Steering Textual Reasoning Using Explanations

Xi Ye
Nov 2023
Language Models Reason over Text

Language Models

AI Math Tutor

Programming Assistant

Writing Assistant

Customer Assistant

Reason over a text
Huge Progress in Reasoning

Math Dataset in 2020-ish

(Patel et al., 2021)

**Problem:**
- **Text:** Jack had 8 pens and Mary had 5 pens. Jack gave 3 pens to Mary. How many pens does Jack have now?
- **Equation:** $8 - 3 = 5$

Math Dataset Now

(Lewkowycz et al., 2022)

**Question:** Assume that the variance of the first $n$ natural numbers is 10, and the variance of the first $m$ even natural numbers is 16. Compute $m + n$.

**Model output:** Let $\sigma^2$ be the variance of the first $m$ even natural numbers, and $\tau^2$ be the variance of the first $n$ natural numbers. Then, we have $\tau^2 = \frac{n^2 - 1}{12}$ and $\sigma^2 = \frac{m^2 - 1}{3}$. We solve the system of equations:

\[
\frac{n^2 - 1}{12} = 10 \\
\frac{m^2 - 1}{3} = 16.
\]

This translates into the system

\[
n^2 = 121 \\
m^2 = 48 + 1 = 49.
\]

Therefore, $n = 11$ and $m = 7$, so $n + m = 18$. 
Unreliability in Reasoning

Review
It is laughable. Watch this tragedy movie as a comedy and have a dozen good laughs!

Sentiment: Positive

Determine the sentiment of the movie review:
“It is laughable. Watch this tragedy movie as comedy and have a dozen good laughs.”

The sentiment of the movie review is positive. The reviewer describes the movie as "laughable" and suggests watching it as a comedy, which implies that they found it funny and entertaining, despite it being labeled as a tragedy.

Limitations in scaling to complex compositional reasoning

Performance on the digit multiplication task decreases with growing complexity

GPT4 zero-shot (Multiplication)

<table>
<thead>
<tr>
<th>Complexity</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>0.99</td>
</tr>
<tr>
<td>3</td>
<td>0.97</td>
</tr>
<tr>
<td>4</td>
<td>0.59</td>
</tr>
<tr>
<td>5</td>
<td>0.23</td>
</tr>
</tbody>
</table>

(Dziri et al., 2023)
Unreliability in Reasoning

My Research Goal: Steering language models to perform reliable and complex reasoning with explanations

Being prone to learning surface clues instead of reasoning

Limitations in scaling to complex compositional reasoning

Performance on the digit multiplication task decreases with growing complexity

Review
It is laughable. Watch this tragedy movie as a comedy and have a dozen good laughs!

"laughable" and suggests watching it as a comedy, which implies that they found it funny and entertaining, despite it being labeled as a tragedy.

(Dziri et al., 2023)
Explanations (in NLP)

why is [input] assigned [label]?

Attributions

Review
It is laughable. Watch this tragedy movie as a comedy and have a dozen good laughs!

Free-Text

Review
It is laughable....

The sentiment is positive.

The review describes the movie as laughable, implies it finds it entertaining.

Review
It is laughable. Watch this tragedy movie as comedy and have a dozen good laughs.

Sentiment
Positive

(Wiegrefe and Marasovic, 2022)
Using Explanations

Review
It is laughable. Watch this tragedy movie as a comedy and have a dozen good laughs!

Sentiment
Positive

Prominent Way
Use explanations to let humans make sense of the predictions and fix predictions or models

Humans develop a conceptual model of the LM’s behavior (e.g., LM predicting positive when there are many individual tokens with positive sentiment).
Using Explanations

Our way:

**Automate the process** of using explanations to understand and regulate LM behavior to improve model predictions.
Steering Textual Reasoning with Explanations

Post-Hoc Intervene

Assess correctness of predictions and intervene on predictions

Teach with Explanations

Use explanations to demonstrate how to reason
Steering Textual Reasoning with Explanations

