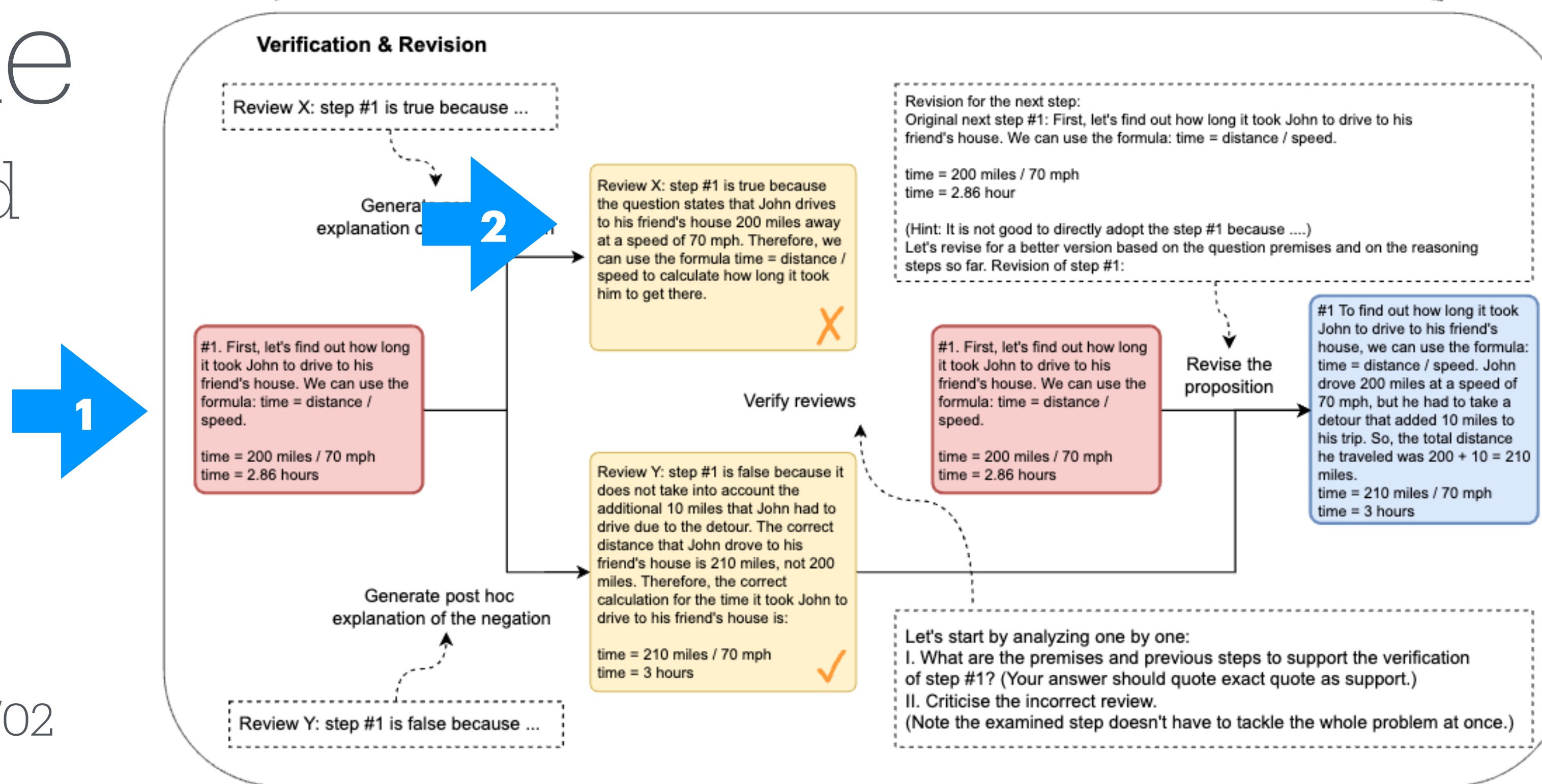
