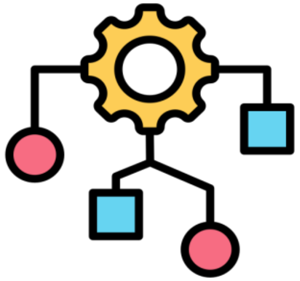
EMNLP 2023

LINC: A Neuro-symbolic Approach for Logical Reasoning by Combining Language Models with First-Order Logic Provers

Olausson et. al. (MIT CSAIL, MIT BCS)

+



=

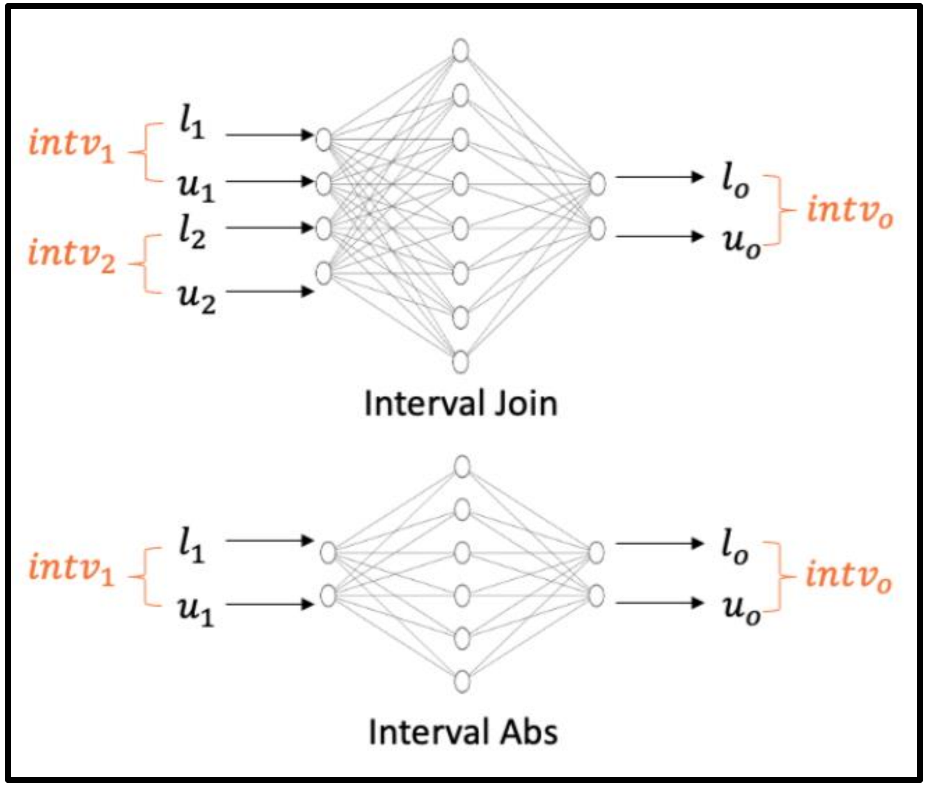
Presented by: Shaurya Gomber

About Me

- 2nd yr PhD student (Advisor: Prof. Gagandeep Singh).
- Learning (and optimization) based techniques for efficient and precise abstract interpretation.

About Me

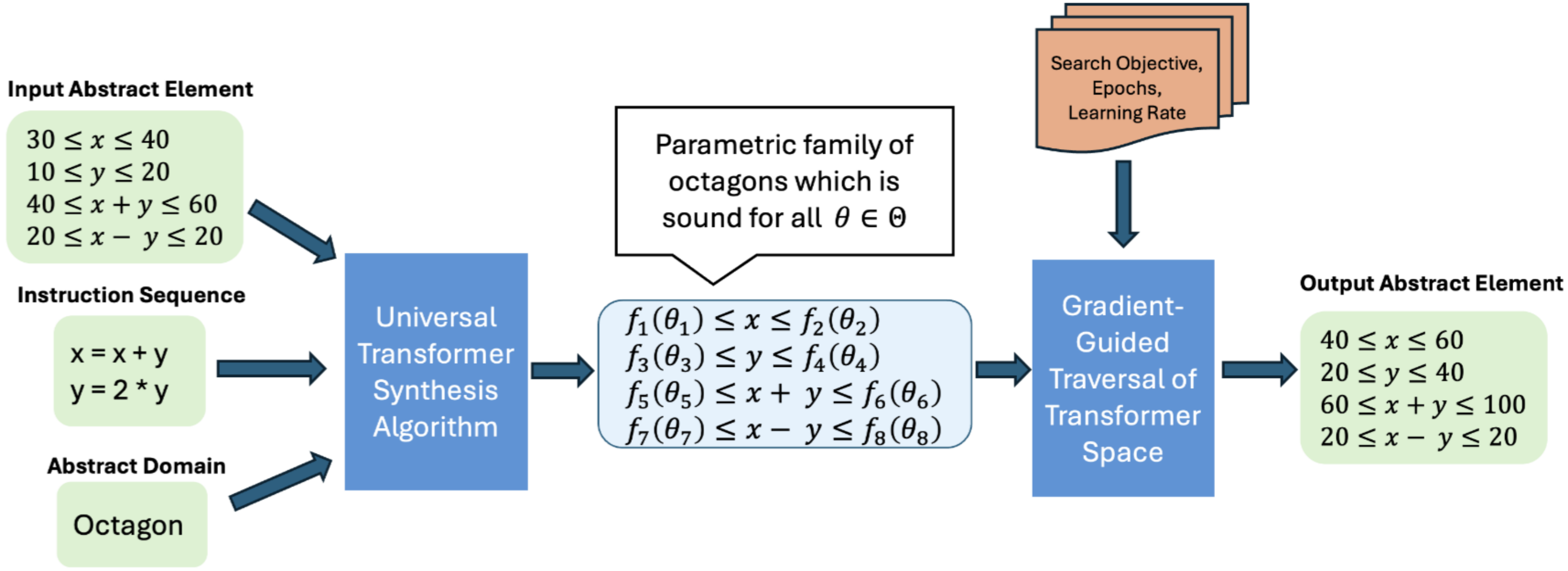
- 2nd yr PhD student (Advisor: Prof. Gagandeep Singh).
- Learning (and optimization) based techniques for efficient and precise abstract interpretation.
- MS Thesis: **Neural Abstract Interpretation** (VerifAI Workshop @ ICLR 2025)



Supervised and Unsupervised learning formulations to train "neural" abstract transformers!

About Me

- 2nd yr PhD student (Advisor: Prof. Gagandeep Singh).
- Learning (and optimization) based techniques for efficient and precise abstract interpretation.
- MS Thesis: Neural Abstract Interpretation (VerifAI Workshop @ ICLR 2025)
- Current: **Universal Synthesis of Differentiably Tunable Numerical Abstract Transformers** (submitted to POPL 2026)



Contents

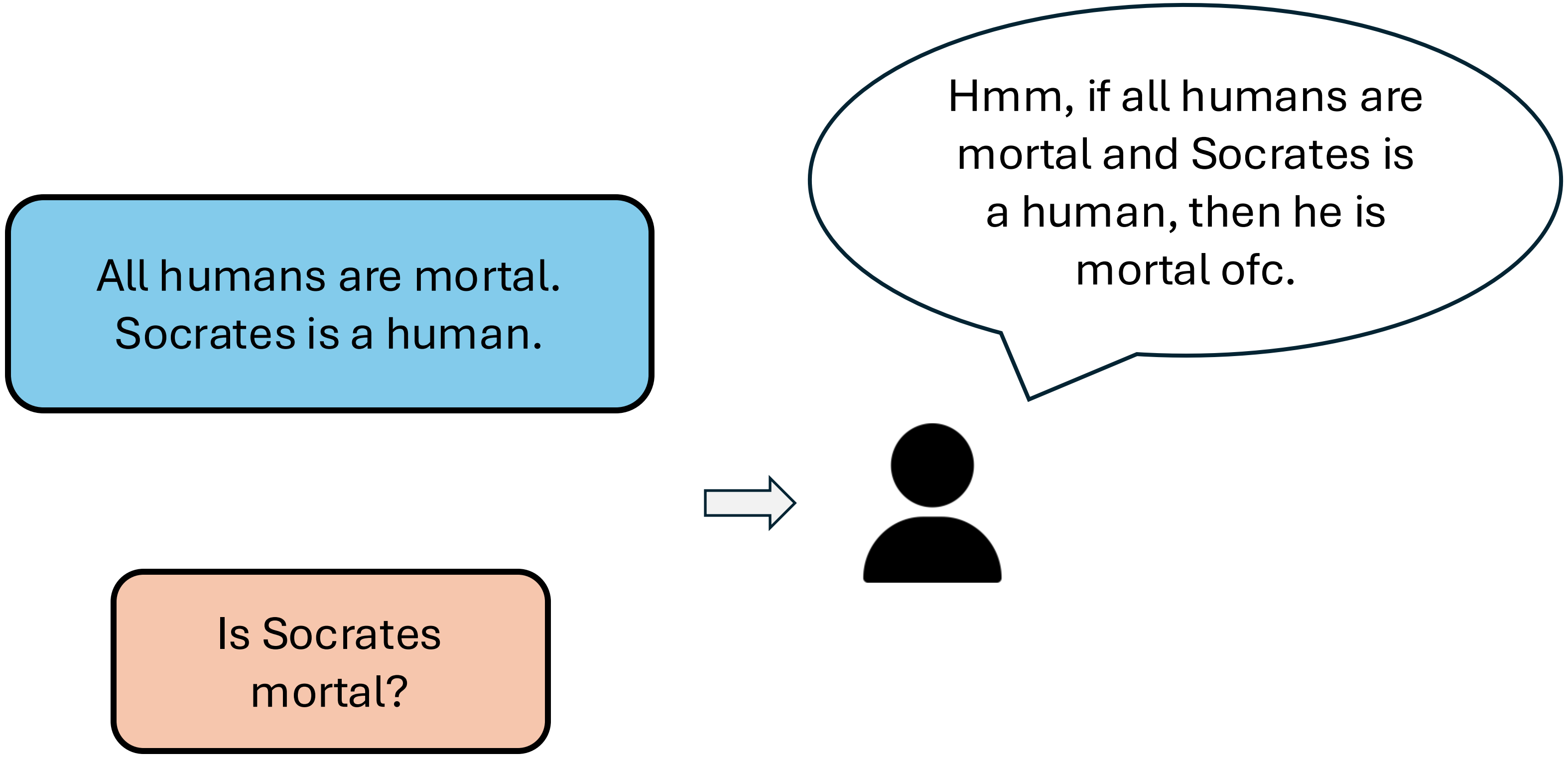
Contents

Logical Reasoning

Structured rules → analyze facts (premises) → answer questions (derive conclusions)!

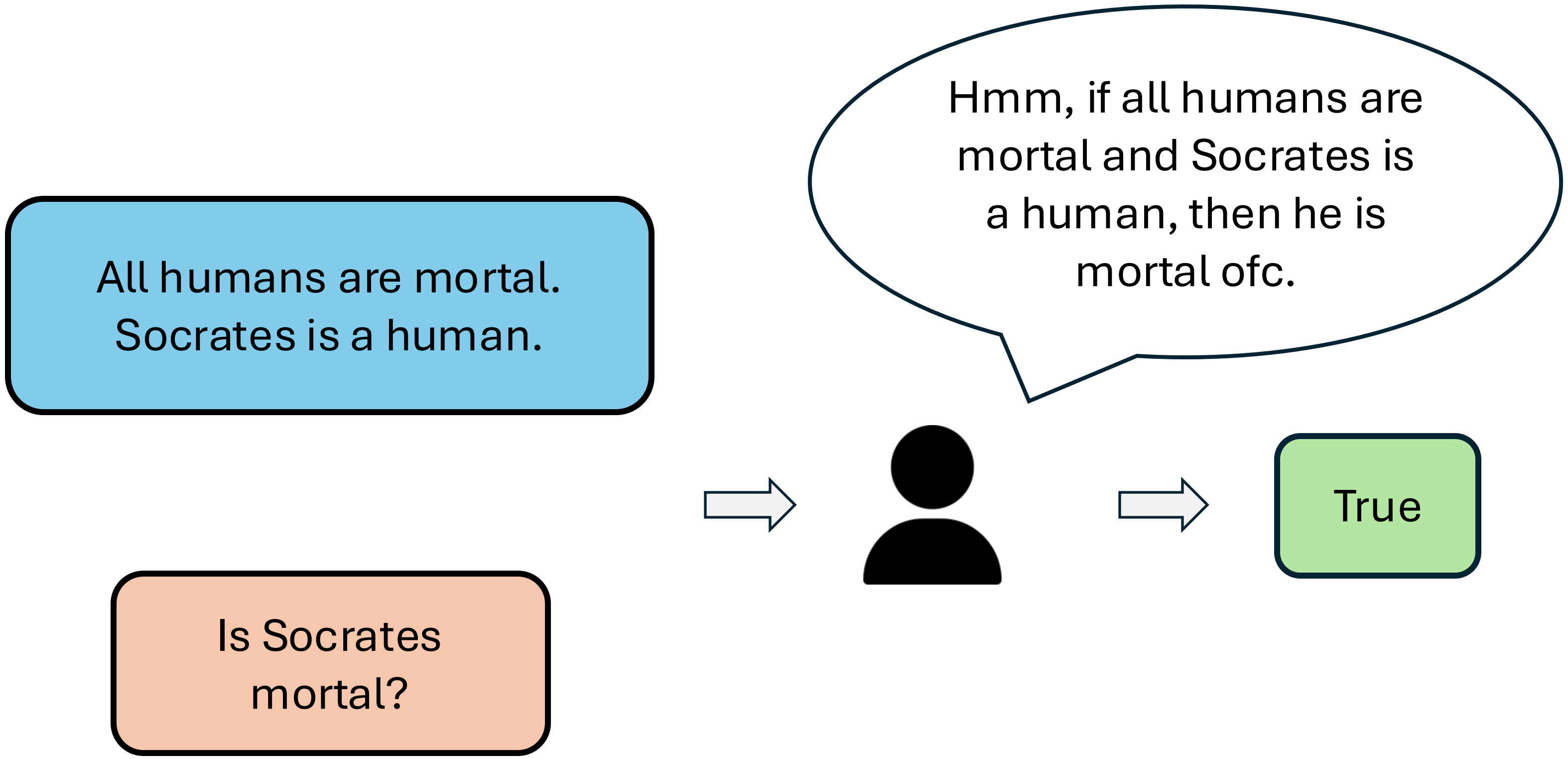
Logical Reasoning

Structured rules → analyze facts (premises) → answer questions (derive conclusions)!