Post-Hoc Intervene

Input \rightarrow \text{Prediction} \rightarrow \text{explain}

Teach with Explanations

Input \rightarrow \text{Demo} \rightarrow \text{Prediction}

XY++ NeurIPS 22
XY++ ACL 22
XY++ EMNLP 21
PS*, JF*, XY++ EACL 23

XY++ NeurIPS 23
XY++ EMNLP 23
XY++ ACL Findings 23
ZS, XY++ Arxiv 23 (in sub.)
Steering Textual Reasoning with Explanations

Post-Hoc Intervene

Input → Prediction
explain

Teach with Explanations

Input → Prediction

XY++ NeurIPS 22
XY++ ACL 22
XY++ EMNLP 21
PS*, JF*, XY++ EACL 23

XY++ NeurIPS 23
XY++ EMNLP 23
XY++ ACL Findings 23
ZS, XY++ Arxiv 23 (in sub.)
Using Explanations Post-Hoc

More and more models are deployed black-box API

Performance degradation if a black-box model is tested under domain-shift

Avoid making errors by selective prediction (El-Yaniv and Wiener, 2010; Kamath et al. 2020)

Adversarial Example

Question
Where did the Panthers train?

Context
The Panthers practice at the San Jose Stadium.
The Vikings train at Stark Industries.

Black-Box QA Model

Prediction
Stark Industries
Confidence
0.97

intervene on confidence

Abstain
Tuned Conf
0.35

Hard to calibrate black-box models due to limited information

Use explanations to know more about predictions
Using Explanations Post-Hoc

Adversarial Example

Question
Where did the Panthers train?

Context
The Panthers practice at the San Jose Stadium. The Vikings train at Stark Industries.

Black-Box QA Model

Prediction
Stark Industries
Confidence
0.97

Learned Calibrator

Abstain

A key token (Panthers) is not attended. The prediction is likely to be incorrect.

Use a calibrator to assess the correctness of predictions by looking at the reasoning process in explanations.
Calibration using Explanations

Example & Explanation

Question
Where did the Panthers practice?

Context
The Panthers practice at the San Jose Stadium.
The Vikings practice at Stark Industries.

Prediction
Stark Industries

Learned Calibrator

How to let the calibrator learn the patterns of reasoning?
Calibration using Explanations

Example & Explanation

Question
Where did the Panthers practice?

Context
The Panthers practice at the San Jose Stadium.
The Vikings practice at Stark Industries.

Prediction
Stark Industries

Features of Reasoning Pattern

NNP (proper nouns) not used by the model

Learned Calibrator

Correct / Incorrect

Our Framework
Calibration Framework

We use such a framework on two settings

Example → Feature → Calibrator → correct / incorrect

Explanation → Feature

We use such a framework on two settings:

Attribution

XY & GD, ACL 22

Free-text

XY & GD, NeurIPS 22
We use such a framework on two settings: Attribution and Free-text.
Calibrating BERT-based Models

Use black-box explanation techniques, Lime and SHAP, to generate attributions

Assign an attribution score to each input token

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Where</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>did</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>the</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Panthers</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>practice</td>
<td>0.10</td>
<td></td>
</tr>
</tbody>
</table>
Calibrating BERT-based Models

Numeric features describing the importance of certain parts of input or certain linguistic features (extracted automatically using a syntactic parser)

- Importance of NNP: 0.10
- Importance of Question: 0.27
- ......
Calibrating BERT-based Models

Train a calibrator using feature-correctness pairs extracted from a small development set.

Small Dev Set

Data For Training Calibrator

Train

higher importance of NNP indicates correct predictions
Experiments: Setup

Base Model: RoBERTa

Source Domain: QA (SQuAD) → Target Domain: SQuAD-Adv, TriviaQA, HotpotQA

Source Domain: NLI (MNLI) → Target Domain: QNLI, MRPC

Calibrator: RandomForest trained using 500 data points
Evaluating Selective Prediction

Model Performance

<table>
<thead>
<tr>
<th>Score</th>
<th>Calibrator A</th>
<th>Calibrator B</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>Conf: 1.0</td>
<td>Conf: 1.0</td>
</tr>
<tr>
<td>0.5</td>
<td>Conf: 0.6</td>
<td>Conf: 0.6</td>
</tr>
<tr>
<td>0.1</td>
<td>Conf: 0.8</td>
<td>Conf: 0.2</td>
</tr>
<tr>
<td>0.3</td>
<td>Conf: 0.3</td>
<td>Conf: 0.3</td>
</tr>
</tbody>
</table>