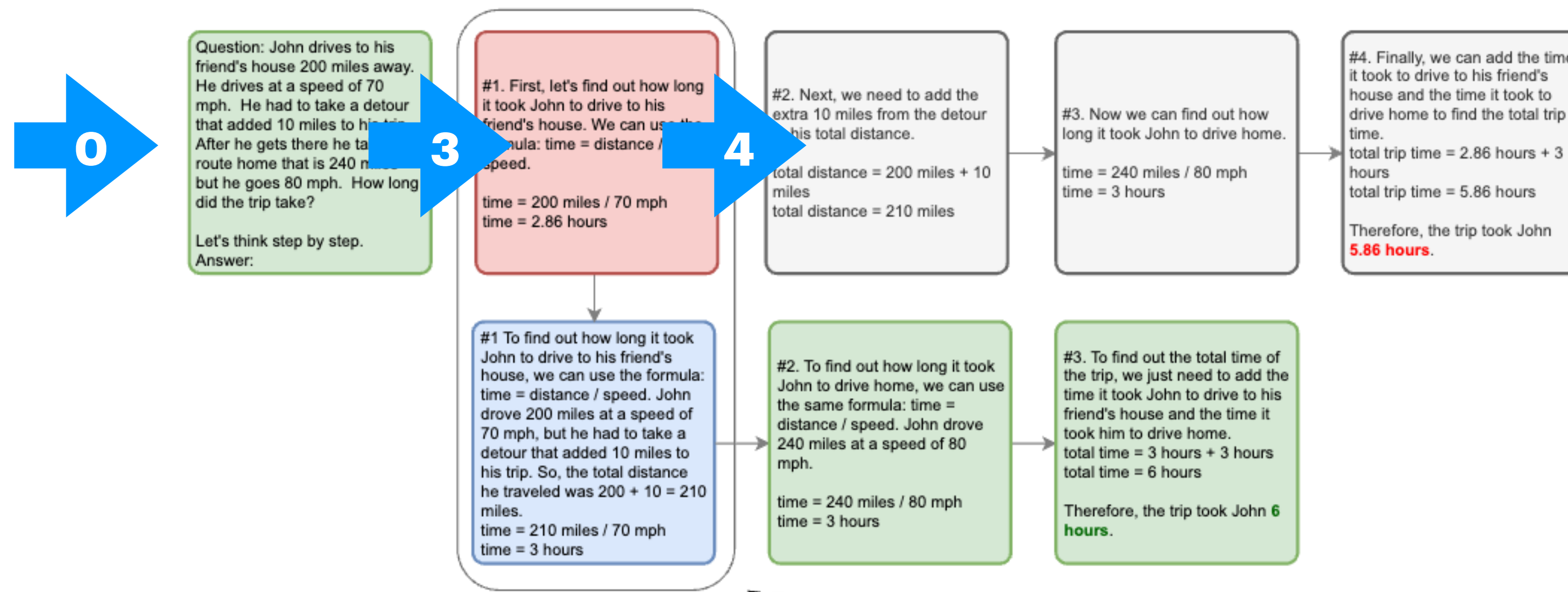
(E_i)

