Interpreting Arithmetic Reasoning in Large Language Models using Game-Theoretic Interactions

NeurIPS 2025 Leilei Wen

Tongji University, Shanghai, China

Oct. 2025

Introduction

- ✓ Recently, LLMs have made significant advancements in arithmetic reasoning. However, the internal mechanism of how LLMs solve arithmetic problems remains unclear.

Related Work

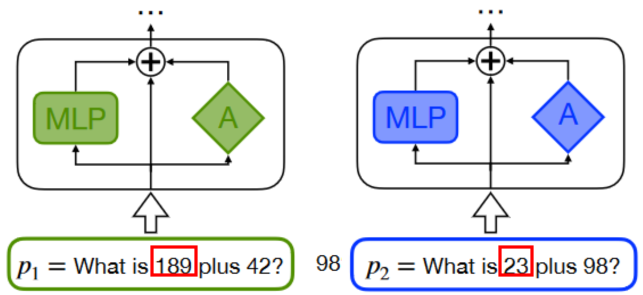
Identify neurons

FFNv	origin	attn transform
12_{4072}	[rd, quarters, PO, Constraint, ran, avas]	[**III**, **three**, **Three**, **3**, **triple**]
11_{2258}	[enz, Trace, lis, vid, suite, HT, ung, icano]	[**XV**, **fifth**, **Fif**, avas, **Five**, **five**, abase, **fif**]

Without theoretical support

The interaction has been proven to be faithful explanations by a series of theoretical guarantees.

Evaluate the influence of each input variable



Without considering interactions between input variables

Analyze how LLMs encode different interactions during the forward propagation process.

Interactions

✓ Prior work has shown that a neural network's output score can be decomposed as the sum of the effects of symbolic interaction concepts.

bird head

bird

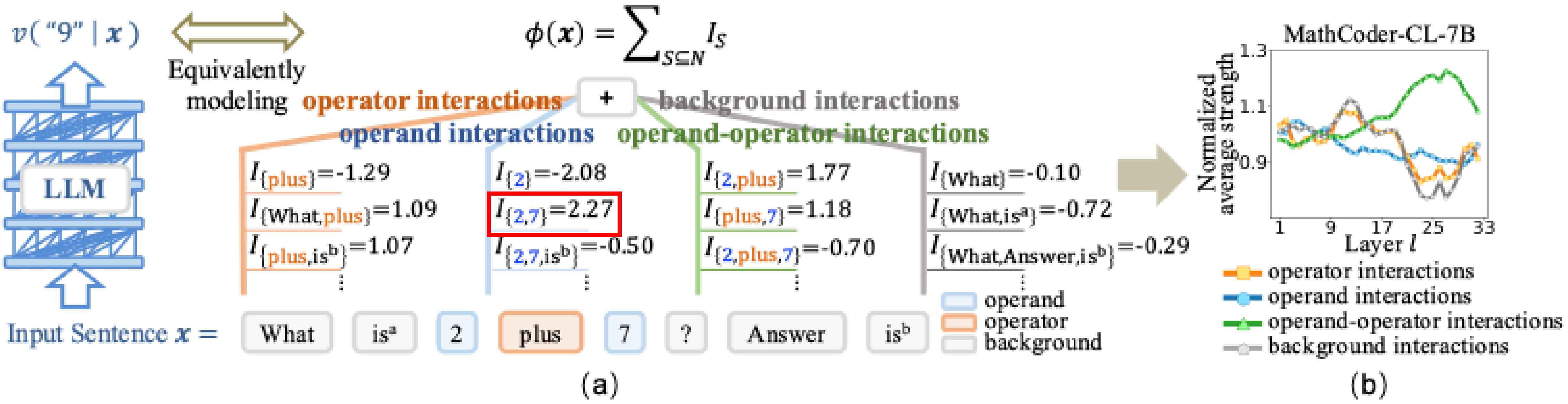
$$v(input) = \sum_{\text{interaction S}} \text{effect of S}$$

