Language Models use Lookbacks to Track Beliefs

Anonymous Author(s)
Affiliation
Address
email

Abstract

How do language models (LMs) represent characters' beliefs, especially when those beliefs differ from reality? We analyze Llama-3-70B-Instruct on Theory of Mind (ToM) reasoning tasks. Using a dataset of short stories where characters act on objects with partial visibility, we uncover a pervasive algorithmic pattern that we call the *lookback mechanism*. This mechanism allows LM to recall information when needed: it binds character–object–state triples via Ordering IDs (OIs) and retrieves them through pointer–address dereferencing in the residual stream. We identify three key lookbacks: binding, answer, and visibility. Our work provides insights into the LM's belief tracking mechanisms, taking a step toward reverse-engineering ToM reasoning in LMs.

1 Introduction

Theory of Mind (ToM), the ability to infer others' mental states, is a cornerstone of human social intelligence [Premack and Woodruff, 1978]. Recent studies show LMs sometimes succeed at ToM tasks [Street et al., 2024, Kosinski, 2024] but their internal mechanisms remain opaque. Behavioral evaluations reveal what models can do, but not how they implement belief tracking [Hu et al., 2025].

We ask: *How do LMs internally represent and update characters' beliefs?* Inspired by canonical ToM tests such as Sally–Anne [Baron-Cohen et al., 1985], we build CausalToM, a dataset of short stories where two characters act on objects, sometimes with visibility constraints. An LM must answer questions about each character's belief regarding object states.

Our main discovery is the lookback mechanism, a recurring computational pattern resembling pointer dereferencing in C programming inside a transformer. Lookbacks enable belief tracking by binding events in earlier tokens and selectively retrieving them later when answering belief queries.

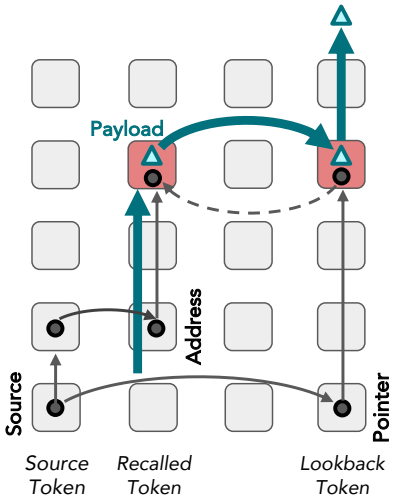


Figure 1: **The lookback mechanism** enables conditional reasoning by duplicating a source token into a *pointer* and an *address* via attention. The address carries a *payload* in the residual stream, which the model retrieves by dereferencing the pointer. Solid lines show information flow, while the dotted line marks the pointer's attention back to the address.

2 The Lookback Mechanism

In a lookback mechanism, information from a source token is copied into two instances: 1) an **address copy**