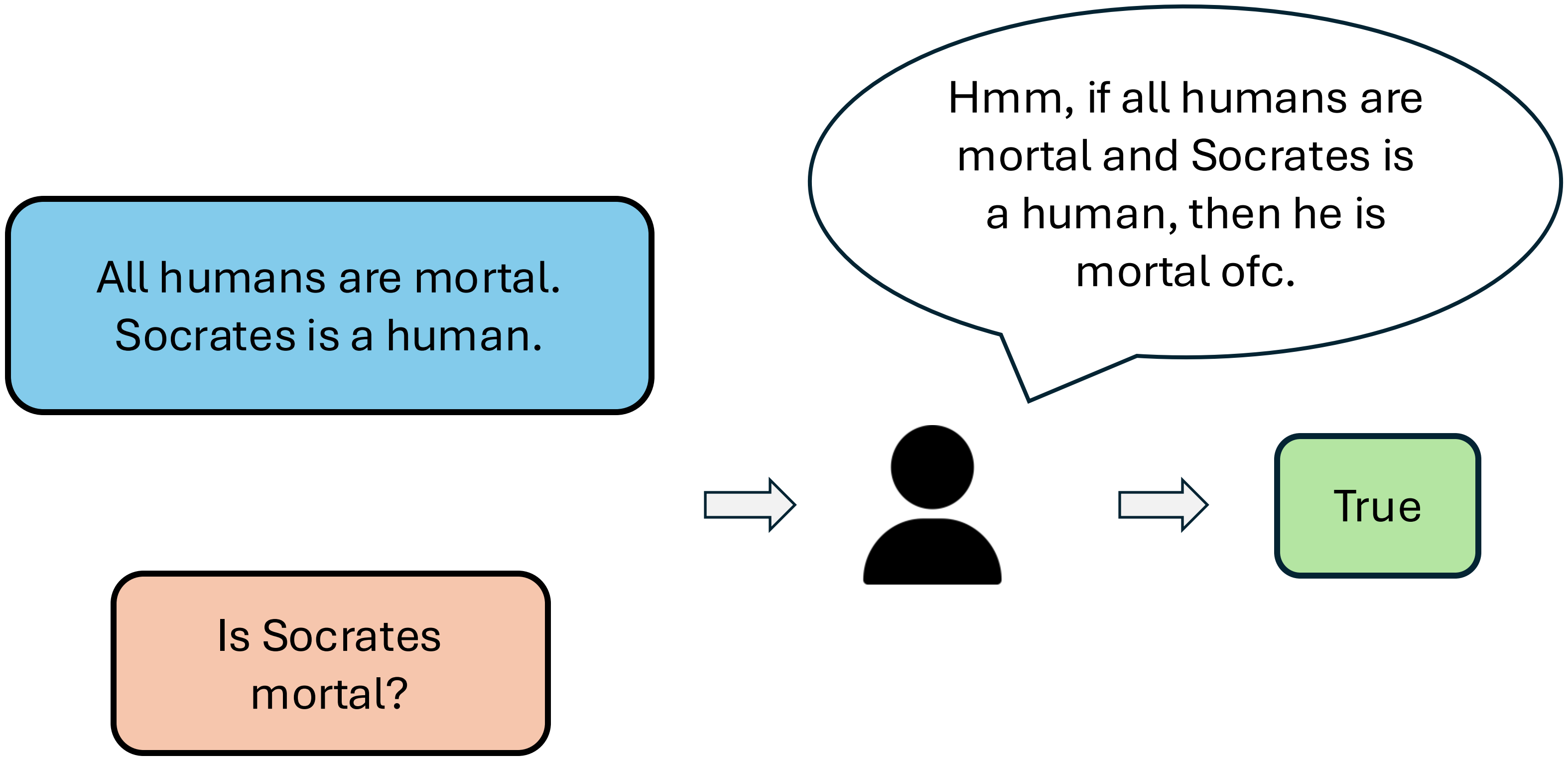
Logical Reasoning

Structured rules → analyze facts (premises) → answer questions (derive conclusions)!



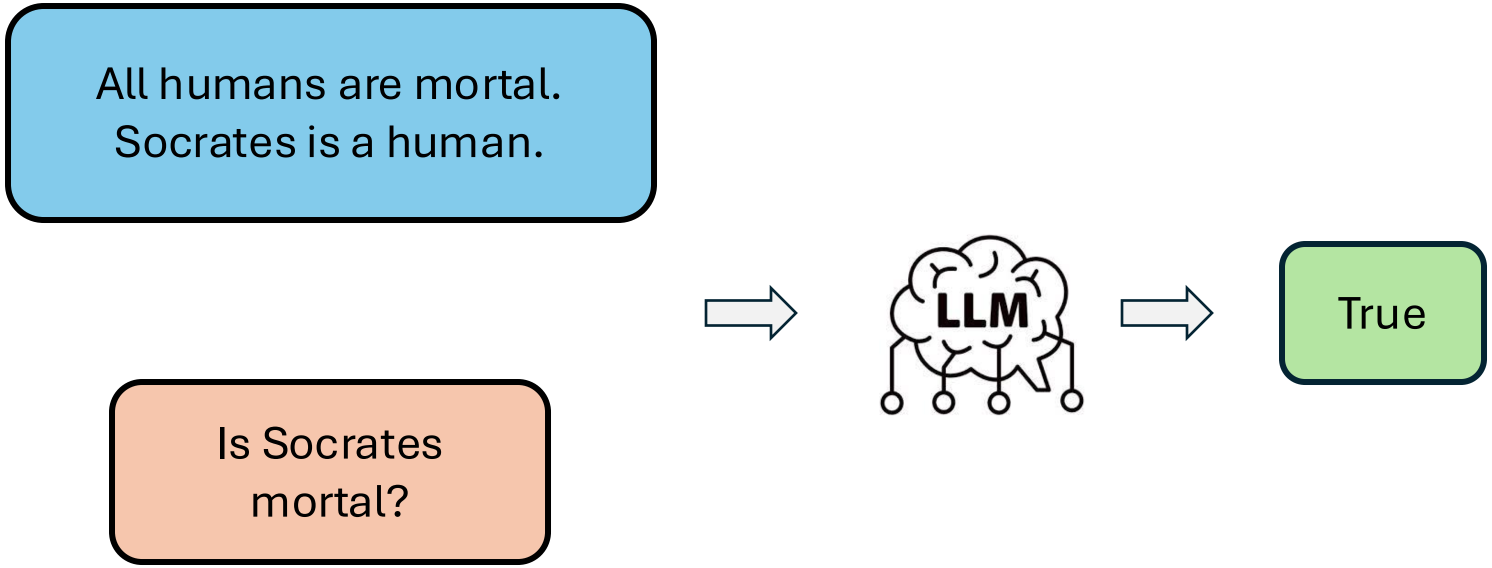
Logical Reasoning

Structured rules → analyze facts (premises) → answer questions (derive conclusions)!



If (A ⇒ B) and A holds, then B holds **(Modus Ponens)**

Logical Reasoning in LLMs



Conclusion: Socrates is mortal.

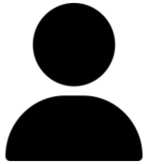
This follows logically and necessarily from the two premises using **modus ponens** (if $A \rightarrow B$ and A, then B).

Logical Reasoning in LLMs

Those who enjoy Poetry write short verses.
Those who enjoy Novels write long stories.
Shaurya writes both short verses and long stories.

Does Shaurya enjoy poetry?

Though Shaurya writes short verses, he may or may not enjoy Poetry. Hard to say.



Uncertain

Logical Reasoning in LLMs

Those who enjoy Poetry write short verses.
Those who enjoy Novels write long stories.
Shaurya writes both short verses and long stories.

Does Shaurya enjoy poetry?

True

Logical Reasoning in LLMs

From the premises:

1. Those who enjoy Poetry → write short verses.
2. Those who enjoy Novels → write long stories.
3. Shaurya writes both short verses and long stories.

By premise 1, writing short verses implies Shaurya enjoys Poetry.

By premise 2, writing long stories implies Shaurya enjoys Novels.

So the truth value of *"Does Shaurya enjoy Poetry?"* is **True**.

(In fact, Shaurya enjoys **both** Poetry and Novels.)

GPT-5's Response

Logical Reasoning in LLMs

From the premises:

1. Those who enjoy Poetry → write short verses.
2. Those who enjoy Novels → write long stories.
3. Shaurya writes both short verses and long stories.

By premise 1, writing short verses implies Shaurya enjoys Poetry.

By premise 2, writing long stories implies Shaurya enjoys Novels.

So the truth value of *"Does Shaurya enjoy Poetry?"* is **True**.

(In fact, Shaurya enjoys **both** Poetry and Novels.)

GPT-5's Response

If $(A \Rightarrow B)$

Logical Reasoning in LLMs

From the premises:

1. Those who enjoy Poetry → write short verses.
2. Those who enjoy Novels → write long stories.
3. Shaurya writes both short verses and long stories.

By premise 1, writing short verses implies Shaurya enjoys Poetry.

By premise 2, writing long stories implies Shaurya enjoys Novels.

So the truth value of *"Does Shaurya enjoy Poetry?"* is **True**.

(In fact, Shaurya enjoys **both** Poetry and Novels.)

GPT-5's Response

If $(A \Rightarrow B)$ and B holds,

Logical Reasoning in LLMs

From the premises:

1. Those who enjoy Poetry → write short verses.
2. Those who enjoy Novels → write long stories.
3. Shaurya writes both short verses and long stories.

By premise 1, writing short verses implies Shaurya enjoys Poetry.

By premise 2, writing long stories implies Shaurya enjoys Novels.

So the truth value of *"Does Shaurya enjoy Poetry?"* is **True**.

(In fact, Shaurya enjoys **both** Poetry and Novels.)

GPT-5's Response

If $(A \Rightarrow B)$ and B holds, then A holds.

Logical Reasoning in LLMs

From the premises:

1. Those who enjoy Poetry → write short verses.
2. Those who enjoy Novels → write long stories.
3. Shaurya writes both short verses and long stories.

By premise 1 writing short verses implies Shaurya enjoys Poetry.

By premise 2, writing long stories implies Shaurya enjoys Novels.

So the truth value of *"Does Shaurya enjoy Poetry?"* is **True**.

(In fact, Shaurya enjoys **both** Poetry and Novels.)

GPT-5's Response

If (A ⇒ B) and B holds, then A holds.

Modus GPT!

Logical Reasoning in LLMs

From the premises:

Affirming the consequent

Article   Talk   Read   Edit   View history   Tools

24 languages

From Wikipedia, the free encyclopedia

"False conversion" redirects here. For the Islamic doctrine, see Taqiya.

In propositional logic, **affirming the consequent** (also known as **converse error**, **fallacy of the converse**, or **confusion of necessity and sufficiency**) is a formal fallacy (or an invalid form of argument) that is committed when, in the context of an indicative conditional statement, it is stated that because the consequent is true, therefore the antecedent is true. It takes on the following form:

> If *P*, then *Q*.
>
> *Q*.
>
> Therefore, *P*.

Problem: LLMs are bad at Logical Reasoning

- **Unreliable**: fail on out-of-domain tasks [1]

Prove: Max is a gorpus.

Predicted answer: Max is a tumpus or a rompus or a lempus. Max is a tumpus. Tumpuses are wumpuses. Max is a wumpus. Rompuses are gorpuses. Max is a gorpus. Max is a gorpus.

Expected answer: Assume Max is a tumpus. Tumpuses are gorpuses. Max is a gorpus. Assume Max is a rompus. Rompuses are gorpuses. Max is a gorpus. Assume Max is a lempus. Lempuses are gorpuses. Max is a gorpus. Since Max is a tumpus or a rompus or a lempus, Max is a gorpus.

FIGURE 5: Example of an incorrect proof generated by GPT-3.5 on an out-of-demonstration disjunction elimination example. The premises (axioms) are given in blue, and invalid steps are given in red. For the full example, see Figure 14 in the Appendix.

1. Testing the General Deductive Reasoning Capacity of Large Language Models Using OOD Examples. Saparov et. Al (NeurIPS 2023)

LLMs are bad at Logical Reasoning

- **Unreliable:** fail on out-of-domain tasks [1], have trouble understanding negation [2]

Prove: Max is a gorpus.

Predicted answer: Max is a tumpus or a ron
Tumpuses are wumpuses. Max is a wumpus. R
Max is a gorpus.

Expected answer: Assume Max is a tumpus.
Assume Max is a rompus. Rompuses are gorpu
Assume Max is a lempus. Lempuses are gorpu
Since Max is a tumpus or a rompus or a lemp

IGURE 5: Example of an incorrect proof generated
limination example. The premises (axioms) are given i
xample, see Figure 14 in the Appendix.

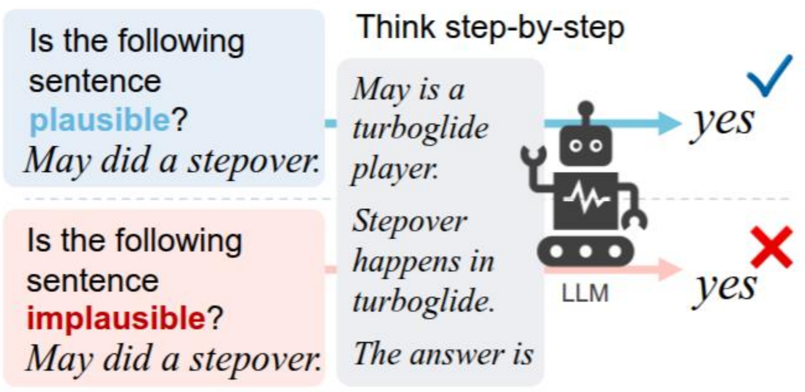


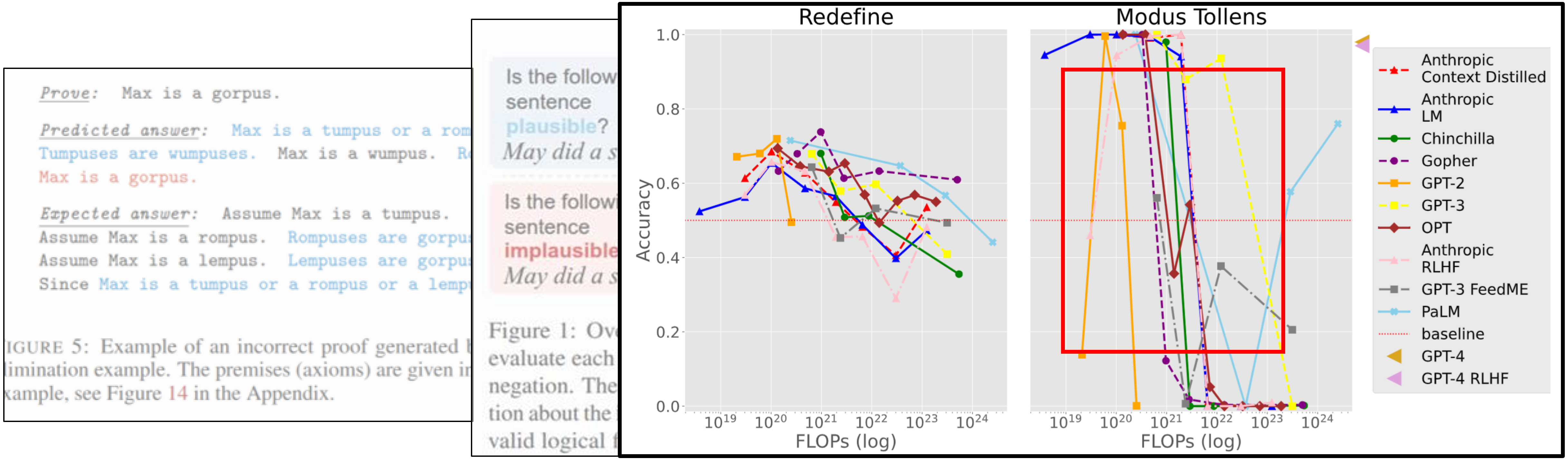
Figure 1: Overview of our experiments conducted to evaluate each model's reasoning ability against lexical negation. The model must answer *no* to the latter question about the **im**plausibility of the sentence based on the valid logical flow. Here, to evaluate the robust logical

1. Testing the General Deductive Reasoning Capacity of Large Language Models Using OOD Examples. Saparov et. Al (NeurIPS 2023)
2. Assessing Step-by-Step Reasoning against Lexical Negation: A Case Study on Syllogism. Ye et. Al (EMNLP 2023)

LLMs are bad at Logical Reasoning

- **Unreliable**: fail on out-of-domain tasks [1], have trouble understanding negation [2]
- **Scaling fails**: bigger models don't improve core logic (e.g., Modus Tollens) [3]

Modus Tollens:
If (A ⇒ B) and B does not hold, then A does not hold.