He is a green hand

beginner

Interactions among input units

Model



✓ **Defining** different types of interactions

$$\Omega^{\text{opd}} = \{S \mid \exists\, \boldsymbol{x}^{\text{opd}} \in S \wedge \nexists\, \boldsymbol{x}^{\text{opr}} \in S\}.$$

$$\Omega^{\text{opr}} = \{S \mid \exists\, \boldsymbol{x}^{\text{opr}} \in S \wedge \nexists\, \boldsymbol{x}^{\text{opd}} \in S\}.$$

$$\Omega^{\text{opd-opr}} = \{S \mid \exists\, \boldsymbol{x}^{\text{opd}} \in S \wedge \exists\, \boldsymbol{x}^{\text{opr}} \in S\}.$$

$$\Omega^{\text{bg}} = \{S \mid \nexists\, \boldsymbol{x}^{\text{opd}} \in S \wedge \nexists\, \boldsymbol{x}^{\text{opr}} \in S\}.$$

Model

✓ **Quantifying** interactions encoded by an LLM in intermediate layers

$$\forall T \subseteq N, v^{(l)}(\boldsymbol{x}_T) = \cos\left(f^{(l)}(\boldsymbol{x}_T), f^{(l)}(\boldsymbol{x}_N)\right) = \frac{\left(f^{(l)}(\boldsymbol{x}_N)\right)^T \cdot f^{(l)}(\boldsymbol{x}_T)}{\left\|f^{(l)}(\boldsymbol{x}_N)\right\|_2 \cdot \left\|f^{(l)}(\boldsymbol{x}_T)\right\|_2}$$

$$\forall S \subseteq N, S \neq \emptyset, I_S = \sum_{S' \subseteq S} (-1)^{|S|-|S'|} \cdot v(\boldsymbol{x}_{S'}), \text{Equation (2)}$$

✓ **Quantifying** different interactions

$$R^{(l)}(\Omega^{type}) = \frac{\mathbb{E}_{S \in \Omega^{type}} \left|I^{(l)}(S)\right|}{Z^{(l)}}, \Omega^{type} \in \{\Omega^{opd}, \Omega^{opr}, \Omega^{opd-opr}, \Omega^{bg}\}$$

different types

$$\kappa_m^{(l)} = \frac{\mathbb{E}_{|S|=m} \left|I^{(l)}(S)\right|}{Z^{(l)}}, m \in \{1,2 \ldots, n\}$$

different orders

normalization term: $Z = \mathbb{E}_{S \subseteq N} |I(S)|$

Comparative studies

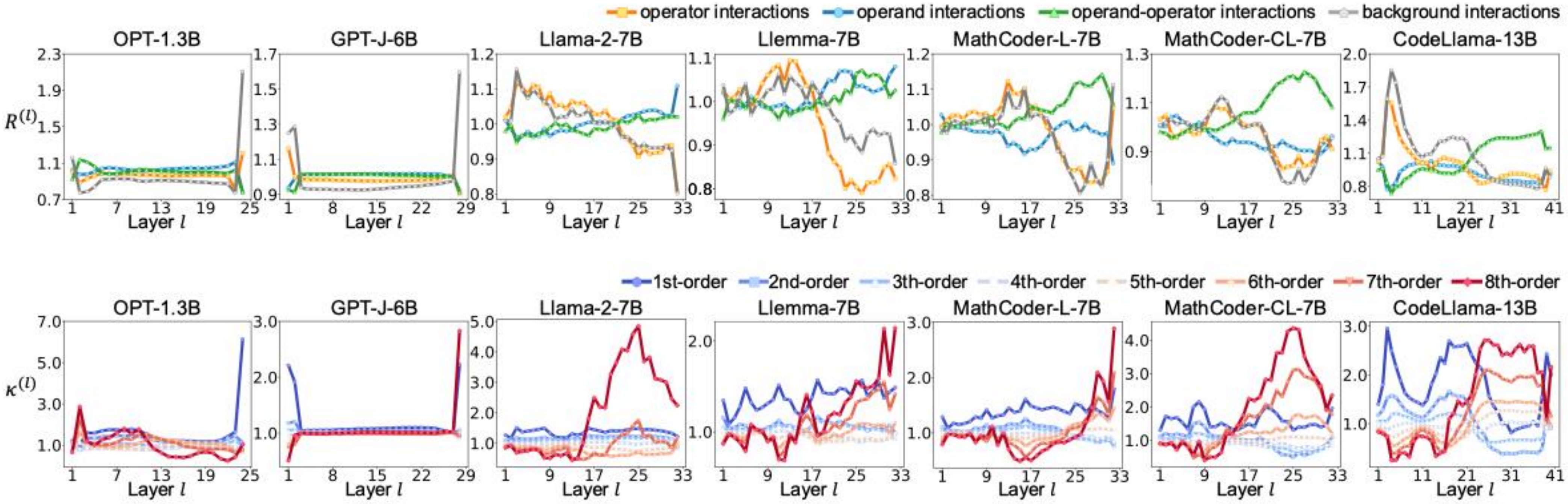
- ✓ We use interactions to analyze seven LLMs for arithmetic reasoning.
- ✓ We conduct experiments on a set of arithmetic problems hand-crafted by humans, including 6 templates for one-operator two-operand queries and 29 templates for two-operator three-operand queries.