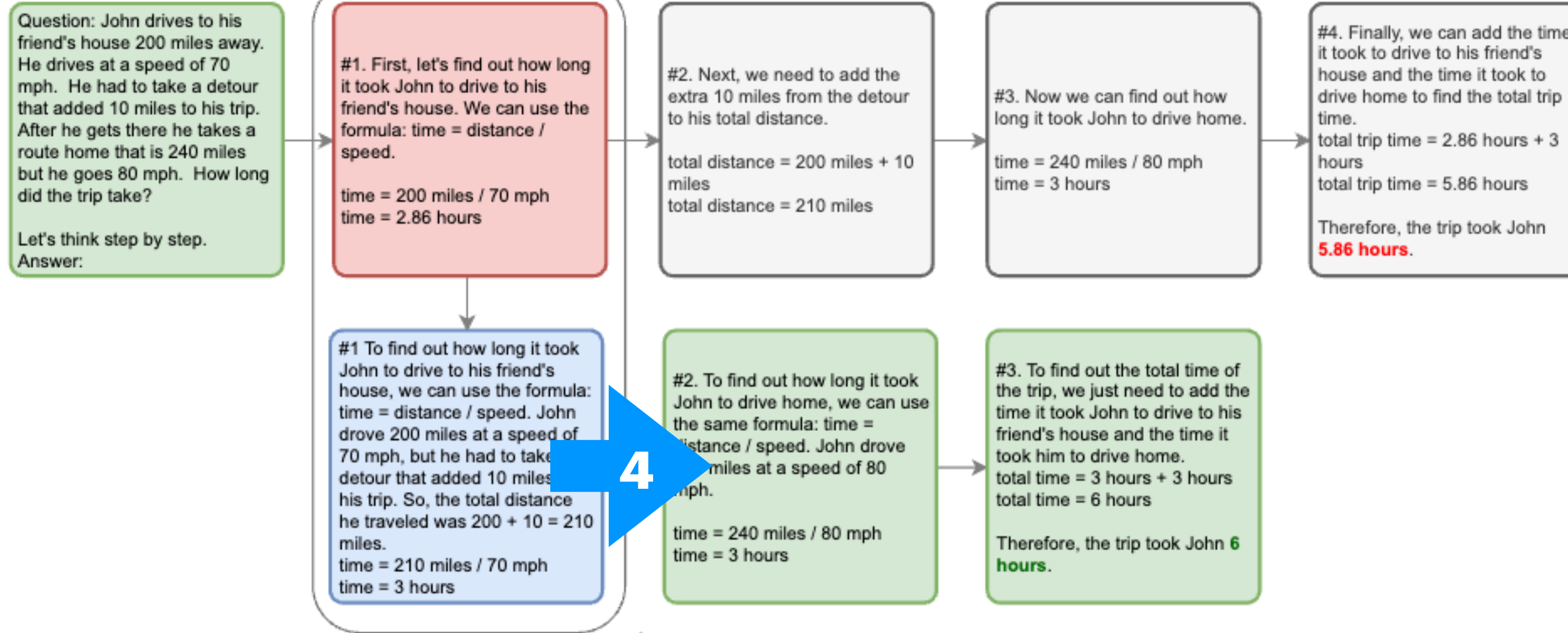
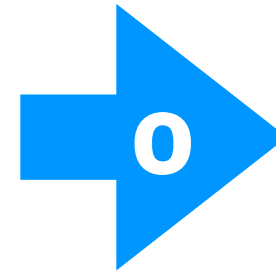
(E_i^\neg)