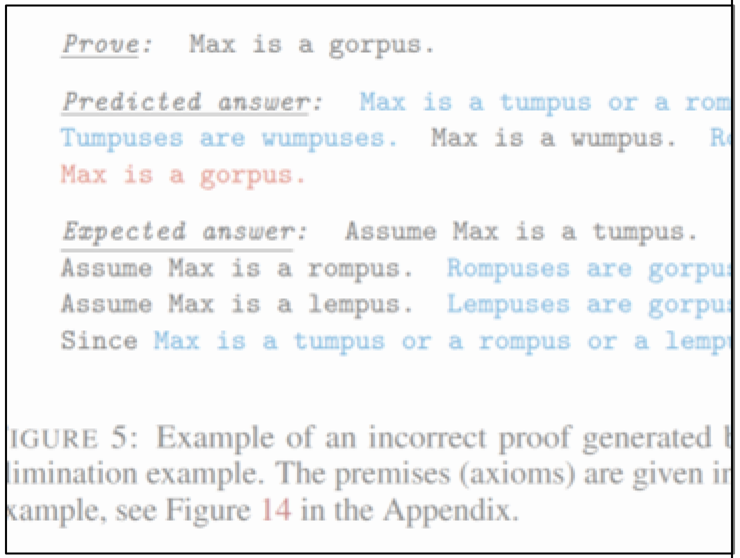


1. Testing the General Deductive Reasoning Capacity of Large Language Models Using OOD Examples. Saparov et. Al (NeurIPS 2023)
2. Assessing Step-by-Step Reasoning against Lexical Negation: A Case Study on Syllogism. Ye et. Al (EMNLP 2023)
3. Inverse Scaling: When Bigger Isn't Better. McKenzie et. Al (TMLR 2023)

LLMs are bad at Logical Reasoning

- **Unreliable**: fail on out-of-domain tasks [1], have trouble understanding negation [2]
- **Scaling fails**: bigger models don't improve core logic (e.g., Modus Tollens) [3]
- Models lean on data patterns, not reasoning skills.

Modus Tollens:
If (A ⇒ B) and B does not hold, then A does not hold.



IGURE 5: Example of an incorrect proof generated l
limination example. The premises (axioms) are given ir
xample, see Figure 14 in the Appendix.

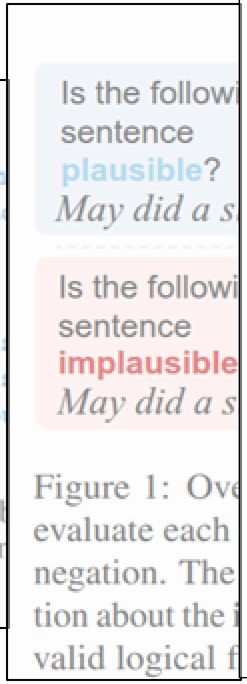
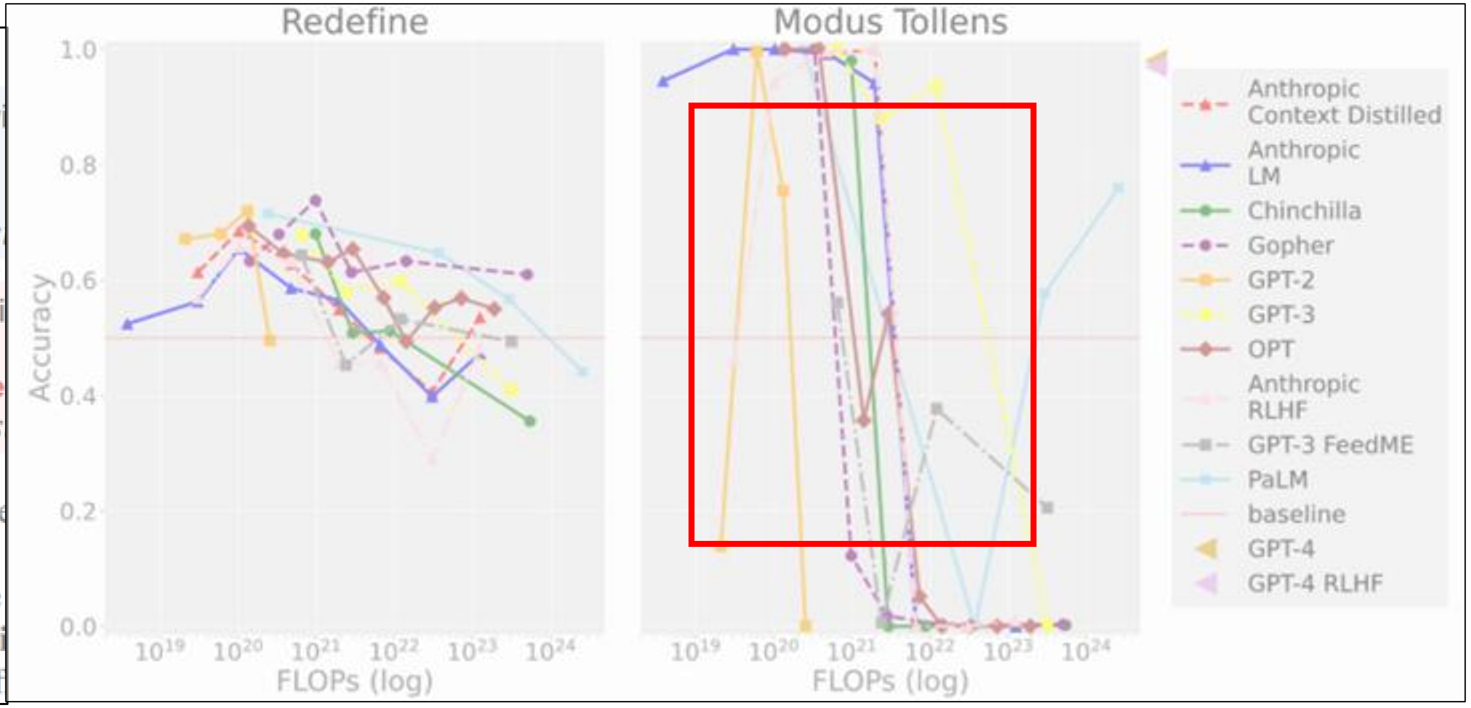


Figure 1: Ove
evaluate each
negation. The
tion about the i
valid logical f



1. Testing the General Deductive Reasoning Capacity of Large Language Models Using OOD Examples. Saparov et. Al (NeurIPS 2023)
2. Assessing Step-by-Step Reasoning against Lexical Negation: A Case Study on Syllogism. Ye et. Al (EMNLP 2023)
3. Inverse Scaling: When Bigger Isn't Better. McKenzie et. Al (TMLR 2023)

Logical Reasoning is essential for AI!

- **Truthfulness in AI systems**: ensures chatbot answers follow from retrieved facts.
- **Advancing science & maths**: enables theorem proving and knowledge discovery.
- **Better education tools**: tutoring systems that teach clarity and rigor.

Contents

Contents

How to make LLMs reason well?

Specialized tools exist for logical reasoning!

First-order Logic

- A formal system for writing **logical statements** about the world.

- **Predicates** state properties or relations of objects.
  - Lazy(X) holds means X is lazy (e.g. Lazy(Shaurya))
  - Loves(X, Y) means X loves Y (e.g. Loves(Shaurya, Food))

- Has **∀ (for all)** and **∃ (there exists)** to talk about general rules.
  - $\forall X\ (\text{PhDStudent}(X) \Rightarrow \text{NeedToTakeQuals}(X))$ means "Every X who is a PhD student has to take quals."

First-order Logic

NL Description	FOL Conversion
Socrates is a human.	Human(Socrates)
All humans are mortal.	$\forall x\ (Human(x) \Rightarrow Mortal(x))$
Those who enjoy Poetry write short verses.	$\forall x\ (EnjoyPoetry(x) \Rightarrow WriteShortVerses(x))$
Shaurya writes both short verses and long stories.	$WriteShortVerses(Shaurya) \land WriteLongStories(Shaurya)$

First-order Logic Prover

- Takes FOL premises and applies sound deduction to reason correctly.

First-order Logic Prover

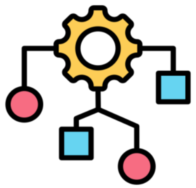
- Takes FOL premises and applies sound deduction to reason correctly.
- Four possible cases:

First-order Logic Prover

- Takes FOL premises and applies sound deduction to reason correctly.
- Four possible cases: (1) True

A $\forall x\,(\text{Human}(x) \Rightarrow \text{Mortal}(x))$
Human(Socrates)

C Mortal(Socrates)



True

Prover is able to prove **A** ⇒ **C** valid!

First-order Logic Prover

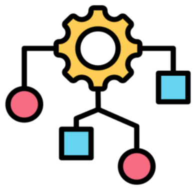
- Takes FOL premises and applies sound deduction to reason correctly.
- Four possible cases: (1) True (2) False

A

$\forall x\, (\text{Human}(x) \Rightarrow \text{Mortal}(x))$
Human(Socrates)

C

$\neg$Mortal(Socrates)



False

Prover is able to prove **A** $\Rightarrow \neg$**C** valid!

First-order Logic Prover

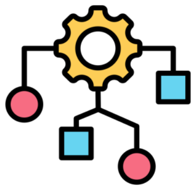
- Takes FOL premises and applies sound deduction to reason correctly.
- Four possible cases: (1) True (2) False (3) Uncertain

A

$\forall x\,(\text{Human}(x) \Rightarrow \text{Mortal}(x))$
Human(Socrates)

C

Mortal(Shaurya)



Uncertain

Prover is not able to prove **A** ⇒ **C** or **A** ⇒ ¬**C** valid!

First-order Logic Prover

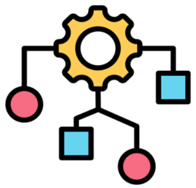
- Takes FOL premises and applies sound deduction to reason correctly.
- Four possible cases: (1) True (2) False (3) Uncertain (4) Error

A

∀x (Human(x) ⇒ Mortal(x)))
Human(Socrates)

C

Mortal(Socrates)



Error

FOL Syntax errors!

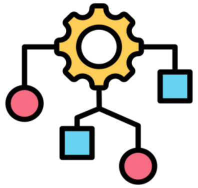
How to make LLMs reason well?

Let the cobbler stick to his last!

What are LLMs good at?

Understanding and parsing natural language,
but not guaranteed logical deduction.

What are provers good at?



Sound logical deductions given the premises in FOL.

Instead of asking LLMs to do everything,
ask them to formalize the NL premises into FOL
and offload the reasoning to a solver.

Methodology

All rectangles have four sides.
All four-sided things are shapes.

Are all rectangles shapes?

→

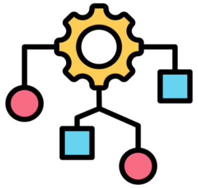
Semantic Parsing using LLMs

→

$\forall x\,(\text{Rectangle}(x) \Rightarrow \text{FourSides}(x))$
$\forall x\,(\text{FourSides}(x) \Rightarrow \text{IsShape}(x))$

$\forall x\,(\text{Rectangle}(x) \Rightarrow \text{IsShape}(x))$

→



Offloading to solver

→

True

Methodology

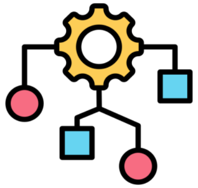
All rectangles have four sides. All four-sided things are shapes.

Are all rectangles shapes?

Semantic Parsing using LLMs

$\forall x\,(\text{Rectangle}(x) \Rightarrow \text{FourSides}(x))$
$\forall x\,(\text{FourSides}(x) \Rightarrow \text{IsShape}(x)))$

$\forall x\,(\text{Rectangle}(x) \Rightarrow \text{IsShape}(x))$



Offloading to solver

Error

Methodology

All rectangles have four sides.
All four-sided things are shapes.

Are all rectangles shapes?

