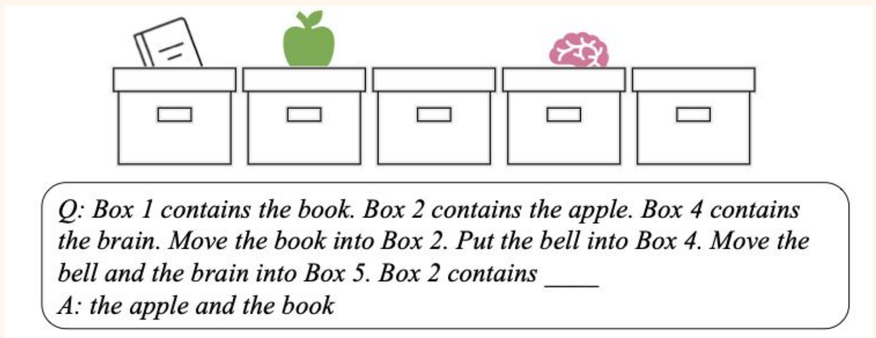
Mechanistic Understanding of Entity Tracking in Natural Language involving Multiple Operations

Zilu (Peter) Tang, Qiao Zhao, Gabriel Franco, Geneva Yang, Angelos Poulis, Derry Wijaya, Aaron Mueller, Sebastian Schuster, Najoung Kim

Contact us with ideas and collab!
zilutang@bu.edu
zhaoqiao@bu.edu

Entity Tracking: Prerequisite for Reasoning



- Prerequisite capability for many tasks like game, reasoning, code variable tracking
- Synthetically generated, verifiable results[1]
- no world knowledge needed

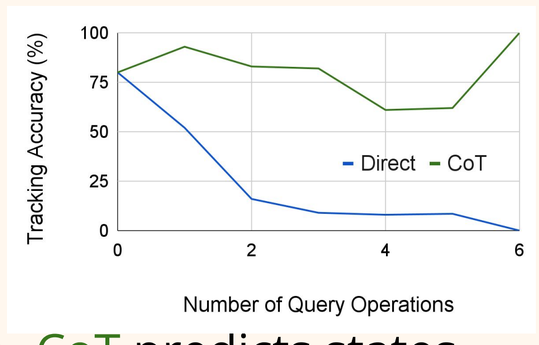
Tracking accuracy improves with LM size & with CoT

- GPT2, Flan-T5-XL (3B)
- ~0% accuracy (direct)
- ~100% (finetune)
- **Small LM cannot perform out-of-the-box, but task is learnable via finetuning**

- Gemma 7B
- Pretrained, 1-shot
- ~40% (direct)
- ~80% (CoT)
- **Intermediate size LMs can perform task with the help of CoT**

- Codellama 13B, Llama 3.1 405B
- Pretrained, 2-shot
- ~95% (direct)
- **Larger LMs can perform task out-of-the-box**

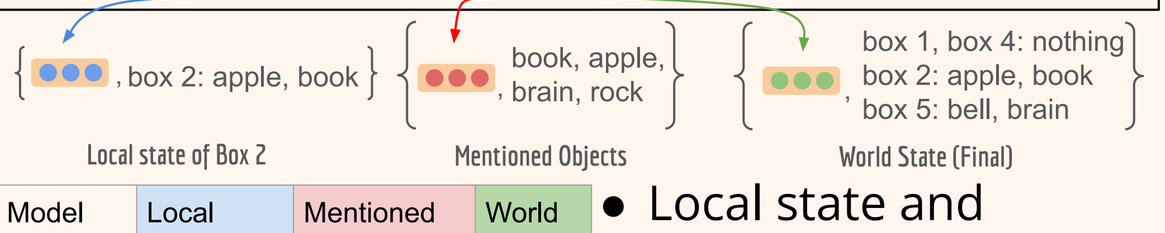
LM sizes



- CoT predicts states sequentially, improves Gemma-7B over directly predicting the final state