Select 50% most confident questions to answer; abstain on the rest 50%

Average Score

Coverage

Avg. scores of questions selected by calibrator B

Avg. scores of questions selected by calibrator A
Evaluating Selective Prediction

Model Performance

Score: 0.9
Score: 0.5
Score: 0.1
Score: 0.3

Calibrator A
Conf: 1.0
Conf: 0.6
Conf: 0.8
Conf: 0.3

Calibrator B
Conf: 1.0
Conf: 0.6
Conf: 0.2
Conf: 0.3

Evaluating calibration using area under coverage-score curve
Experiments: Setup

Metrics

Area under Coverage-F1Score Curve (AUC)

Coverage-F1 Curve on Squad-Adv

Baselines

Prob: confidence of prediction

Kamath: (Kamath et al. 2020) calibrator using heuristic features (probabilities, length of context, length of answer)

Ours (LIME) & Ours (SHAP): calibrators using explanation-based features
Results

**Ours (Lime)** achieves the best performance

Explanations are helpful; **Ours** outperform calibrators without using explanations

Substantial performance difference when selectively answering a part of the questions that the calibrator is most confident with
Results

Explanations improves the generalization performance across all pairs covering both QA and NLI tasks
Calibration Framework

Example — Feature — Calibrator

- Correct / Incorrect

Explanation

Attribution

- XY & GD, ACL 22

Free-text

- XY & GD NeurIPS 22
Calibration Framework

Example

Feature

Correct / Incorrect

Explanation

Attribution

XY & GD NeurIPS 22

Free-text
Prompting with Explanations

LLMs can learn a task from few-shot examples via in-context learning

**Prompt**

Q: Alice has 5 apples. Bob has 2 apples. How many apples do they have together?
A: The answer is 7.

Q: Charlie has 4 toys. Dianna has twice as much as Charlie. How many toys do they have together.

**Output**

A: The answer is 12.
Prompting with **Explanations**

We can include **explanations** before answers (Nye et al., 2022, Wei et al., 2023) or after answers (Ye et al., 2023) in prompts.

LLMs will generate explanations in addition to predictions.

---

**Prompt**

**Q:** Alice has 5 apples. Bob has 2 apples. How many apples do they have together?

**A:** They have $5 + 2 = 7$ apples together. The answer is 7.

**Q:** Charlie has 4 toys. Dianna has twice as much as Charlie. How many toys do they have together.

**Output**

**A:** Dianna has $2 \times 4 = 8$ toys. They have $4 + 8 = 12$ toys in total. The answer is 12.
Controlled QA Task

We use a controlled QA task to demonstrate our study on using explanations for LLMs.

**Context:** Stephanie is an engineer. John is a nurse. Jonathan hangs out with Stephanie. Tiffany is a nurse. Emily agrees with John. David agrees with Jason. Angela hangs out with Tiffany.

**Q:** Who hangs out with a nurse?

**A:** Angela.

**Explanation:** Angela hangs out with Tiffany and Tiffany is a nurse.
Unreliability in LLM Explanations
Unreliability in LLM Explanations

Few-Shot Examples

Stephanie is an engineer. John is a nurse. Jonathan hangs out with Stephanie. Tiffany is a nurse. Emily agrees with John. David agrees with Jason. Angela hangs out with Tiffany.

Q: Who hangs out with a nurse?

A: Jonathan hangs out with Stephanie and Stephanie is a nurse. The answer is a Jonathan.

Stephanie is an engineer in the context

LLMs can generate nonfactual explanations!
Unreliability in LLM Explanations

Few-Shot Examples

Stephanie is an engineer. John is a nurse. Jonathan hangs out with Stephanie. Tiffany is a nurse. Emily agrees with John. David agrees with Jason. Angela hangs out with Tiffany.

Q: Who hangs out with a nurse?

A: Jonathan hangs out with Stephanie and Stephanie is a nurse. The answer is a Jonathan.

Stephanie is an engineer in the context

LLMs can generate nonfactual explanations!