→

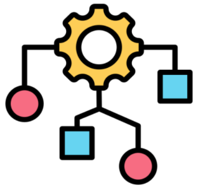
Semantic Parsing using LLMs

→

$\forall x\,(\text{Rectangle}(x) \Rightarrow \text{FourSides}(x))$
$\forall x\,(\text{FourSided}(x) \Rightarrow \text{IsShape}(x))$

$\forall x\,(\text{Rectangle}(x) \Rightarrow \text{IsShape}(x))$

→

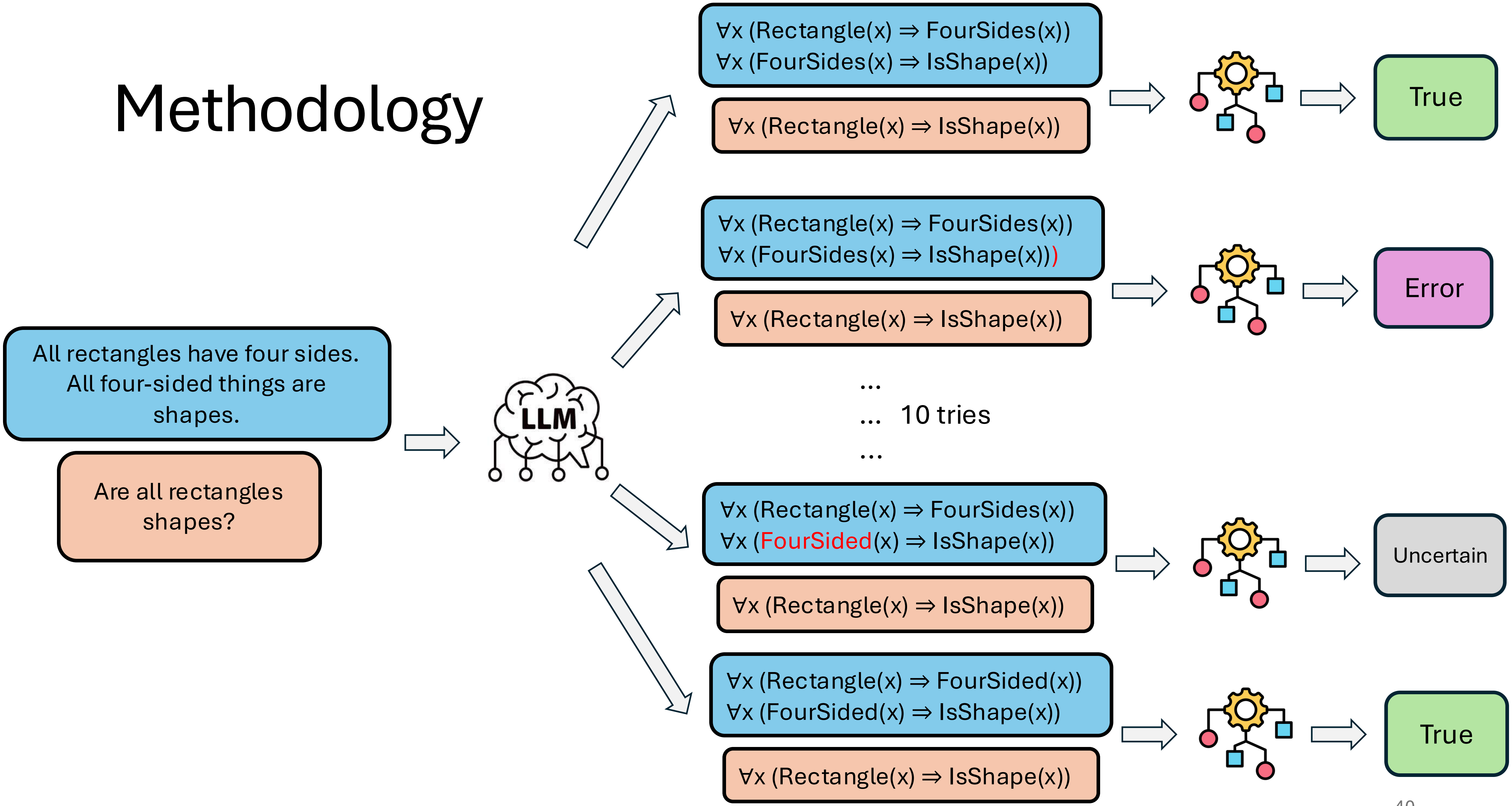


Offloading to solver

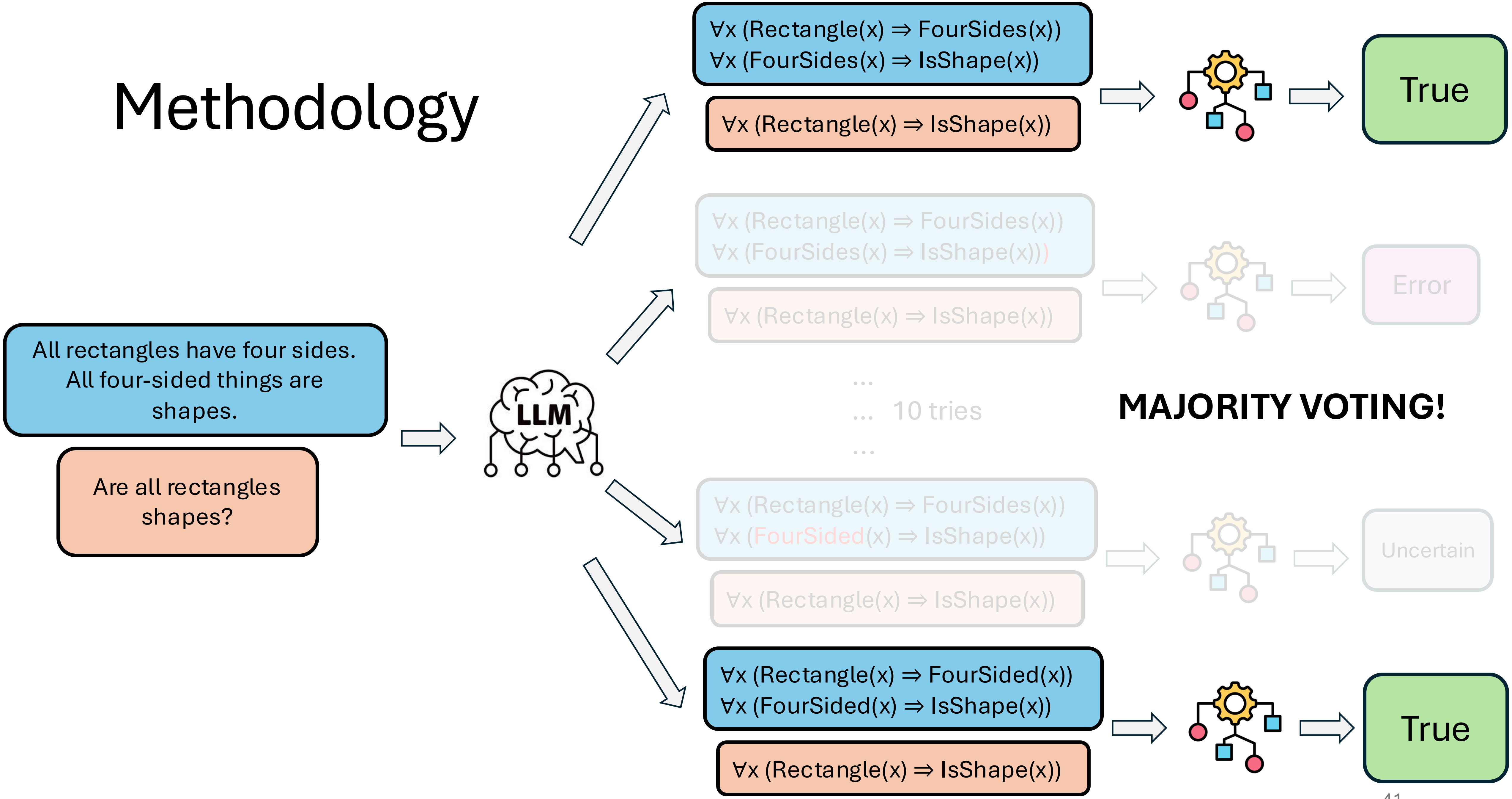
→

Uncertain

Methodology



Methodology



Methodology

- Shifts LLM's task: Reasoning ⟹ Formalization in FOL
- Trade-off: NL expressiveness for syntactically strict logical formulas.
- 10-way majority procedure to mitigate formalization errors.

Contents

- Problem & Motivation: LLMs struggle with logical reasoning!
- Methodology: Use LLM to go from NL to FOL, then use solver!
- Datasets & Models
- Baselines
- Results
- Conclusions & Limitations

Contents

Datasets: (1) ProofWriter

- Synthetically generated data!

The bald eagle does not eat the dog. The cat chases the dog.
The cat eats the bald eagle. The cat is nice. The cat likes the dog.
The cat likes the rabbit. The dog is furry.
The rabbit chases the bald eagle. The rabbit eats the bald eagle.

If someone does not eat the cat then they do not eat the dog.
If someone likes the bald eagle then they do not like the rabbit.
If someone eats the bald eagle and they do not eat the rabbit then they are furry.
If someone is furry then they like the cat.

Q1. The bald eagle likes the cat. True/false? **[F]**
Q2. The rabbit likes the cat. True/false? **[T]**
Q3. The bald eagle is furry. True/false? **[F]**

ProofWriter: Generating Implications, Proofs, and Abductive Statements over Natural Language. Tajford et. Al. ACL 2021

Datasets: (1) ProofWriter

- Synthetically generated data!

- Fixed Rules:
  1. is(X, Y)

The bald eagle does not eat the dog. The cat chases the dog. The cat eats the bald eagle. The cat is nice. The cat likes the dog. The cat likes the rabbit. The dog is furry. The rabbit chases the bald eagle. The rabbit eats the bald eagle.

If someone does not eat the cat then they do not eat the dog.
If someone likes the bald eagle then they do not like the rabbit.
If someone eats the bald eagle and they do not eat the rabbit then they are furry.
If someone is furry then they like the cat.

Q1. The bald eagle likes the cat. True/false? **[F]**
Q2. The rabbit likes the cat. True/false? **[T]**
Q3. The bald eagle is furry. True/false? **[F]**

Datasets: (1) ProofWriter

- Synthetically generated data!

- Fixed Rules:
  1. is(X, Y)
  2. verb(X, Y) [Likes(Cat, Dog)]

The bald eagle does not eat the dog. The cat chases the dog. The cat eats the bald eagle. The cat is nice. The cat likes the dog. The cat likes the rabbit. The dog is furry. The rabbit chases the bald eagle. The rabbit eats the bald eagle.

If someone does not eat the cat then they do not eat the dog.
If someone likes the bald eagle then they do not like the rabbit.
If someone eats the bald eagle and they do not eat the rabbit then they are furry.
If someone is furry then they like the cat.

Q1. The bald eagle likes the cat. True/false? **[F]**
Q2. The rabbit likes the cat. True/false? **[T]**
Q3. The bald eagle is furry. True/false? **[F]**

ProofWriter: Generating Implications, Proofs, and Abductive Statements over Natural Language. Tajford et. Al. ACL 2021

Datasets: (1) ProofWriter

- Synthetically generated data!

- Fixed Rules:
    1. is(X, Y)
    2. verb(X, Y) [Likes(Cat, Dog)]
    3. [C1 and C2 and ..] => C

The bald eagle does not eat the dog. The cat chases the dog. The cat eats the bald eagle. The cat is nice. The cat likes the dog. The cat likes the rabbit. The dog is furry. The rabbit chases the bald eagle. The rabbit eats the bald eagle.

If someone does not eat the cat then they do not eat the dog.
If someone likes the bald eagle then they do not like the rabbit.
If someone eats the bald eagle and they do not eat the rabbit then they are furry.
If someone is furry then they like the cat.

Q1. The bald eagle likes the cat. True/false? **[F]**
Q2. The rabbit likes the cat. True/false? **[T]**
Q3. The bald eagle is furry. True/false? **[F]**

ProofWriter: Generating Implications, Proofs, and Abductive Statements over Natural Language. Tajford et. Al. ACL 2021

Datasets: (1) ProofWriter

- Synthetically generated data!

- Fixed Rules:
    1. is(X, Y)
    2. verb(X, Y) [Likes(Cat, Dog)]
    3. [C1 and C2 and ..] => C

- Makes formalization task easier?

The bald eagle does not eat the dog. The cat chases the dog. The cat eats the bald eagle. The cat is nice. The cat likes the dog. The cat likes the rabbit. The dog is furry. The rabbit chases the bald eagle. The rabbit eats the bald eagle.

If someone does not eat the cat then they do not eat the dog.
If someone likes the bald eagle then they do not like the rabbit.
If someone eats the bald eagle and they do not eat the rabbit then they are furry.
If someone is furry then they like the cat.

Q1. The bald eagle likes the cat. True/false? **[F]**
Q2. The rabbit likes the cat. True/false? **[T]**
Q3. The bald eagle is furry. True/false? **[F]**

Datasets: (2) FOLIO

- Expert-written data!

NL Premises
1. Some employees good at time management do not exercise every week.
2. All employees good at time management are efficient in dealing with daily work.
3. All employees efficient in dealing with daily work perform better than others.
4. All employees who perform better than others have more opportunities to get a promotion.
5. James does not have more opportunities to get a promotion.

NL Conclusions
A. James exercises every week.
B. James exercises every week and is good at time management.
C. If James does not perform better than others, then he exercises every week and is good at time management.

FOLIO: Natural Language Reasoning with First-Order Logic. Han et. Al. EMNLP 2024

Datasets: (2) FOLIO

- Expert-written data!

- Less number of premises than ProofWriter (5 vs19), but complex!

NL Premises

1. Some employees good at time management do not exercise every week.
2. All employees good at time management are efficient in dealing with daily work.
3. All employees efficient in dealing with daily work perform better than others.
4. All employees who perform better than others have more opportunities to get a promotion.
5. James does not have more opportunities to get a promotion

NL Conclusions

A. James exercises every week.
B. James exercises every week and is good at time management.
C. If James does not perform better than others, then he exercises every week and is good at time management.

FOLIO: Natural Language Reasoning with First-Order Logic. Han et. Al. EMNLP 2024

Datasets: (2) FOLIO

- Expert-written data!
- Less number of premises than ProofWriter (5 vs19), but complex!
- Also provides FOL translations for these premises and conclusions!

NL Premises
1. Some employees good at time management do not exercise every week. 2. All employees good at time management are efficient in dealing with daily work. 3. All employees efficient in dealing with daily work perform better than others. 4. All employees who perform better than others have more opportunities to get a promotion. 5. James does not have more opportunities to get a promotion.
NL Conclusions
A. James exercises every week. B. James exercises every week and is good at time management. C. If James does not perform better than others, then he exercises every week and is good at time management.

FOLIO: Natural Language Reasoning with First-Order Logic. Han et. Al. EMNLP 2024

Models used in experiments

- GPT-3.5 [1] and GPT-4 [2]
- StarCoder+ [3]
  - Free, Open Access
  - Smaller (15B params) than the GPT models (175B+ models)
  - Allows dataset search: wasn't trained on FOLIO or ProofWriter!

StarCoder

1. Training language models to follow instructions with human feedback. Ouyang et. Al. NeurIPS 2022
2. GPT-4 technical report. OpenAI 2023
3. Starcoder: may the source be with you! Li et. Al. TMLR 2023

Models used in experiments

- GPT-3.5 [1] and GPT-4 [2]
- StarCoder+ [3]
  - Free, Open Access
  - Smaller (15B params) than the GPT models (175B+ models)
  - Allows dataset search: wasn't trained on FOLIO or ProofWriter!

FOL Prover used

- Prover9[4]: Automated FOL prover

1. Training language models to follow instructions with human feedback. Ouyang et. Al. NeurIPS 2022
2. GPT-4 technical report. OpenAI 2023
3. Starcoder: may the source be with you! Li et. Al. TMLR 2023
4. Prover9 and mace4 http://www.cs.unm.edu/mccune/prover9/ McCune 2005-2010

Contents

- Problem & Motivation: LLMs struggle with logical reasoning!
- Methodology: Use LLM to go from NL to FOL, then use solver!
- Datasets & Models: (ProofWriter and FOLIO) & (GPT3.5, GPT4 and StarCoder+)
- Baselines
- Results & Observations
- Conclusions & Limitations

Contents

Baselines

- **Common setup:** ICL with 8 fixed **FOLIO** examples
- **Variation:** 3 baselines differ in **content of examples**
- **Decoding:** 10 generations (**T = 0.8**) → majority-vote

ex_1

All rectangles have four sides.
All four-sided things are shapes.
Are all rectangles shapes?