+ G-D gap [2]: promise to improve by discerning the quality of its generation

Example

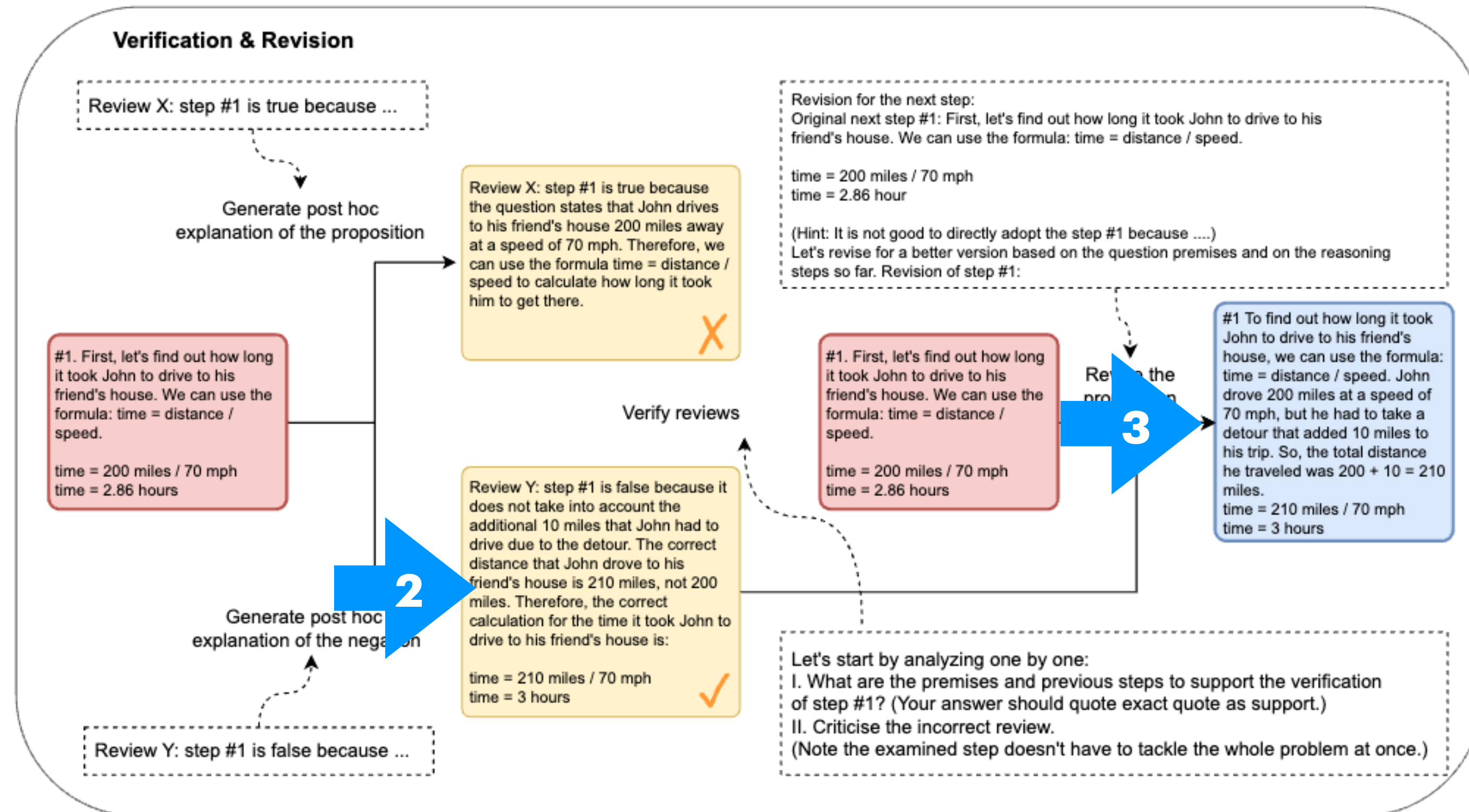
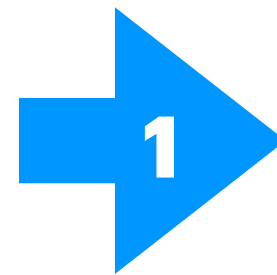
LoT explained