Table 1: Overall accuracy (%) of different LLMs on arithmetic queries.

Model	**1-opr**	**2-opr**
OPT-1.3B	3.2	1.7
GPT-J-6B	14.7	5.8
Llama-2-7B	65.1	10.1
Llemma-7B	75.1	15.3
MathCoder-L-7B	74.0	8.2
MathCoder-CL-7B	62.6	9.3
CodeLlama-13B	71.1	15.0
OPT-1.3B Fine-tuned	83.6	69.7

Comparative studies

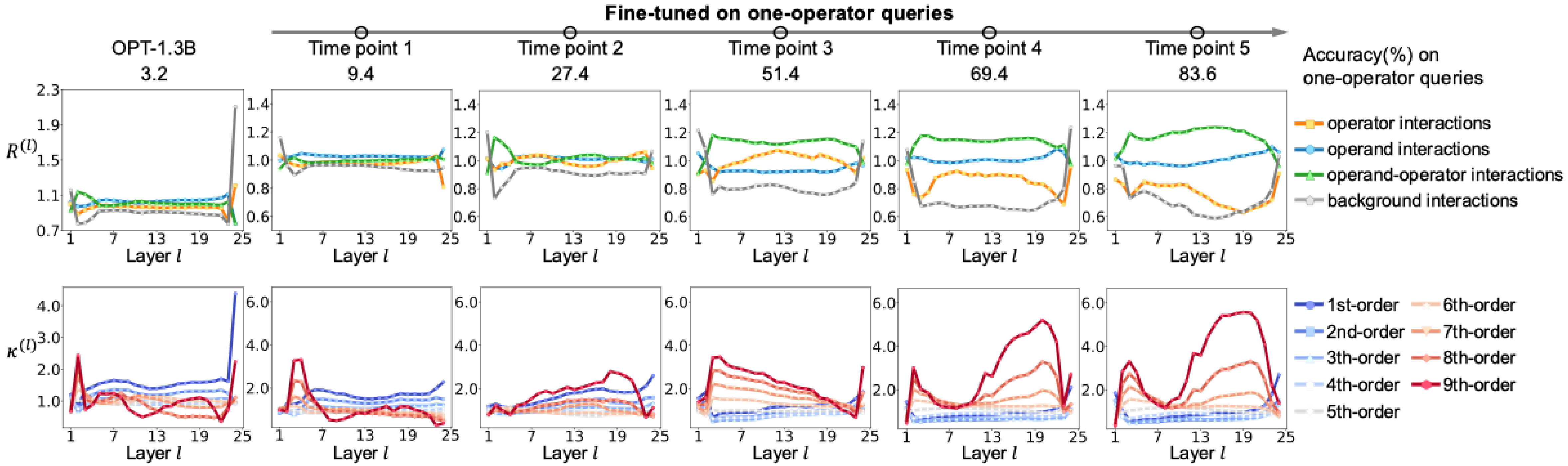
Insight 1： The internal mechanism of LLMs for solving simple one-operator arithmetic problems is their capability to encode2 operand-operator interactions and high-order interactions from input samples.