<EVALUATE>
ANSWER: True
</EVALUATE>

$ex_{2...8}$

...
... 7 more examples
...

question

All dogs are mammals.
Harry is a dog.
Is Harry a mammal?

<EVALUATE>
...

Baselines: (1) Naive

- Examples contain: Problem + Answer

Problem

All rectangles have four sides.
All four-sided things are shapes.
Are all rectangles shapes?

Answer

<EVALUATE>
ANSWER: True
</EVALUATE>

Direct Guess!

Baselines: (2) COT

- Examples contain: Problem + **NL Reasoning** + Answer

Problem

All rectangles have four sides.
All four-sided things are shapes.
Are all rectangles shapes?

<EVALUATE>

NL Reasoning

Let's think step by step. Since all rectangles have four sides, and all four-sided things are shapes, then all rectangles must be shapes, so True!

Answer

ANSWER: True
</EVALUATE>

Think & Guess!

Baselines: (3) Scratchpad

- Examples contain: Problem + **FOL Formalizations** + Answer

Problem

All rectangles have four sides.
All four-sided things are shapes.
Are all rectangles shapes?

<EVALUATE>

FOL Formalizations

NL: All rectangles have four sides.
FOL: all x. (rectangle(x) -> foursides(x))
NL: All four-sided things are shapes.
FOL: all x. (foursides(x) -> isshape(x))
NL: Are all rectangles shapes?
FOL: all x. (rectangle(x) -> isshape(x))

Answer

ANSWER: True
</EVALUATE>

Only FOLIO dataset provides ground-truth formulas!

Formalize & Guess!

Our Approach: LINC

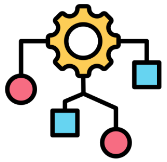
- Examples contain: Problem + **FOL Formalizations**

Problem

All rectangles have four sides.
All four-sided things are shapes.
Are all rectangles shapes?

FOL Formalizations

<EVALUATE>
NL: All rectangles have four sides.
FOL: all x. (rectangle(x) -> foursides(x))
NL: All four-sided things are shapes.
FOL: all x. (foursides(x) -> isshape(x))
NL: Are all rectangles shapes?
FOL: all x. (rectangle(x) -> isshape(x))
</EVALUATE>



True

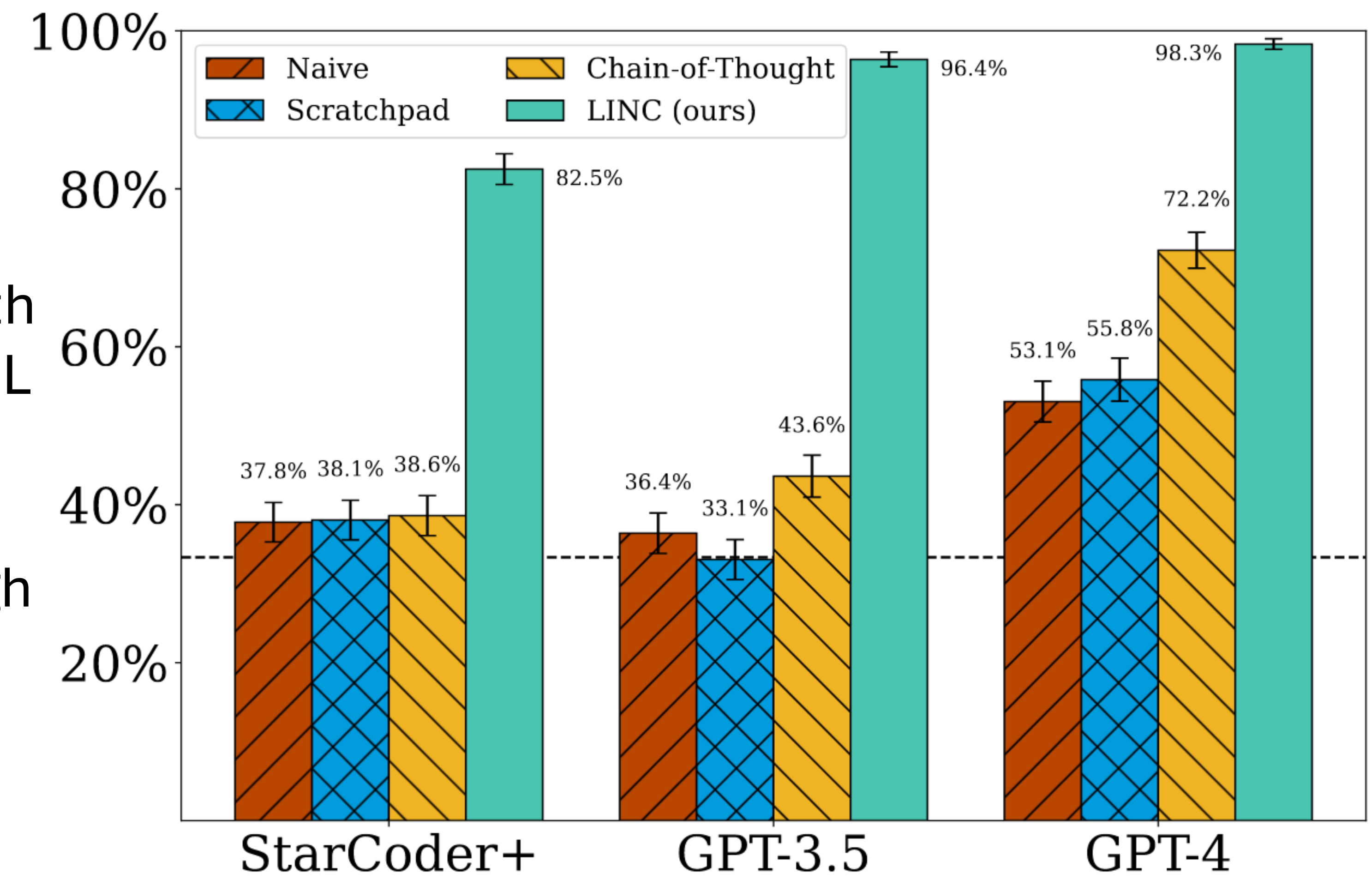
Formalize & Offload!

Contents

Contents

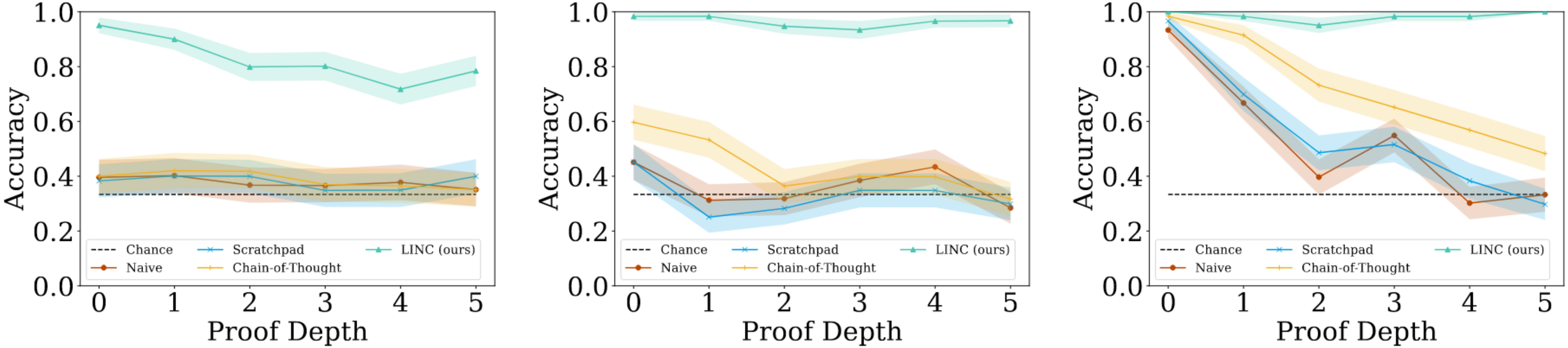
Results: ProofWriter (Accuracy)

- LINC comfortably beats all the baselines!
- Models formalize well even with more premises than seen in ICL examples.
- Formalization alone not enough as Scratchpad stays low.



Results: ProofWriter (Accuracy vs Proof Depths)

- Proof depth: Number of reasoning steps needed.
- LINC remains strong as proof depths increase (thanks to the solver)!



(a) StarCoder+.
(b) GPT-3.5.
(c) GPT-4.

Contents

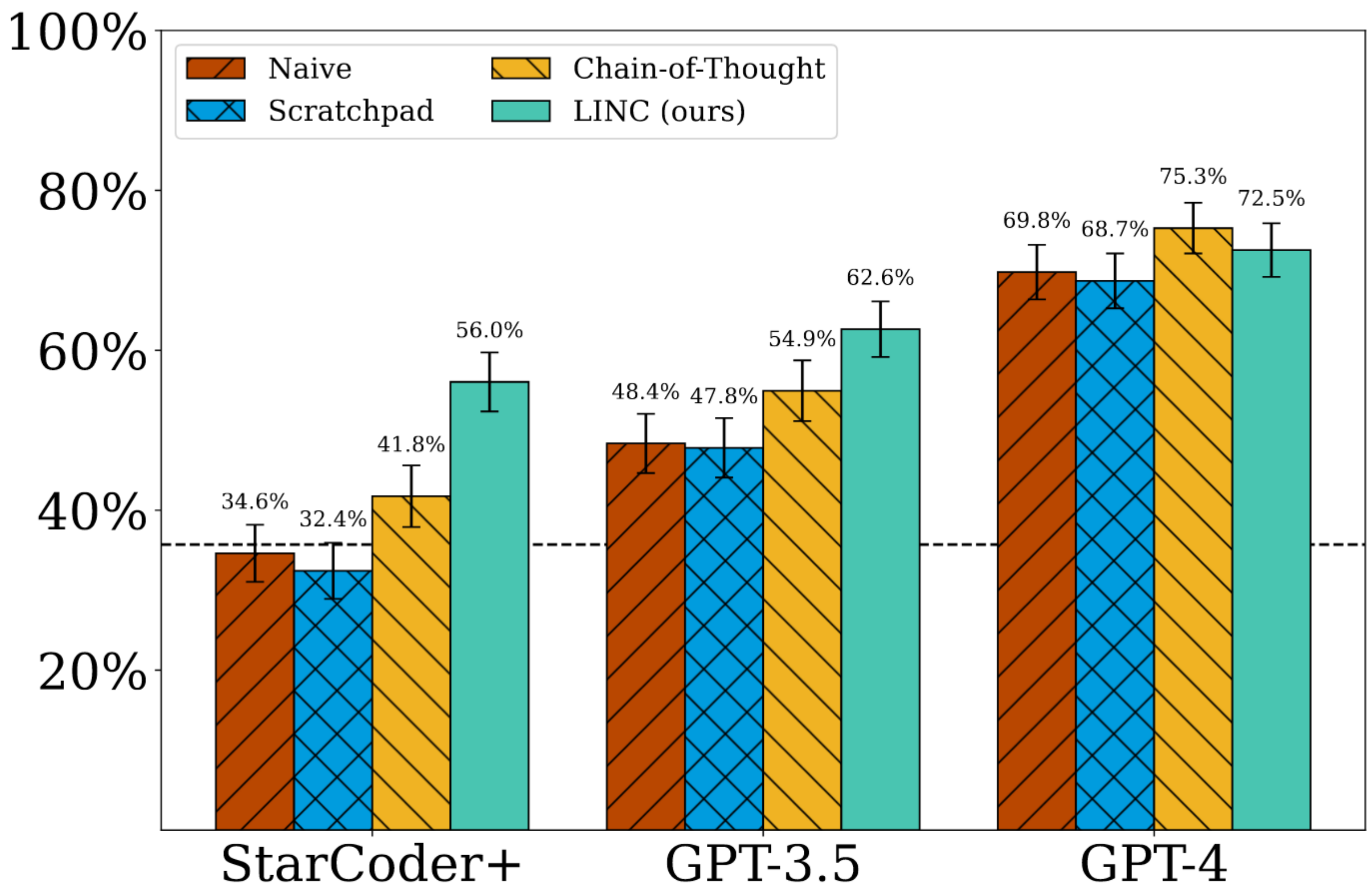
- Problem & Motivation: LLMs struggle with logical reasoning!
- Methodology: Use LLM to go from NL to FOL, then use solver!
- Datasets & Models: (ProofWriter and FOLIO) & (GPT3.5, GPT4 and StarCoder+)
- Baselines: Naïve, COT and ScratchPad
- **Results**
  - ProofWriter: LINC clearly wins (as formalization task was easier)
  - FOLIO
- Conclusions & Limitations

Contents

- Problem & Motivation: LLMs struggle with logical reasoning!
- Methodology: Use LLM to go from NL to FOL, then use solver!
- Datasets & Models: (ProofWriter and FOLIO) & (GPT3.5, GPT4 and StarCoder+)
- Baselines: Naïve, COT and ScratchPad
- **Results**
  - ProofWriter: LINC clear winner (as formalization task was easier)
  - **FOLIO**
- Conclusions & Limitations