Example

LoT explained



Result

Accuracy

- On various domain tasks
- **Improved** performance

	LoT	GSM8K	AQuA	Date	SocialQA	Cau.Eff.	Objects	Letter	OddOut
Vicuna-7b	✗	17.52	21.65	7.24	37.00	52.94	34.00	0.00	25.58
	✓	17.68	20.47	7.24	36.50	52.94	35.00	0.00	25.58
		(+0.16)	(-1.18)	(0.00)	(-0.50)	(0.00)	(+1.00)	(0.00)	(0.00)
Vicuna-13b	✗	33.79	22.05	32.31	41.00	68.75	31.00	2.00	29.07
	✓	37.56	23.62	33.15	48.50	68.75	31.50	4.00	45.35
		(+3.77)	(+1.57)	(+0.84)	(+7.50)	(0.00)	(+0.50)	(+2.00)	(+16.28)
Vicuna-33b	✗	40.33	26.38	15.70	37.50	52.94	32.00	14.67	40.70
	✓	40.49	29.53	20.35	47.50	68.75	34.50	14.00	43.02
		(+0.16)	(+3.15)	(+4.65)	(+10.00)	(+15.81)	(+2.50)	(-0.67)	(+2.32)
GPT-3.5-turbo	✗	78.75	57.09	51.26	72.00	92.16	60.75	67.33	81.40
	✓	80.15	60.63	52.37	72.00	92.16	58.25	67.33	81.40
		(+1.40)	(+3.54)	(+1.11)	(0.00)	(0.00)	(-2.50)	(0.00)	(0.00)
GPT-4	✗	94.29	71.56	83.09	77.50	100.00	100.00	92.61	95.35
	✓	95.71	74.31	85.16	77.50	100.00	100.00	93.14	96.51
		(+1.42)	(+2.75)	(+2.07)	(0.00)	(0.00)	(0.00)	(+0.53)	(+1.16)

Table 1: We evaluate the accuracy of our methods and compare them against baseline approaches using various models and datasets, with computation based on ground truth annotations. The percentage difference of CoT (Kojima et al., 2022) without (✗) and with (✓) LoT enhancement using different LLMs is shown below each cell group (green if positive, red if negative). CoT generally gains better performance when being enhanced by LoT. Larger models, such as GPT-4, exhibit strong and robust self-correction ability.

Result

Revision frequency

- On various domain tasks
- **Improved** performance
- Revisions



Model scale

weak to verify

hesitate

confident basis

Revision	GSM8K	AQuA	Date	SocialQA	Cau.Eff.	Objects	Letter	OddOut
Vicuna-7b	2%	4%	2%	1%	2%	0%	3%	0%
Vicuna-13b	7%	10%	5%	5%	0%	7%	2%	0%
Vicuna-33b	2%	9%	8%	7%	6%	9%	1%	7%
GPT-3.5-turbo	16%	28%	32%	5%	20%	9%	4%	16%
GPT-4	3%	20%	7%	2%	0%	1%	0%	8%

Table 2: The average step-wise revision frequency is presented as a percentage, reflecting how often a reasoning step is revised by LoT.

Result

Revision frequency

- On various domain tasks
- **Improved** performance
- Revisions



Model scale

- No major increased steps

weak to verify
hesitate
confident basis

Revision	GSM8K	AQuA	Date	SocialQA	Cau.Eff.	Objects	Letter	OddOut
Vicuna-7b	2%	4%	2%	1%	2%	0%	3%	0%
Vicuna-13b	7%	10%	5%	5%	0%	7%	2%	0%
Vicuna-33b	2%	9%	8%	7%	6%	9%	1%	7%
GPT-3.5-turbo	16%	28%	32%	5%	20%	9%	4%	16%
GPT-4	3%	20%	7%	2%	0%	1%	0%	8%

Table 2: The average step-wise revision frequency is presented as a percentage, reflecting how often a reasoning step is revised by LoT.

	LoT	GSM8K	AQuA	Date	SocialQA	Cau.Eff.	Objects	Letter	OddOut
Vicuna-7b	✗	1.22	1.16	1.34	1.09	1.00	2.54	3.46	1.00
	✓	1.27	1.21	1.35	1.10	1.02	2.54	3.49	1.00
Vicuna-13b	✗	2.81	2.89	5.06	2.69	1.00	2.93	1.66	1.00
	✓	2.74	2.87	5.05	2.71	1.00	2.96	1.69	1.00
Vicuna-33b	✗	1.94	1.99	2.31	3.26	1.00	3.26	1.20	1.70
	✓	1.94	1.91	2.33	3.13	1.06	3.23	1.21	1.64
GPT-3.5-turbo	✗	4.17	6.83	3.66	2.50	1.73	3.02	4.84	1.57
	✓	4.08	6.24	3.56	2.51	1.92	3.05	4.81	1.70
GPT-4	✗	3.42	4.22	2.71	2.33	1.00	3.00	4.05	1.00
	✓	3.41	4.39	2.72	2.33	1.00	3.01	4.05	1.08