Comparative studies

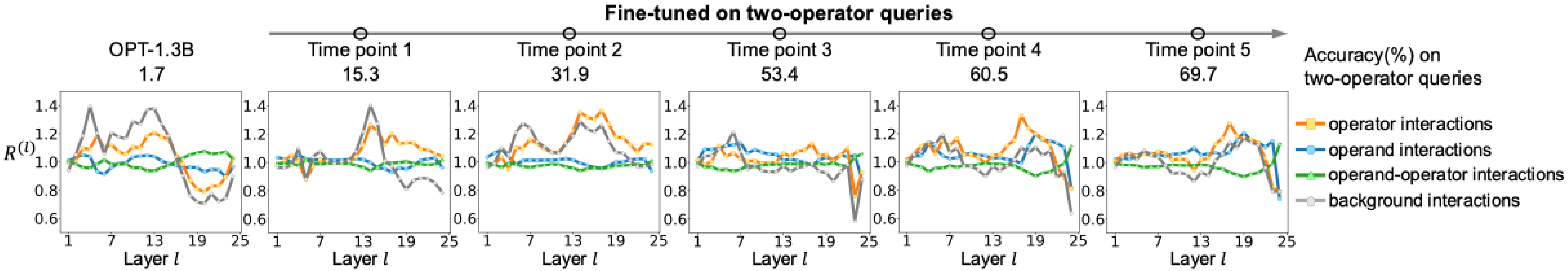
✓ We further explore how an LLM learns to solve arithmetic problems.

✓ We investigate how an LLM encodes different interactions when trained on arithmetic problem data.



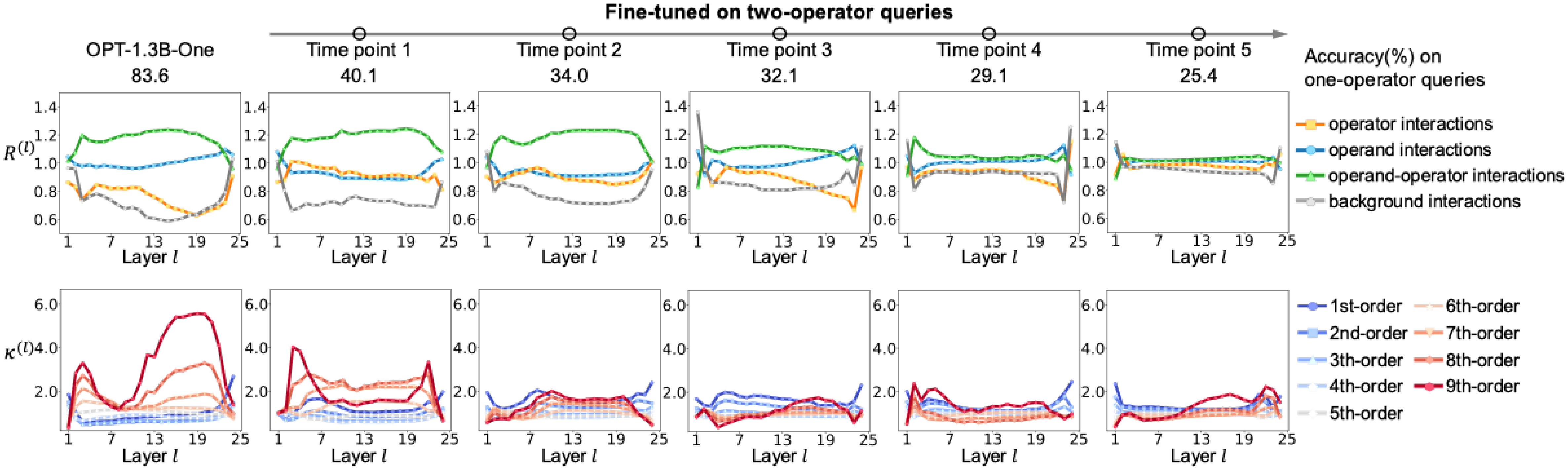
Comparative studies

Insight 2: The internal mechanism of LLMs for solving relatively complex two-operator arithmetic problems is their capability to encode operator interactions and operand interactions from input samples.



Comparative studies

Insight 3: We explain the task-specific nature of the LoRA method from the perspective of interactions.



Summary

In this paper, we use interactions to provide a deep understanding of the internal mechanism of LLMs for arithmetic reasoning.

- ✓ The internal mechanism of LLMs for solving simple one-operator arithmetic problems is their capability to encode operand-operator interactions and high-order interactions.
- ✓ The internal mechanism of LLMs for solving relatively complex two-operator arithmetic problems is their capability to encode operator interactions and operand interactions.
- ✓ We also explain the task-specific nature of the LoRA method from the perspective of interactions.

Thank you！

NeurIPS 2025 Leilei Wen

Tongji University, Shanghai, China

Oct. 2025