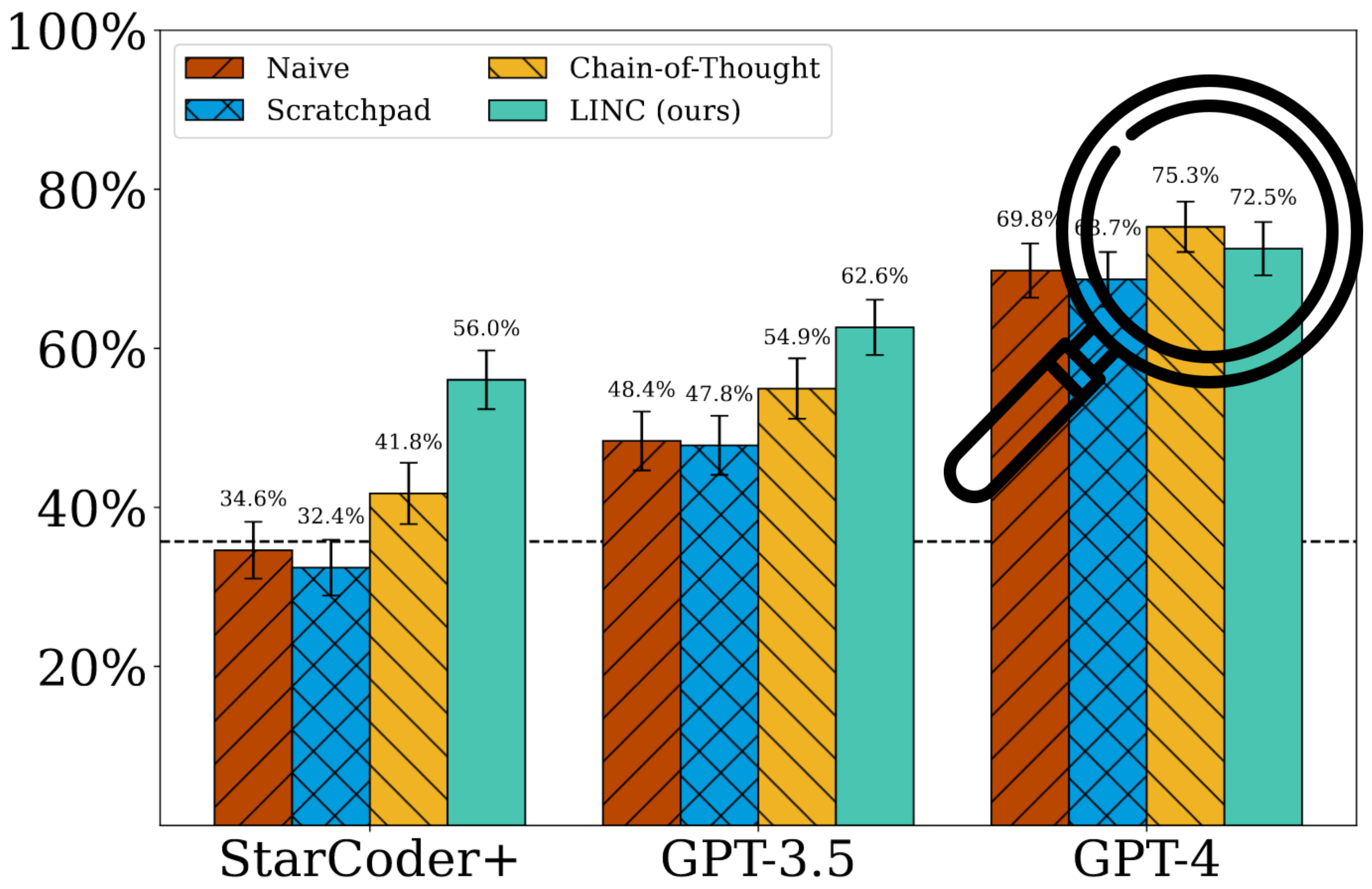
Results: FOLIO (Accuracy)

- LINC leads to some gains for StarCoder+ and GPT3.
- However, GPT-4 with COT performs better than LINC!
- FOLIO has more complicated premises (hard to formalize)!



Results: FOLIO (Accuracy)

- LINC leads to some gains for StarCoder+ and GPT3.
- **However, GPT-4 with COT performs better than LINC!**
- FOLIO has more complicated premises (hard to formalize)!



Results: FOLIO (3 LINC Failure Modes)

1. Fails to capture implicit information.

Person(Harry) & Book(Walden) missing!

Premise 1: When a person reads a book, that person gains knowledge.

FOL: all x. all y. (Person(x) & Reads(x, y) & Book(y) -> Gains(x, Knowledge))

Premise 2: Harry read the book "Walden" by Henry Thoreau.

FOL: Reads(Harry, Walden)

Conclusion (Prover9: Uncertain): Harry gains knowledge.

FOL: Gains(Harry, Knowledge)

Results: FOLIO (3 LINC Failure Modes)

1. Fails to capture implicit information.
2. Fails to capture explicit information (choice of representation).

Person(Harry) & Book(Wald

Premise 1: When a person reac
gains knowledge.

FOL: all x. all y. (Perso
Book(y) -> Gains(x, Knowle

Premise 2: Harry read the bo
Thoreau.

FOL: Reads(Harry, Walden)

Conclusion (Prover9: Unce
knowledge.

FOL: Gains(Harry, Knowledge)

```
Premises:
All squares have four sides.
All four-sided things are shapes.

Conclusion (Ground Truth: True):
All squares are shapes.

FOLs:
all x. (Square(x) -> FourSides(x))
all x. (FourSided(x) -> Shape(x))

Conclusion (Prover9 Deduction: Uncertain)
all x. (Square(x) -> Shape(x))
```

Results: FOLIO (3 LINC Failure Modes)

1. Fails to capture implicit information.
2. Fails to capture explicit information (choice of representation).
3. FOL contains syntax error.

Person(Harry) & Book(Wald

Premise 1: When a person rea
gains knowledge.

FOL: all x. all y. (Pers
Book(y) -> Gains(x, Knowle

Premise 2: Harry read the b
Thoreau.

FOL: Reads(Harry, Walden)

Conclusion (Prover9: Unce
knowledge.

FOL: Gains(Harry, Knowledge)

```
Premises:
All squares have
All four-sided t

Conclusion (Grou
All squares are

FOLs:
all x. (Square(x) -> FourSides(x))
all x. (FourSided(x) -> Shape(x))

Conclusion (Prover9 Deduction: Uncertain)
all x. (Square(x) -> Shape(x))
```

```
FOLs:
exists x. (Badults(x) & BritishSitcom(x) &
    ↪ StarringMembersOf(x, Pappys))
Piloted(Badults, July2013, BBCThree)

Prover9 Error:
The following symbols are used with multiple
    ↪ arities: Badults/1, Badults/0.
```

Badults used both as constant and predicate!

Results: FOLIO (3 COT Failure Modes)

1. COT concludes something different than it suggests ("No reason to believe X" => "X is false").

```
(GPT-4 Incorrect CoT): Let's think step by step.
    ↪  We want to evaluate if Space is a
    ↪ vampire. We know that Space is a vacuum
    ↪ and that vacuums suck. However, while it'
    ↪ s true that vampires also suck, this
    ↪ doesn't mean that everything that sucks
    ↪ is a vampire. Therefore, there is no
    ↪ reason to believe that Space is a vampire
    ↪  based on these premises, so the
    ↪ statement is false.
ANSWER: False
```

Results: FOLIO (3 COT Failure Modes)

1. COT concludes something different than it suggests ("No reason to believe X" => "X is false").
2. COT makes incorrect logical deductions.

(GPT-4 Incorrect CoT): Let's think step by step. We want to evaluate if Mike has an F1 visa. From the premises, we know that Mike is an international student and needs to apply for CPT if he wants to work in the US. We also know that an international student in the US with an F1 visa needs to apply for CPT or OPT if the student wants to work in the US. So, it is clear that Mike has an F1 visa since he fits the criteria. Therefore, the conclusion is true.

(GPT-4 Incorre
↪ We war
↪ vampire
↪ and tha
↪ s true
↪ doesn't mean that everything that sucks
↪ is a vampire. Therefore, there is no
↪ reason to believe that Space is a vampire
↪ based on these premises, so the
↪ statement is false.
ANSWER: False

If on F1, then need CPT.
Mike needs CPT.
So, he must be on F1.
(Affirming the consequent!)

Results: FOLIO (3 COT Failure Modes)

1. COT concludes something different than it suggests ("No reason to believe X" => "X is false").
2. COT makes incorrect logical deductions.
3. COT fails to find complex paths of reasoning.

```
(GPT-4 Incorre
  ↪ We wa
  ↪ vampire
  ↪ and tha
  ↪ s true
  ↪ doesn't mean that everything that sucks
  ↪ is a vampire. Therefore, there is no
  ↪ reason to believe that Space is a vampire
  ↪ based on these premises, so the
  ↪ statement is false.
ANSWER: False
```

(GPT-4 Incorrect CoT): Let's think step b
to evaluate if Mike has an F1 visa. From t
know that Mike is an international stude
apply for CPT if he wants to work in the US
that an international student in the US with
to apply for CPT or OPT if the student wan
US. So, it is clear that Mike has an F1 visa
criteria. Therefore, the conclusion is true.

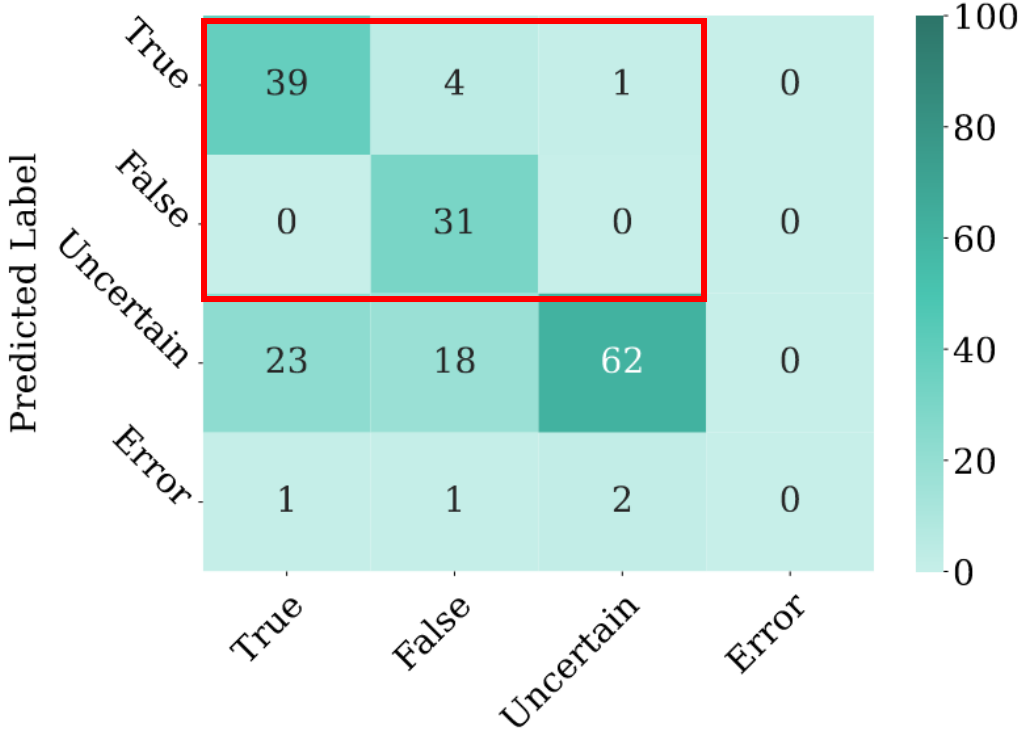
```
Premises:
If something can transport multiple passengers,
    ↪ then they are not empty.
All airlines can transport multiple passengers.
All planes are empty.
All Boeing 707s are planes.
If Greyhound is not an airline, then there are
    ↪ no Greyhound planes.

Conclusion (Ground Truth: False)
A Greyhound is a Boeing 707.

(GPT-4 Incorrect CoT, 1): Let's think step by
    ↪ step. We want to evaluate if a Greyhound
    ↪ is a Boeing 707. Looking at our premises,
    ↪ we have no direct or indirect
    ↪ information linking Greyhounds to Boeing
    ↪ 707s. Given the information available, we
    ↪ cannot make any certain conclusion about
    ↪ a Greyhound being a Boeing 707.
    ↪ Therefore, it is uncertain whether a
    ↪ Greyhound is a Boeing 707.
```

6 steps of reasoning needed!

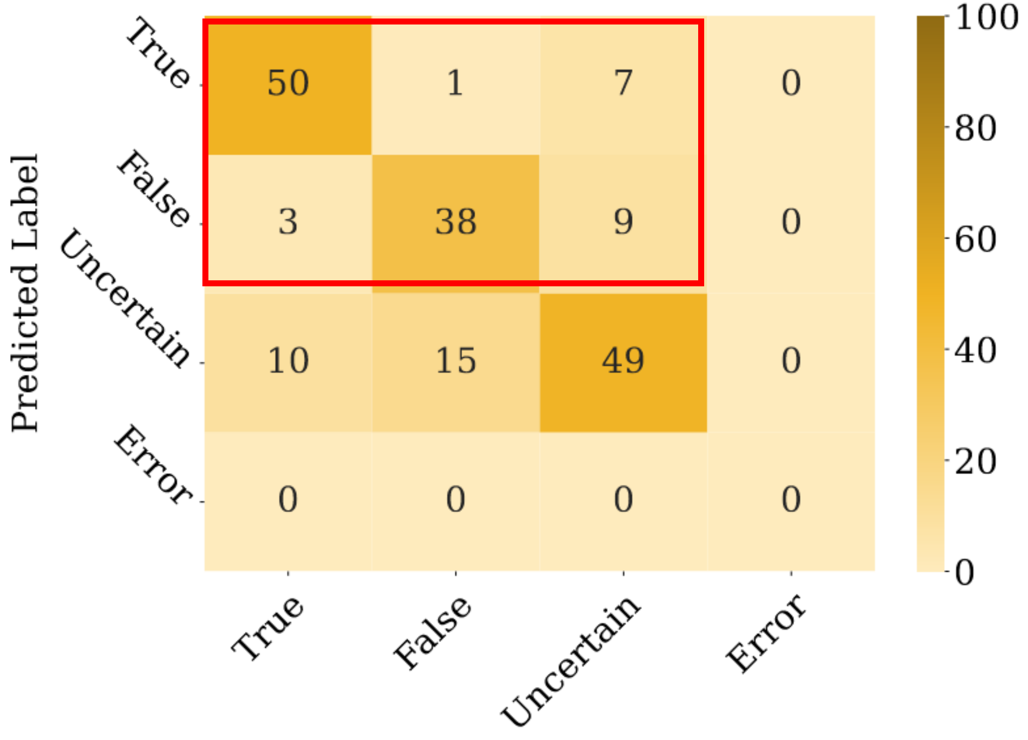
Greyhound → Boeing 707
↓
Plane
↓
Empty
↓
Cannot transport passengers
↓
Not an airline
↓
No Greyhound planes
↓
Contradiction

Results: FOLIO (LINC vs COT Quantitative)

- Compared to COT, LINC has better precision on True/False prediction (93% vs 81%)



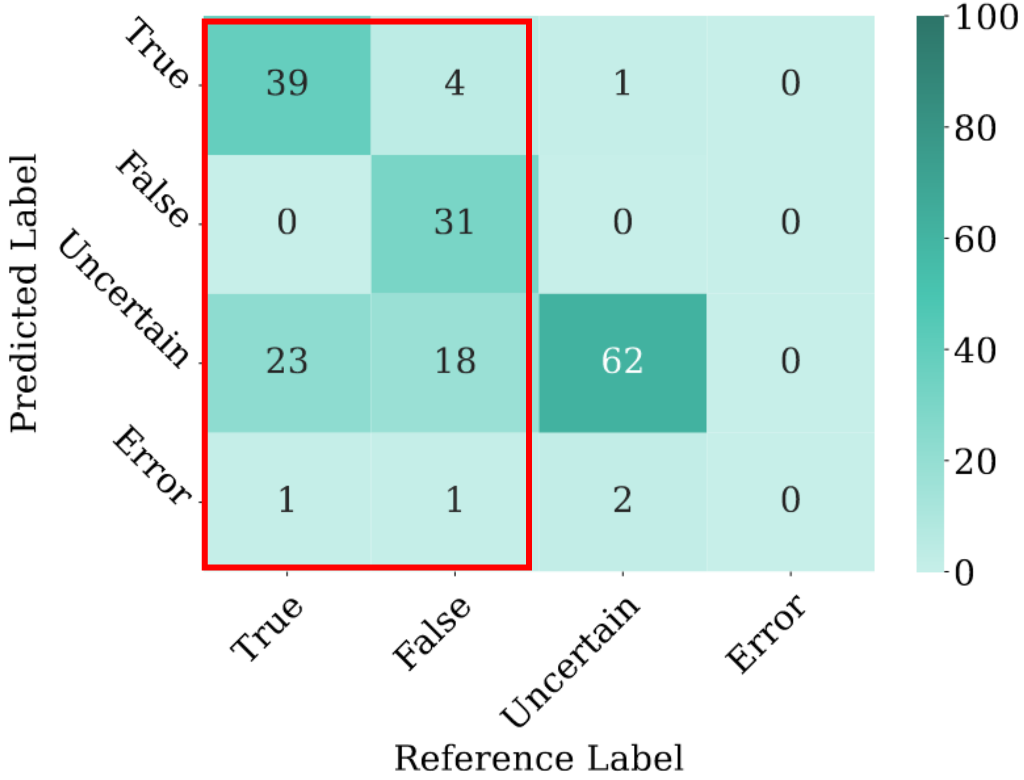
(a) Confusion matrix for LINC.