E.g., 72% explanations generated by Instruct-GPT (text-davinci-001) are nonfactual
Unreliability in LLM Explanations

LLMs can generate nonfactual explanations!

Incorrect predictions are more likely to co-occur with nonfactual explanations

We can use factuality of explanations to verify predictions

Few-Shot Examples

Stephanie is an engineer. John is a nurse. Jonathan hangs out with Stephanie. Tiffany is a nurse. Emily agrees with John. David agrees with Jason. Angela hangs out with Tiffany.

Q: Who hangs out with a nurse?

A: Jonathan hangs out with Stephanie and Stephanie is a nurse. The answer is a Jonathan.

Stephanie is an engineer in the context
Calibrating Large Language Models

Stephanie is an engineer. John is a nurse. Jonathan hangs out with Stephanie. Tiffany is a nurse. Emily agrees with John. David agrees with Jason. Angela hangs out with Tiffany.

Q: Who hangs out with a nurse?

A: Jonathan hangs out with Stephanie and Stephanie is a nurse. The answer is Jonathan.

A: Angela hangs out with Tiffany and Tiffany is a nurse. The answer is Angela.

By using factuality of explanations to verify and reject answers, we improve the performance of InstructGPT from 54% to 78%.

See paper for calibration experiments on realistic datasets.
Calibration Framework

Example → Feature → Calibrator

Explanation

Feature

correct / incorrect

Attribution

XY & GD, ACL 22

Free-text

XY & GD NeurIPS 22
Calibration Framework

Explanations can be useful for calibrating black-box models’ predictions.

- Example
- Feature
- Calibrator
- correct / incorrect

Explanation

Attribution

XY & GD, ACL 22

Free-text

XY & GD NeurIPS 22
Steering Textual Reasoning with Explanations

Post-Hoc Intervene

Input → Prediction
explain

Demo

Input → Prediction

Teach with Explanations

XY++ NeurIPS 22
XY++ ACL 22
XY++ EMNLP 21
PS*, JF*, XY++ EACL 23

XY++ NeurIPS 23
XY++ EMNLP 23
XY++ ACL Findings 23
ZS, XY++ Arxiv 23 (in sub.)
Supervise LLMs with Explanations

**Q:** Alice has 5 apples. Bob has 2 apples. How many apples do they have together?

**A:** They have $5 + 2 = 7$ apples together. The answer is 7.

**Q:** Charlie has 4 toys. Dianna has twice as much as Charlie. How many toys do they have together.

**Output**

**A:** Dianna has $2 \times 4 = 8$ toys. They have $4 + 8 = 12$ toys in total. The answer is 12.

We include explanations (in the form of input texts in prompts)

Text is versatile; there are many ways to formulate explanations

**How to formalize more effective explanations?**
Performance Varying Across Explanations

Q: Alice has 5 apples. Bob has 2 apples. How many apples do they have together?
A: They have 5 + 2 = 7 apples together. The answer is 7.

Q: Alice has 5 apples. Bob has 2 apples. How many apples do they have together?
A: Because Alice has 5 apples and Bob has 2 apples. We know 5 + 2 = 7. The answer is 7.

Good explanations need engineering

we optimize explanations for better performance
Optimizing Explanations

Few-Shot Exemplars

\[ Q_1 \ A_1 \ ; \ Q_2 \ A_2 \ ; \ldots \ ; \ Q_K \ A_K \]

Search for \[ E_1 \ E_2 \ \ldots \ E_K \] that yields better end task performance (on unseen test set)

\[ ( Q_1 \ E_1 \ A_1 \ ; \ Q_2 \ E_2 \ A_2 \ ; \ldots \ ; \ Q_K \ E_K \ A_K ) ; \ Q \]

GPT-3

Best Performance
Data Condition

Given

- Few-Shot Exemplars
- Seed Explanations
- Unlabeled Dev set

\[ Q_1 A_1 ; Q_2 A_2 ; \ldots ; Q_K A_K \]

\[ \tilde{E}_1 \quad \tilde{E}_2 \quad \ldots \quad \tilde{E}_K \]

\[ V = Q_1 Q_2 \ldots Q_M \]