Table 3: The average number of resultant reasoning steps without (✗) and with (✓) LoT applied.

Result

Ablation for self-check

- On various domain tasks
- **Improved** performance
- Revisions



Model scale

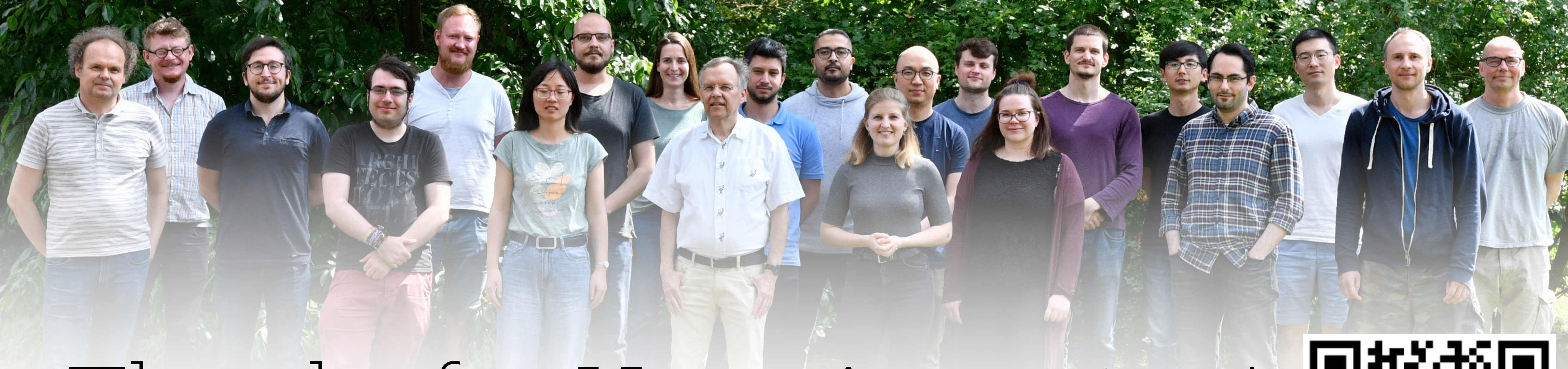
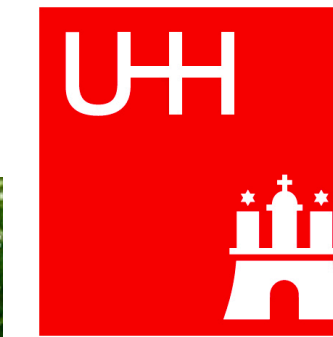
- No major increased steps
- Better self-check with post-hoc explanations

Xufeng Zhao, 2024/05/02

Method	GSM8K	AQuA	Date
CoT	78.75	57.09	51.26
Self-Check	76.15	56.19	51.57
Cmps-LoT	77.67	57.48	52.37
LoT	80.15	60.63	52.37

Table 4: Zero-shot accuracy results (in %) in comparison of LoT (i.e. Adpt-) and its ablated variants. The underlying LLM is GPT-3.5-turbo.

LREC-COLING 2024



Thanks for Your Attention!

All questions & discussions & collaborations are welcome . . .