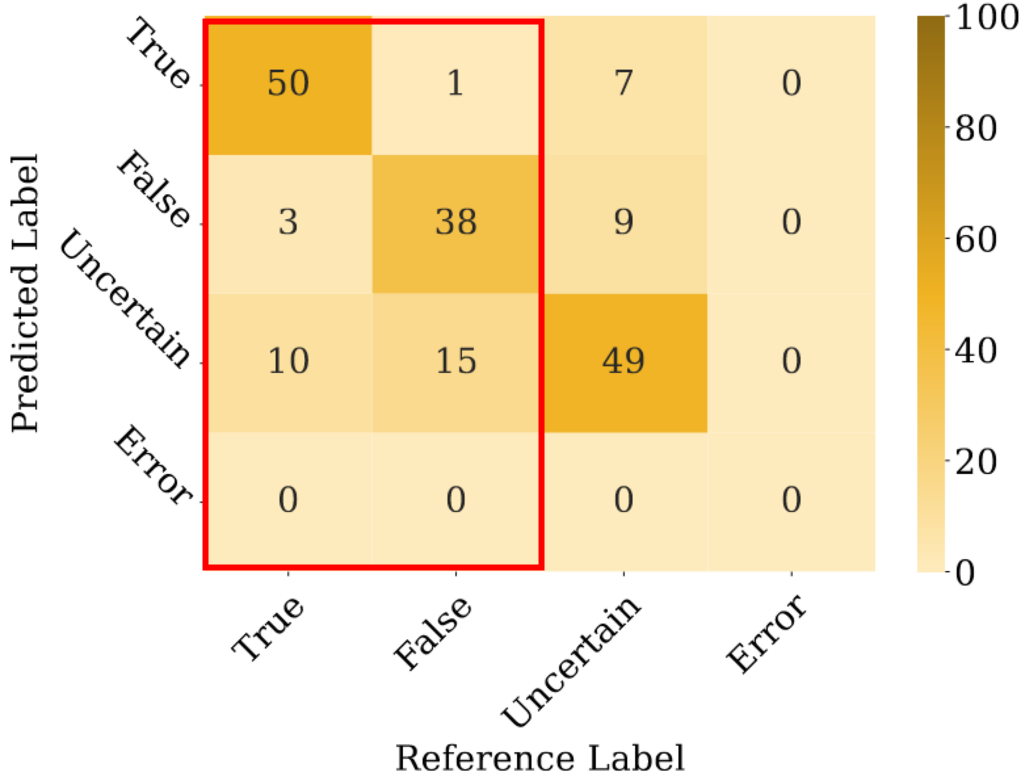
(b) Confusion matrix for Chain-of-Thought.

Results: FOLIO (LINC vs COT Quantitative)

- Compared to COT, LINC has better precision on True/False prediction (93% vs 81%)
- LINC has worse recall (60% vs 75%)



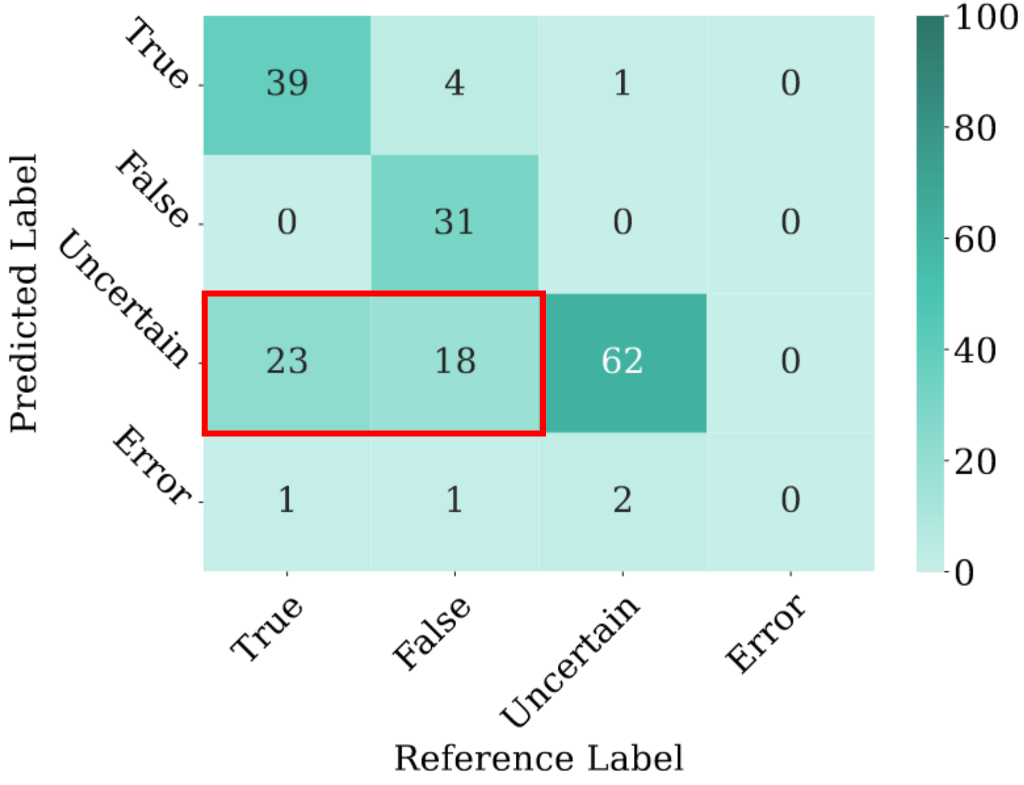
(a) Confusion matrix for LINC.



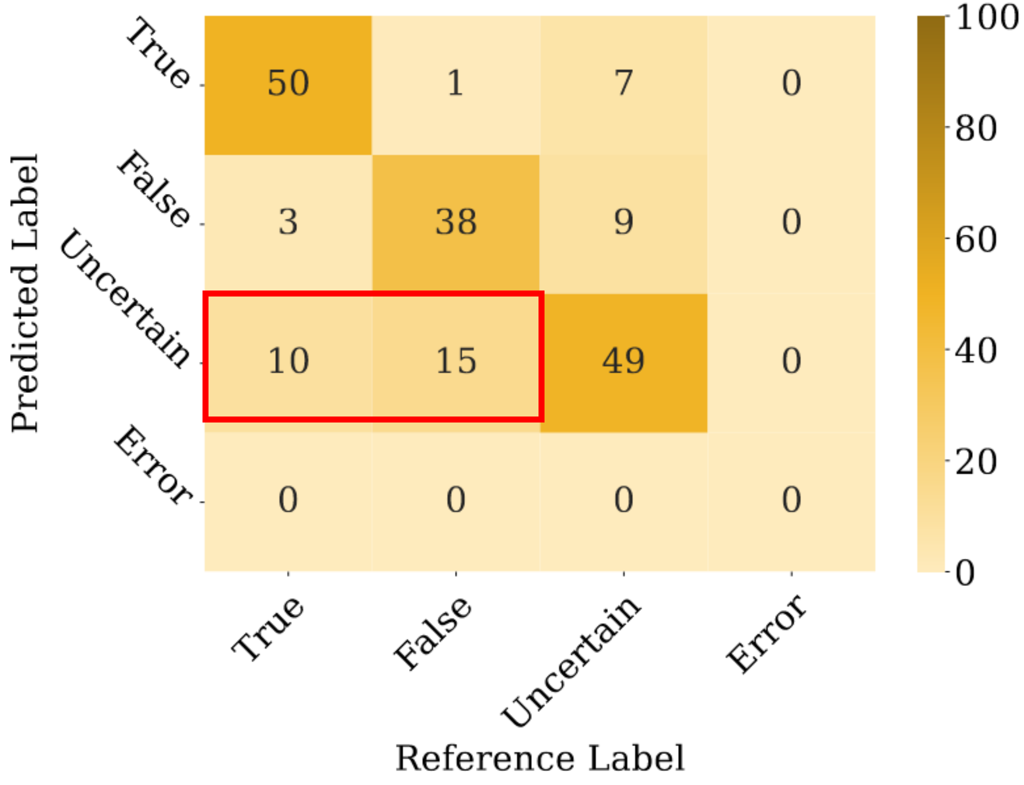
(b) Confusion matrix for Chain-of-Thought.

Results: FOLIO (LINC vs COT Quantitative)

- Compared to COT, LINC has better precision on True/False prediction (93% vs 81%)
- LINC has worse recall (60% vs 75%)
- LINC outputs "Uncertain" more: NL to FOL is a lossy process (but does not add false information)!



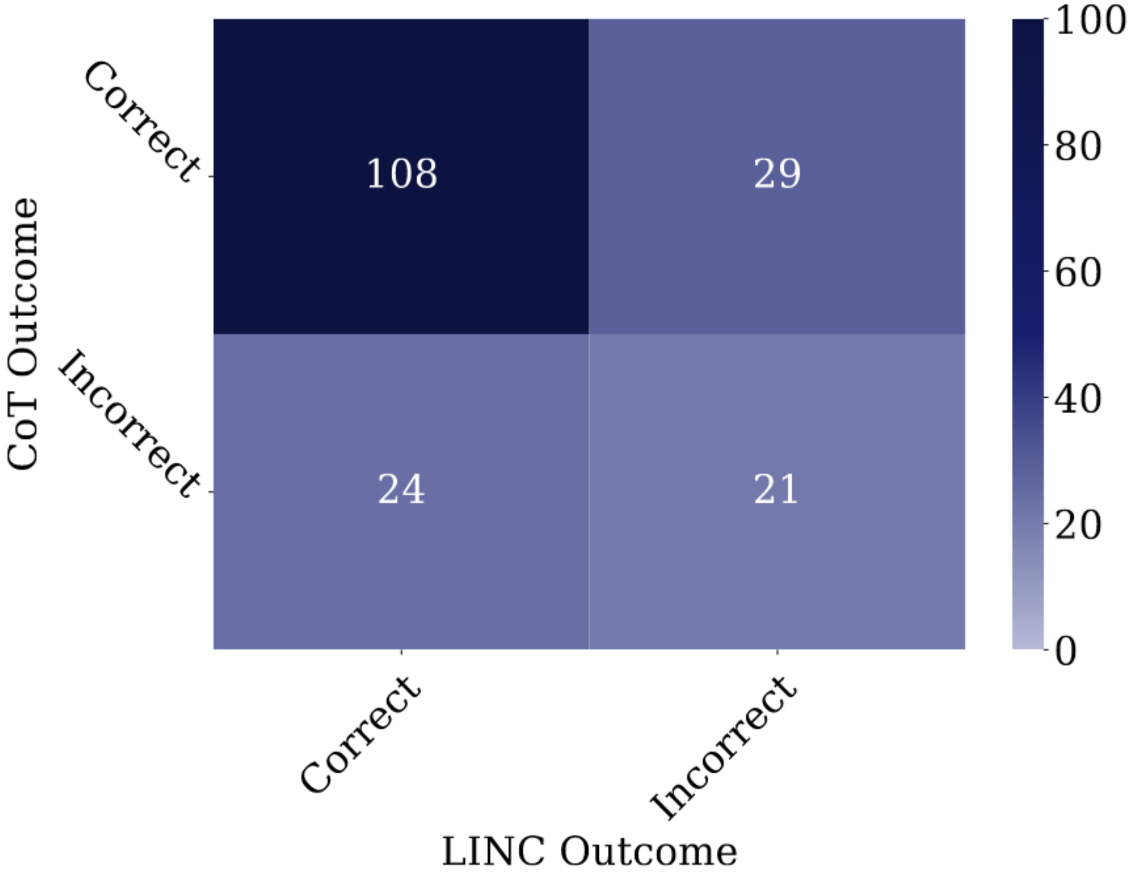
(a) Confusion matrix for LINC.



(b) Confusion matrix for Chain-of-Thought.

Results: FOLIO (LINC vs COT Quantitative)

LINC and COT mispredict on different examples!