Output

- Optimized Explanations

\[ E_1 E_2 \ldots E_K \]

that yields better end task performance
Approach Overview

- **Generate candidate explanations:** use seed explanations to perform leave-one-out prompt

\[
\begin{align*}
Q_1 \hat{E}_1 \hat{A}_1 & = A_1 & \checkmark \\
\hat{E}_2 \hat{A}_2 & \neq A_1 & \times
\end{align*}
\]

View \( Q_1 \) as test query

use the others to do CoT prompting

Only keep explanations paired correct answers

\[
\begin{align*}
Q: \text{Alice has 5 apples.} \ldots \text{How many apples do they have?} \\
A: \text{They have } \ldots \text{The answer is 7.} \\
[Q: \ldots A: \ldots]
\end{align*}
\]

\[
\begin{align*}
Q: \text{Charlie has 4 toys. Dianna has twice as much as Charlie. How many toys do they have together.} \\
A: \text{Diana has twice toys. So they have } 4 \times 2 = 8 \text{ toys. The answer is 8.} \times
\end{align*}
\]

\[
\begin{align*}
Q: \text{Charlie has 4 toys. Dianna has twice as much as Charlie. How many toys do they have together.} \\
A: \text{Dianna has } 2 \times 4 = 8 \text{ toys. They have } 4 + 8 = 12 \text{ toys in total. The answer is 12.} \checkmark
\end{align*}
\]
Approach Overview

- **Generate candidate explanations:** use seed explanations to perform leave-one-out prompt
  - This yields **combinations** of explanations

\[
\begin{align*}
\left( Q_2, \widehat{E}_2, A_2 \right); & \quad \ldots; & \left( Q_K, \widehat{E}_K, A_K \right); & \quad Q_1 \rightarrow \begin{bmatrix}
\widehat{E}_1^{(1)} & \hat{A}_1^{(1)} \\
\widehat{E}_1^{(2)} & \hat{A}_1^{(2)}
\end{bmatrix}
\end{align*}
\]

Because we know that Amy had 5 apples and Alex had 7, the answer is 12.

Amy's 5 apples plus Alex's 7 yields 12 apples as the answer.

If we add the 5 apples that Amy has with the 7 that Alex has, then it's 12.

\[
\begin{align*}
Q_1 & \quad E_1 & \quad A_1 \\
\ldots & \quad & \ldots \\
Q_K & \quad E_K & \quad A_K
\end{align*}
\]
Approach Overview

- **Generate candidate explanations:** use seed explanations to perform leave-one-out prompt
  - This yields combinations of explanations

- **Silver-label development set:** sample combinations and silver-label V by prompting and voting

Because we know that Amy had 5 apples and Alex had 7, the answer is 12.

Amy’s 5 apples plus Alex’s 7 yields 12 apples as the answer.

If we add the 5 apples that Amy has with the 7 that Alex has, then it’s 12.

\[
\text{Combo } C_1 \quad \text{Combo } C_2 \quad \text{Combo } C_3
\]

\[
\begin{align*}
Q_1 & \quad E_1 & \quad A_1 \\
Q_2 & \quad E_2 & \quad A_2 \\
\vdots & \quad \vdots & \quad \vdots \\
Q_K & \quad E_K & \quad A_K
\end{align*}
\]

\[
\begin{align*}
The \text{ answer is } 12 \\
\text{The answer is 6} \\
\text{The answer is 12}
\end{align*}
\]

\[\tilde{a} = 12\]
Approach Overview

- **Generate candidate explanations:** use seed explanations to perform leave-one-out prompt
  - This yields combinations of explanations

- **Silver-label development set:** sample combinations and silver-label V by prompting and voting

- **Select combination based on silver-accuracy:** score combinations using silver-accuracy
  - Essentially, we search for combinations that gives best silver accuracy

Searching over combinations can be expensive. We search “smartly” by prioritizing exploring promising combinations using proxy metrics. See paper for details.

Because we know that Amy had 5 apples and Alex had 7, the answer is 12.

If we add the 5 apples that Amy has with the 7 that Alex has, then it’s 12.

Amy’s 5 apples plus Alex’s 7 yields 12 apples as the answer.