Probing: LMs Dynamically Retrieve Answer

Box 1 contains the book, Box 2 contains the apple, Box 4 contains the brain. Move the book into Box 2. Put the bell into Box 4. Move the bell and the brain into Box 5. Box 2 contains the apple and the book

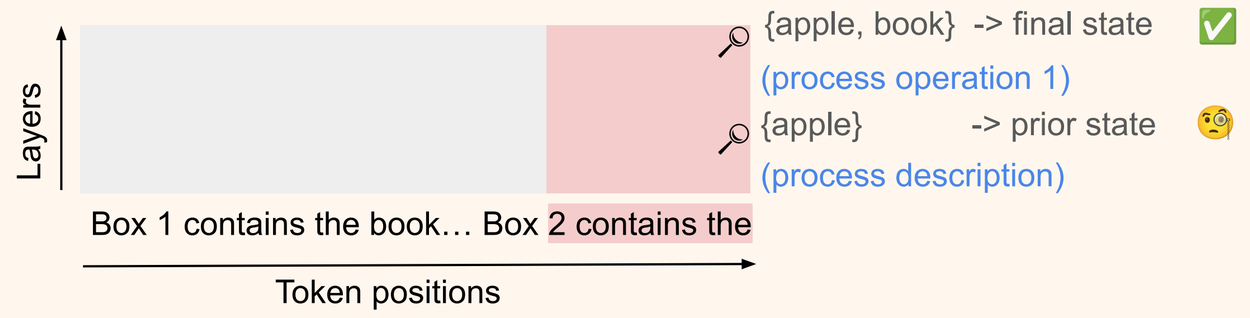


Model	Local State Acc	Mentioned Objects Acc	World State
CodeLlama-13B	87.8%	89.1%	22.5%
Llama-3.1-70B	97.1%	88.5%	22.6%

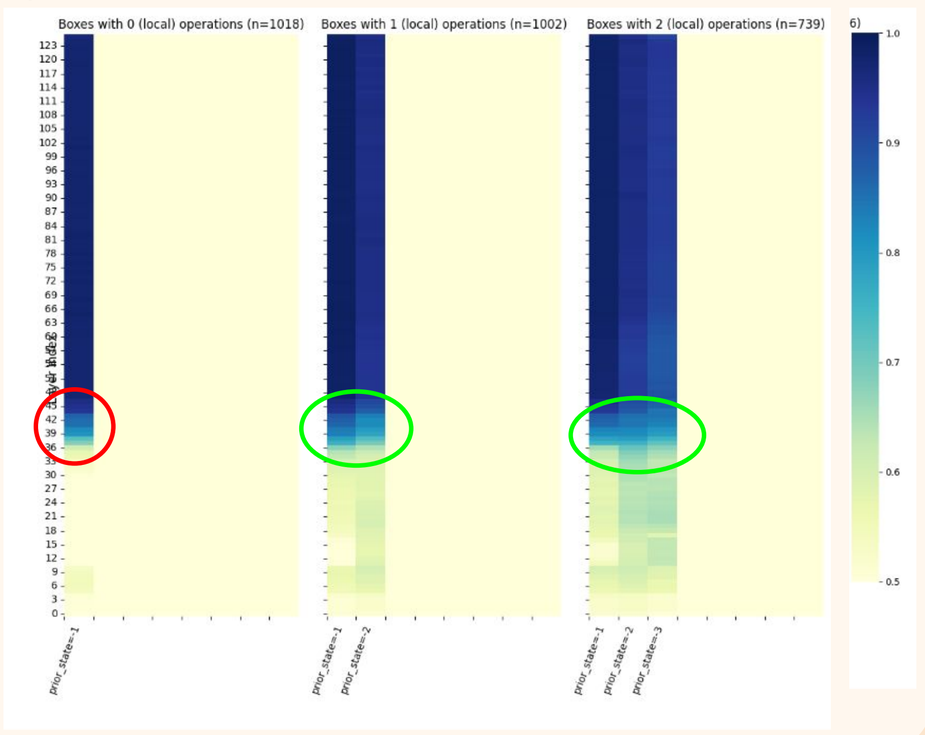
- Local state and mentioned objects are decodable, while global state is **not** decodable at the final layer