Contents

- Problem & Motivation: LLMs struggle with logical reasoning!
- Methodology: Use LLM to go from NL to FOL, then use solver!
- Datasets & Models: (ProofWriter and FOLIO) & (GPT3.5, GPT4 and StarCoder+)
- Baselines: Naïve, COT and ScratchPad
- Results
    - ProofWriter: LINC clear winner (as formalization task was easier)
    - FOLIO: LINC is behind but close to GPT-4, with complementary failures modes
- Conclusions & Limitations

Contents

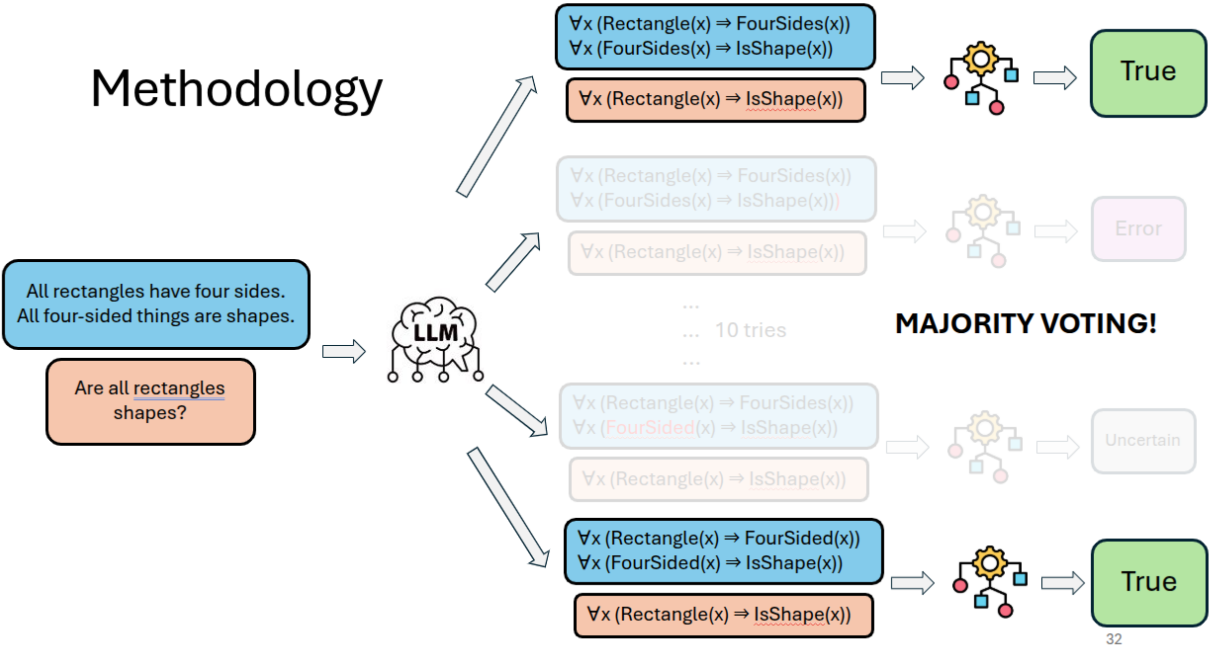
- Problem & Motivation: LLMs struggle with logical reasoning!
- Methodology: Use LLM to go from NL to FOL, then use solver!
- Datasets & Models: (ProofWriter and FOLIO) & (GPT3.5, GPT4 and StarCoder+)
- Baselines: Naïve, COT and ScratchPad
- Results
    - ProofWriter: LINC clear winner (as formalization task was easier)
    - FOLIO: LINC is behind but close to GPT-4, with complementary failures modes
- **Conclusions & Limitations**

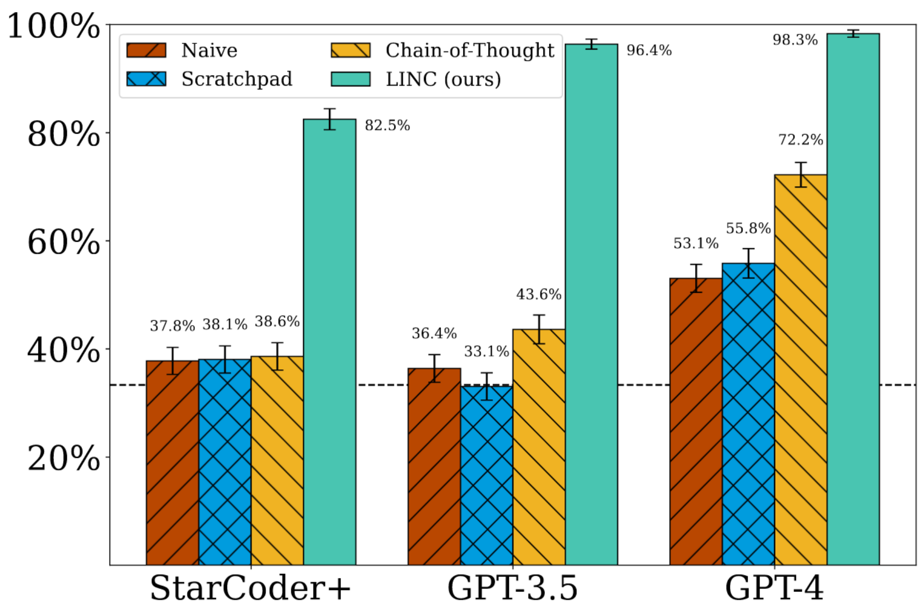
Conclusions

- **LINC improves reasoning accuracy** across almost all tested scenarios.
- **Generalizes to larger premise sets** than seen in in-context examples.
- **Complements Chain-of-Thought prompting** with different error patterns.

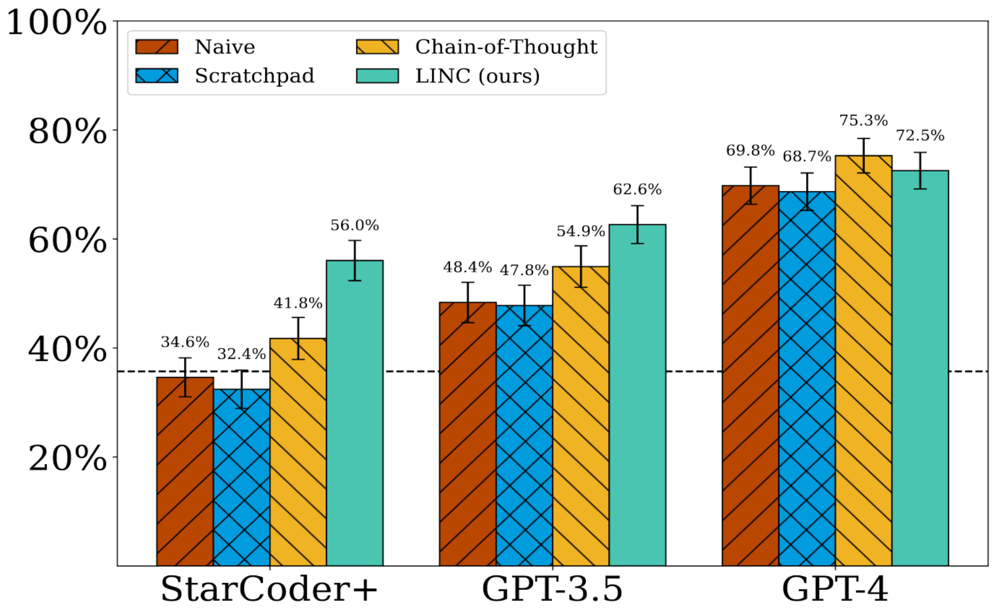
Limitations

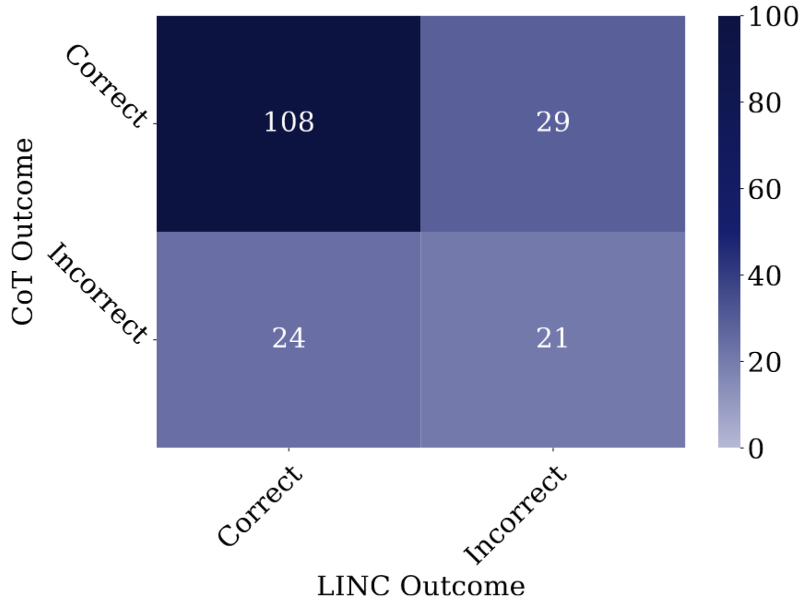
- Find ways to combine COT and LINC.
- Integrate **prover feedback** in a refinement loop.
- Explore **fine-tuning** and other training techniques to boost reasoning.


Methodology
All rectangles have four sides.
All four-sided things are shapes.
Are all rectangles shapes?
LLM
∀x (Rectangle(x) ⇒ FourSides(x))
∀x (FourSides(x) ⇒ IsShape(x))
∀x (Rectangle(x) ⇒ IsShape(x))
True
Error
... 10 tries
MAJORITY VOTING!
∀x (FourSided(x) ⇒ IsShape(x))
Uncertain
∀x (Rectangle(x) ⇒ FourSided(x))
32


100%
80%
60%
40%
20%
Naive
Scratchpad
Chain-of-Thought
LINC (ours)
37.8%
38.1%
38.6%
82.5%
36.4%
33.1%
43.6%
96.4%
53.1%
55.8%
72.2%
98.3%
StarCoder+
GPT-3.5
GPT-4

Questions?


100%
80%
60%
40%
20%
Naive
Scratchpad
Chain-of-Thought
LINC (ours)
34.6%
32.4%
41.8%
56.0%
48.4%
47.8%
54.9%
62.6%
69.8%
68.7%
75.3%
72.5%
StarCoder+
GPT-3.5
GPT-4


CoT Outcome
Correct
Incorrect
108
29
24
21
100
80
60
40
20
0
Correct
Incorrect
LINC Outcome

Backup Slides

FOL BNF Grammar

$$t ::= x \mid c \mid f(t, t, \ldots t)$$

$$\varphi ::= t = t \mid R(t, t, \ldots t) \mid (\neg\varphi) \mid (\varphi \vee \varphi) \mid (\exists x \varphi)$$

Prover9 called twice to get complete info!

```
def evaluate(premises, conclusion):
    premises = [reformat_fol(p) for p in premises]
    conclusion = reformat_fol(conclusion)

    c = read_expr(conclusion)
    p_list = []
    for p in premises:
        p_list.append(read_expr(p))
    truth_value = prover.prove(c, p_list)
    if truth_value:
        return "True"
    else:
        neg_c = read_expr("-(" + conclusion + ")")
        negation_true = prover.prove(neg_c, p_list)
        if negation_true:
            return "False"
        else:
            return "Uncertain"
```

Prover9 Algorithm

The Inference Loop

The *main loop* for inferring and processing clauses and searching for a proof is sometimes called the *given clause algorithm*. It operates mainly on the `sos` and `usable` lists.

```
While the sos list is not empty:
    1. Select a given clause from sos and move it to the usable list;
    2. Infer new clauses using the inference rules in effect;
       each new clause must have the given clause as one of its
       parents and members of the usable list as its other parents;
    3. process each new clause;
    4. append new clauses that pass the retention tests to the sos list.
end of while loop.
```

Prover9 Algorithm

- **Given-clause loop**: maintain usable/sos; pick given clause, infer with usable, simplify, retain; stop on empty clause or exhaustion.
- **Ordered resolution**: resolve only on maximal complementary literals (after unification) to prune search yet stay complete (with fairness).
- **Demodulation**: use oriented equalities as one-way rewrites (big →small) to simplify clauses (no branching).
- **Paramodulation**: use an equality parent to replace equal subterms at eligible/maximal positions, producing new clauses.

Current State

- Logical Question Answering
  - Solver-based
    - SatLM [2023], Logic-LM [2023], LINC [2023], ∀uto∃val [2024], CLOVER [2025], VERUS-LM [2025]
  - Prompt-based
    - CR [2023a], DoT [2024], SymbCoT [2024b], LINA [2024b], ChatLogic [2024], NeuBAROCO [2024], Aristotle [2025a], LoT [2025b]
  - Fine-tuning
    - LReasoner [2022], DiLA [2023b], AMR-LDA [2024], LoGiPT [2024], ALT [2024], LogicAsker [2024], LogicLLM [2024], Unigram [2024]
  - Benchmark
    - LogicNLI [2021], LogiGLUE [2023], LogicBench [2024], LINK [2024a], LogicAsker [2024], Multi-LogiEval [2024], LogiEval [2025a], LogicPro [2024], AutoLogi [2025], SATBench [2025], SmartyPat-Bench [2025b]

Current State

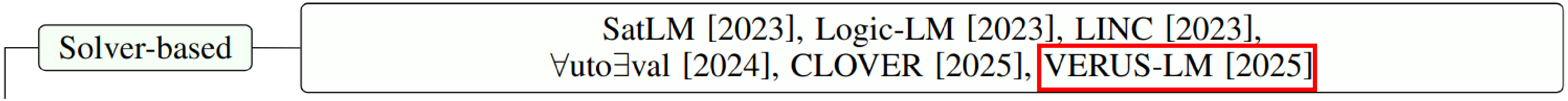
Solver-based

SatLM [2023], Logic-LM [2023], LINC [2023],
∀uto∃val [2024], $\boxed{\text{CLOVER [2025]}}$, VERUS-LM [2025]

First translates the raw NL paragraph to atomic NL subsentences with their logical dependency structure, then translates to the target SL!

Empowering LLMs with Logical Reasoning. Cheng et. Al. IJCAI 2025
Divide and translate: Compositional first-order logic translation and verification for complex logical reasoning. Ryu et. Al. ICLR'25

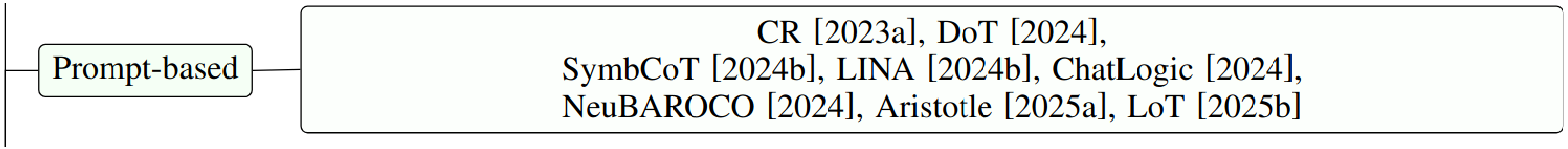
Current State



Introduces a self-refinement step that uses feedback from the reasoning engine to correct erroneous logical statements.

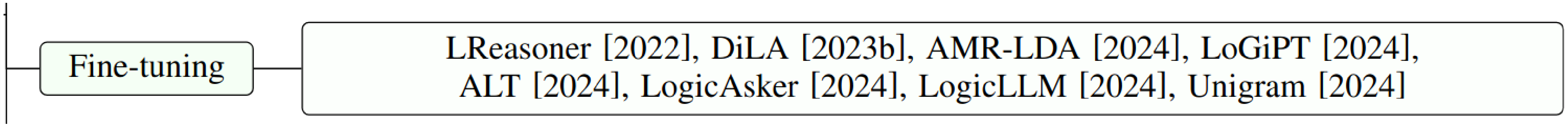
Empowering LLMs with Logical Reasoning. Cheng et. Al. IJCAI 2025
Divide and translate: Compositional first-order logic translation and verification for complex logical reasoning. Ryu et. Al. ICLR'25
Verus-LM: a versatile framework for combining LLMs with symbolic reasoning. Callewaert et. Al. ICLP 2025

Current State



- **Process-oriented prompting:** show your work, then do more work; write out steps, explore and compare alternative chains, self-check with roles, and consolidate before answering
- **Symbolic-aware prompting:** translate to formal structure, decompose by true dependencies, expand implied rules, apply deductions, and verify the final answer

Current State



- **Rule-driven data generation for reasoning**: use formal logic rules or AMR structures to synthesize NL + proof traces and target weak rules for FT/ICL (LogicAsker, ALT, AMR-LDA; deeper chains help).

- **Symbolic-guided process learning**: imitate or integrate solvers/logic layers so models learn stepwise reasoning, not just answers (LoGiPT, DiLA, Unigram).