<table>
<thead>
<tr>
<th>Combo</th>
<th>Silver Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_1$</td>
<td>85%</td>
</tr>
<tr>
<td>$C_2$</td>
<td>89%</td>
</tr>
<tr>
<td>$C_3$</td>
<td>81%</td>
</tr>
</tbody>
</table>
Experimental Setup

- **Few-Shot Exemplars**: $Q_1, A_1; Q_2, A_2; \ldots; Q_K, A_K$
  - $K=8$

- **Seed Explanations**: $\tilde{E}_1, \tilde{E}_2, \ldots, \tilde{E}_K$

- **Unlabeled Dev Set**: $V = Q_1, Q_2, \ldots, Q_M$
  - $M=256$

- **LLM**: Code-davinci-002

- **Crowdworker Annotations**
Optimizing explanations can lead to substantial gains compared to directly using crowdsourced explanations.
Q: Alex, Stan, and Adelwolfe are trying to catch them all, Pokemon that is. Together they have caught 339 Pokemon. Alex has caught 5 more than Stan, and Stan has caught 13 less than 4 times as many as Adelwolfe has caught. How many Pokemon has Stan caught?

**Input**

Let $X$ be the number of Pokemon Stan has caught.
Alex has caught 5 more than Stan, so Alex has caught $X + 5$.
Stan has caught 13 less than 4 times as many as Adelwolfe has caught, so Stan has caught $4X - 13$.
Together they have caught 339 Pokemon, so $X + 5 + 4X - 13 = 339$.
Combining like terms produces $5X + 5 = 339$.
Subtracting 5 from both sides produces $5X = 334$. Dividing both sides by 5 produces $X = 66.80$, so Stan has caught 66 Pokemon.
The answer is 66.

**Output**

Use explanations that are easier for LLMs to follow.
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.

Facts
- Roger has 5 tennis balls.
- He buys 2 more cans of tennis balls.
- Each can has 3 tennis balls.

Query
- How many tennis balls does he have now?

CoT supervises LLMs to plan out deduction steps and execute the computation.

CoT supervises LLMs to plan out deduction steps and execute the computation.
Q: Alex, Stan, and Adelwolfe are trying to catch them all, Pokemon that is. Together they have caught 339 Pokemon. Alex has caught 5 more than Stan, and Stan has caught 13 less than 4 times as many as Adelwolfe has caught. How many Pokemon has Stan caught?

Let $X$ be the number of Pokemon Stan has caught.
Alex has caught 5 more than Stan, so Alex has caught $X + 5$.
Stan has caught 13 less than 4 times as many as Adelwolfe has caught, so Stan has caught $4X - 13$.
Together they have caught 339 Pokemon, so $X + 5 + 4X - 13 = 339$.
Combining like terms produces $5X + 5 = 339$.
Subtracting 5 from both sides produces $5X = 334$. Dividing both sides by 5 produces $X = 66.80$, so Stan has caught 66 Pokemon.
The answer is 66.
Form of Explanations: Chain-of-Thought

CoT supervise LLMs to plan out deduction steps and execute the computation

- Not good at planning complex solving procedure
  - (Valmeekam et al., 2022; Ribeiro et al., 2023)
- Not good at executing intensive computation
  - (Chen et al., 2022; Gao et al., 2023; Lyu et al., 2023)

Performance decreases with growing complexity

(Dziri et al., 2023)
Form of Explanations: Chain-of-Thought

CoT supervise LLMs to plan out deduction steps and execute the computation

- Not good at planning complex solving procedure
- Not good at executing intensive computation
- Good at interpreting the semantics in NL problems

(Valmeekam et al., 2022; Ribeiro et al., 2023)
(Chen et al., 2022; Gao et al., 2023; Lyu et al., 2023)

💡 We let LLMs focus on interpreting the NL problem
And offload the work of planning and executing to a symbolic solver
Each of 5 students—Hubert, Lori, Paul, Regina, and Sharon—will visit one of three cities—Montreal, Toronto, or Vancouver. Sharon visits a different city than Paul. Each student visits one of the cities with at least one of the other four students. [...] Q: Is it true that if any of the students visits Montreal, Lori visits Montreal?