LMs Process Multiple Operations In Parallel

- If LMs process multi-operations sequentially across the layers, *prior states* should be decodable in earlier layers.
- If so, we expect *{apple}* to be decodable earlier below:



- Final state probing accuracy increases sharply
- Prior state probing accuracy emerges around same layers
- Processing is **in parallel**!

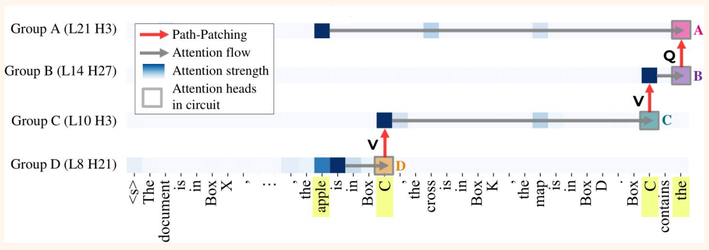


Previously Identified Circuit is not Adequate

- Prior work[2] found no-op circuit w. iterative path patching w/ metric:

$$(p^{\text{obj(s)}}_{\text{patch}} - p^{\text{obj(s)}}_{\text{clean}})/p^{\text{obj(s)}}_{\text{clean}}$$

- We find such a no-op circuit on gemma-2-2b but it tracks only the last mentioned query phrase object (obj1). (Table →)
- 1-put circuit reveals different components are responsible for obj0 (from description) and obj1 (from put)



(↑) Previous circuit (with four groups of heads) found model retrieve label through the *position* of the object

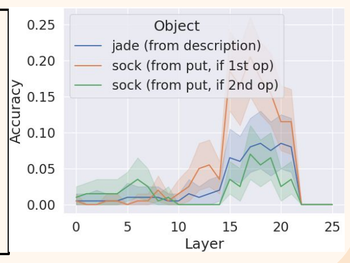
Mode/objective	Argmax Acc	topK Acc (obj 0)	topK Acc (obj 1)
Full Model	0.71	0.52	0.65
1put, obj0+1	0.67	0.70	0.18
1put, obj0	0.64	0.83	0.13
1put, obj1	0.46	0.20	0.54
No-op (prior)[2]	0.98	0.09	0.98

(↑) we find different circuit for obj0/1 by varying which object(s) we use as our label.

Activation patching reveals similar "put" mechanism

- Desiderata-based patching[2] (residual stream) indicates similarity in layers between put and description, and they share retrieval circuit through position (↓)

- Counterfactual: Cat is in Box 3, ball is in Box 1, … Put pear in Box 1. Put hat in Box 2. Box 1 contains the ball, pear
- Clean: Book is in Box 1, jade is in Box 2,… Put sock in Box 2. Put cake in Box 1. Box 1 contains the book, cake
- Label (if encoding position info): jade (obj0), sock (obj1)



Future Plans: Single & Multi-Operation Mechanisms

- Need to patch at multiple prediction positions?
- Positional information stored in the same subspaces?
- Counterfactuals for individual operations? (e.g. put)
- Can we find mechanistic advantages of CoT? How do we analyze CoT circuits?
- Is multi-op circuit compositions of single-op circuits?
- How do LMs encode different operations internally, e.g, local removals v.s. global removals.

Citations & Acknowledgements

1. Kim and Schuster. Entity Tracking in Large Language Models. 2023
2. Prakash et al. Fine-tuning Enhances Existing Mechanisms: a Case Study on Entity Tracking 2024
3. Li et al. Emergent world representations: Exploring a Sequence Model Trained on a Synthetic Task (OthelloGPT) 2023

This project is supported by MassMutual.