Xufeng Zhao, 2024/05/02



Appendix+

Pseudo codes

LoT variants

- Difference

Algorithm 2 Adpt-LoT Reasoning

Require: P , LLM

Initialize $\mathcal{T} \leftarrow \{P\}$

$T_1, T_2, \dots, T_N \leftarrow \text{LLM}(\mathcal{T})$ \triangleright Vanilla CoT

$i \leftarrow 1$

while $i \leq N$ **do**

$E_i^- \leftarrow \text{post hoc LLM}(E|\neg T_i; \mathcal{T})$

$E_i \leftarrow \text{post hoc LLM}(E|T_i; \mathcal{T})$

$\hat{E} \leftarrow \text{LLM}(E_i; E_i^- | \mathcal{T})$ \triangleright Adopt

if \hat{E} is E_i^- **then**

$T'_i \leftarrow \text{LLM}(T|\mathcal{T}; T_i; E_i^-)$ \triangleright Revise

$T_i \leftarrow T'_i$

$\{T_{>i}\}_{N'} \leftarrow \text{LLM}(\mathcal{T} \cup T_i)$ \triangleright Adapt

$N \leftarrow N'$

end if

$\mathcal{T} \leftarrow \mathcal{T} \cup T_i$ \triangleright Update

$i \leftarrow i + 1$

end while

return \mathcal{T}

Algorithm 3 Cmps-LoT Reasoning

Require: P , LLM

Initialize $\mathcal{T} \leftarrow \{P\}$

$T_1, T_2, \dots, T_N \leftarrow \text{LLM}(\mathcal{T})$ \triangleright Vanilla CoT

$i \leftarrow 1$

while $i \leq N$ **do**

$E_i^- \leftarrow \text{post hoc LLM}(E_i|\neg T_i; \mathcal{T})$

$C \leftarrow \text{LLM}(E_i^- | \mathcal{T})$ \triangleright Compose

if C is *False* **then**

$T'_i \leftarrow \text{LLM}(T|\mathcal{T}; T_i; E_i^-)$ \triangleright Revise

$T_i \leftarrow T'_i$

$\{T_{>i}\}_{N'} \leftarrow \text{LLM}(\mathcal{T} \cup T_i)$ \triangleright Adapt

$N \leftarrow N'$

end if

$\mathcal{T} \leftarrow \mathcal{T} \cup T_i$ \triangleright Update

$i \leftarrow i + 1$

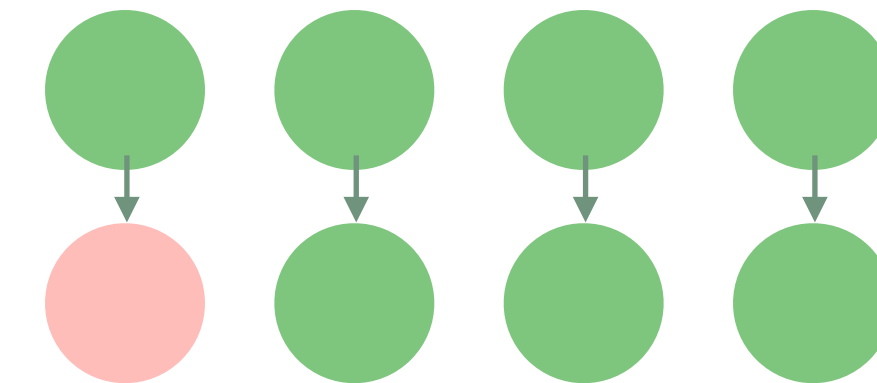
end while

return \mathcal{T}

Motivation

Chain-of-Thought Prompting (CoT)

An aeroplane flies along the four sides of a square at the speeds of 200, 400, 600 and 800km/hr. Find the average speed of the plane around the field? Options: OptA) 384 OptB) 562 OptC) 458 OptD) 156 OptE) 452
Let's think step by step.



Majority Voting

e.g. [1] Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. "Chain-of-thought prompting elicits reasoning in large language models." *Advances in Neural Information Processing Systems* 35 (2022): 24824-24837.