LLMs parse the NL problem into declarative formal specifications

students=[Hubert, Lori, Paul, Regina, Sharon], cities=[Montreal, Toronto, Vancouver]
visits = Function(students, cities)
# Sharon visits a different city than Paul
 visits(Sharon) != visits(Paul)
# Each student visits one of the cities with at least one other student
 ForAll([s1], Exists([s2], And(s2 != s1, visits(s1) == visits(s2)))]
[...]
solve(Implies(Exists([s], visits(s) == Montreal), visits(L) == Montreal))  # Question

SMT solver generates and executes a plan using automated theorem proving
Q: Alex, Stan, and Adelwolfe are trying to catch them all, Pokemon that is. Together they have caught 339 Pokemon. Alex has caught 5 more than Stan, and Stan has caught 13 less than 4 times as many as Adelwolfe has caught. How many Pokemon has Stan caught?

Input

\[
\begin{align*}
\text{total}_\text{pokemon} &= 339 \\
\text{stan}_\text{pokemon} &= \text{Variable()} \\
\text{alex}_\text{pokemon} &= \text{stan}_\text{pokemon} + 5 \\
\text{stan}_\text{pokemon} &= 4 \times \text{adelwolfe}_\text{pokemon} - 13 \\
\text{total}_\text{pokemon} &= \text{alex}_\text{pokemon} + \\
\text{stan}_\text{pokemon} &= \text{adelwolfe}_\text{pokemon} \\
\text{result} &= \text{stan}_\text{pokemon} \\
\text{solve}(&\text{result})
\end{align*}
\]

Output

SAT Problem: Declarative

Program: Imperative

\[
\begin{align*}
\text{total}_\text{pokemon} &= 339 \\
\text{alex}_\text{pokemon} &= 5 \\
\text{stan}_\text{pokemon} &= 4 \\
\text{adelwolfe}_\text{pokemon} &= 13 \\
\text{stan}_\text{pokemon} &= (\text{total}_\text{pokemon} - \text{alex}_\text{pokemon} - \text{adelwolfe}_\text{pokemon} * \text{stan}_\text{pokemon}) / (1 - \text{stan}_\text{pokemon}) \\
\text{result} &= \text{stan}_\text{pokemon}
\end{align*}
\]

SMT Solver

\[
\text{Z3} \quad \text{result} = 147
\]

Python interpreter

\[
\text{result} = -94 \quad \text{X}
\]

(Chen et al., 2022; Gao et al., 2023; Lyu et al., 2023)
SAT-Aided Framework

Input → parse → SAT Specification → solve → Answer

Z3
SAT Problem

Define the meaning of some symbols in formulas, e.g., +, =

Formulas $\Phi$  
{\begin{align*}
x + y &= 3, \\
x - y &= 1
\end{align*}}

Query $Q$  
$x - 2$

Theory $T$  
Theory of integers, Theory of equality

Solver finds value assignment that can satisfy all formulas  
$x = 2, y = 1$

Evaluate the value of the query

SMT formulation is expressive

Allows handling a lot of reasoning tasks with unified formulation and solver using theory of linear arithmetic, theory of arrays, theory of strings, etc.
Each of 5 students—Hubert, Lori, Paul, Regina, and Sharon—will visit one of three cities—Montreal, Toronto, or Vancouver. Sharon visits a different city than Paul. [...]

students=[Hubert, Lori, Paul, Regina, Sharon],
cities=[Montreal, Toronto, Vancouver]
visits = Function(students, cities)
# Sharon visits a different city than Paul
visits(Sharon) != visits(Paul) [...]
Solving with Z3

(De Moura and Bjørner, 2008)

Input → SAT Specification → Z3 → Answer

Parse

SAT specification

students=[Hubert,Lori,Paul,Regina,Sharon],
cities=[Montreal,Toronto,Vancouver]
visits = Function(students, cities)
# Sharon visits a different city than Paul
visits(Sharon) != visits(Paul)
[...]

Extract formulas \( \Phi \) and query \( Q \)

Translate to actual python code that can be executed using Z3py

\[
\begin{align*}
\text{students} &= [\text{Hubert}, \text{Lori}, \text{Paul}, \text{Regina}, \text{Sharon}] \\
\text{translate} & \quad \rightarrow \\
\text{translate} & \quad \rightarrow \\
\text{students} &= \text{EnumSort}([“Hubert”,…,”Sharon”])
\end{align*}
\]
Experiments: Setup

Baselines

**CoT**: imperative NL explanations

**PAL**: imperative python programs

**SatLM**: declarative SAT specifications

Model

**gpt-3.5 (code-davinci-002)**

Tasks

**Arithmetic Reasoning**

**Logical Reasoning**

**Symbolic Reasoning**
(reason over arrays)

**Regex Synthesis**
(reason over strings)
Results: Arithmetic Reasoning

A subset of GSM requiring more complex backward reasoning or constraint solving

Problems from algebra textbooks
Results: Arithmetic Reasoning

A subset of GSM requiring more complex backward reasoning or constraint solving

Problems from algebra textbooks

SatLM substantially outperform baselines using imperative specifications on more challenging datasets GSM-Sys and ALGEBRA
Results: Logical Reasoning

Problems from law-school admission test, CoT on par with random guess

Requires commonsense reasoning
Results: Logical Reasoning

Using SMT solver, SatLM can handle problems requiring reasoning of high depth.

Problems from law-school admission test, CoT on par with random guess.

Requires commonsense reasoning.
Benefits of using SMT Solver

SMT solver can spot semantic errors in the specification

- **Unsatisfiable**
  - Conflicting formulas
    - \( y = x + 1 \)
    - \( z = x - 1 \)
    - \( x = y + 1 \)

- **Ambiguous**
  - Multiple feasible solutions
    - \( x = y + 1 \)
    - \( x > 0 \)

- **Exception**
  - Syntax errors, time-out, etc.
  - Program interpreters typically can only spot this type of errors
Selective Prediction with SMT Solver

SMT solver can spot semantic errors in the specification

When SatLM successfully returns an answer, it is more likely to be correct

SatLM answers fewer questions but achieves higher accuracy.

Selective Accuracy

Correct answer / coverage

PAL: 45.2
SatLM: 77.1
Q: Farmer Brown has 60 animals on his farm, all either chickens or cows. He has twice as many chickens as cows. How many legs do the animals have, all together?

animals_total = 60
animals_chickens = Variable()
animals_cows = Variable()
animals_chickens = animals_cows * 2
animals_total = animals_chickens + animals_cows
legs_chickens = animals_chickens * 2
legs_cows = animals_cows * 4
legs_total = legs_chickens + legs_cows

LLMs can perform commonsense reasoning while parsing
Use SAT specification as explanations for a diverse of reasoning tasks

Offload planning and execution to SMT solver
Steering Textual Reasoning with Explanations

Post-Hoc Intervene

Input → Prediction

explain ↓

XY++ NeurIPS 22
XY++ ACL 22
XY++ EMNLP 21
PS*, JF*, XY++ EACL 23

Teach with Explanations

Input → Prediction

Demo

XY++ NeurIPS 23
XY++ EMNLP 23
XY++ ACL Findings 23
ZS, XY++ Arxiv 23 (in sub.)
Steering Textual Reasoning with Explanations

Post-Hoc Intervene

Input → Prediction

explain

Teach with Explanations

Input → Prediction

Demo

Empirical analysis on what makes explanations effective

XY++ NeurIPS 22
XY++ ACL 22
XY++ EMNLP 21
PS*, JF*, XY++ EACL 23

XY++ NeurIPS 23
XY++ EMNLP 23
XY++ ACL Findings 23
ZS, XY++ Arxiv 23 (in sub.)
Steering Textual Reasoning with Explanations

**Post-Hoc Intervene**

Input → Prediction

-explain-

Use explanations to investigate reasoning process and calibrate model predictions post-hoc

**Teach with Explanations**

Input → Prediction

Demo

Construct effective explanations written in the right style and in the right form
Questions

Post-Hoc Intervene

Input → Prediction

explain

XY++ NeurIPS 22
XY++ ACL 22
XY++ EMNLP 21
PS*, JF*, XY++ EACL 23

Teach with Explanations

Input → Prediction

Demo

XY++ NeurIPS 23
XY++ EMNLP 23
XY++ ACL Findings 23
ZS, XY++ Arxiv 23 (in sub.)