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

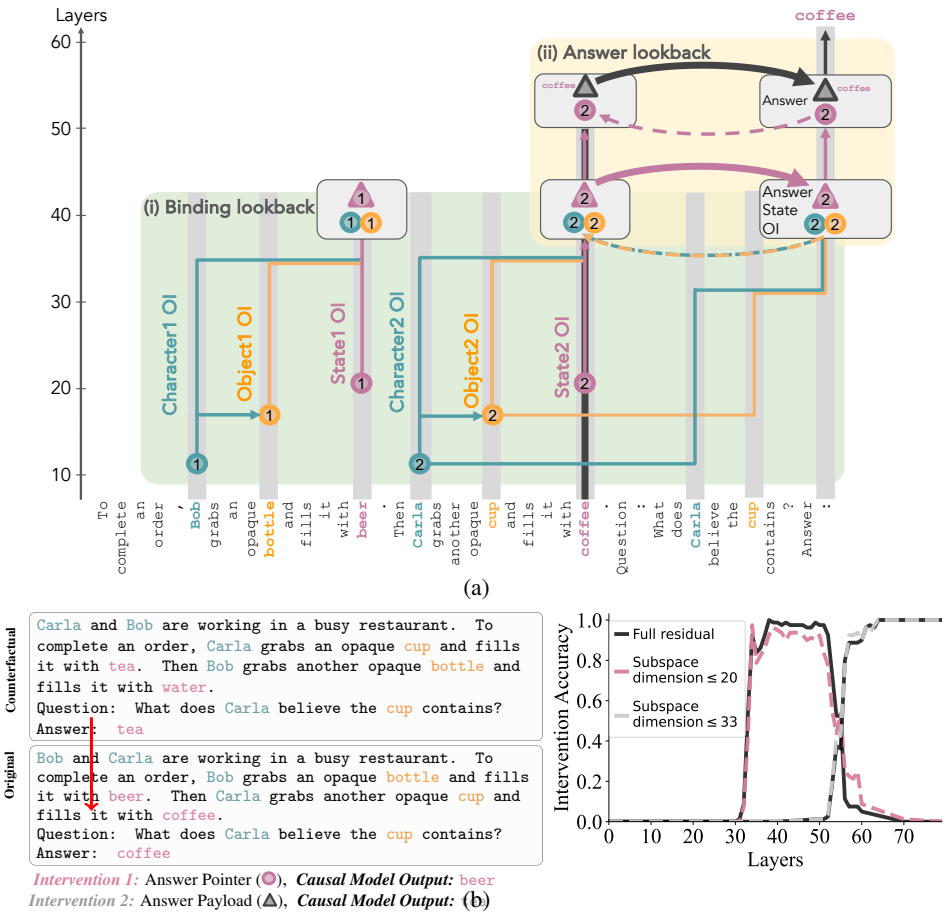


Figure 2: **Belief Tracking in Language Models:** We task the LM with tracking the beliefs of two characters that manipulate the states of two objects. We hypothesize that the LM solves this task by implementing a causal model with two lookback mechanisms (2a). To support our hypothesis, we conduct a causal analysis where we measure whether interventions on a high-level causal model produce the same output as equivalent interventions on the LM. For instance in 2b, we show results for an experiment distinguishing the pointer and payload in the answer lookback. Refer to Appendix D for detailed full causal model.

(a) **Belief Tracking with *No Visibility* between Characters:** Our hypothesized causal model for this kind of story has two lookbacks that operate on ordering IDs (OIs) that encode whether a token appears first or second. In the **binding lookback (i)**, the LM first represents the two events in the story by binding together each character-object-state triple in the residual stream of the state token. When questioned about a particular character and object, the LM looks back to the corresponding triple and retrieves an OI to that state token. Notice that in this lookback, that payload is later used as a pointer, i.e., what a C programmer would call a double pointer. In the **answer lookback (ii)**, the LM dereferences the pointer to the answer token to generate the correct answer. Color indicates the information content, while shape indicates the role of that information in lookback (see Fig. 1), e.g., the state OI is a payload (▲) in the binding lookback and a pointer/address (●) in the answer lookback.

(b) **Answer Lookback Pointer and Payload**: To test our hypothesized causal model, we run the LM on pairs of slightly different stories, and then intervene by patching a specific representation state from the counterfactual run to the original run, observing any change in the output. The causal model predicts that if we alter the "answer payload ▲" of the original to instead take the value of the counterfactual answer payload, the output should change from coffee to tea; the gray curve in the line plot shows this does occur with $p \approx 1.0$ when patching states at the ":" token beyond layer 56, providing evidence that the answer payload resides in those states. On the other hand the causal model predicts that taking the counterfactual "answer pointer ●" would change the original run output from coffee to beer—a new output that matches *neither* the original nor the counterfactual!—and we do see this surprising effect, again with $p \approx 1.0$, when patching layers between 34 and 52, providing strong evidence that the answer pointer is encoded at those layers. Collected over $N = 80$ samples, these measurements suggest the Answer Lookback occurs between layers 52 and 56. Furthermore the representations of the causal variables are small: the interventions can be localized even further to subspaces of dimension 33 (payload) or 20 (pointer), tiny portions of the 8192-dimensional state space.

stored with a **payload** in the residual stream of the recalled token, and 2) a **pointer copy** placed in the residual stream of a later lookback token.

When needed, attention connects pointer and address (via a QK circuit), allowing the payload to be retrieved into the lookback token, as shown in Fig.1. This enables conditional reasoning without passing information directly through every intermediate token.

We identify three lookbacks that used for belief tracking:

1. **Binding lookback**: binds character and object OIs to state tokens.
2. **Answer lookback**: retrieves the correct state token when queried.
3. **Visibility lookback**: integrates information when one character can observe another.

3 Dataset and Methodology

Existing ToM datasets evaluate behavior but lack counterfactuals needed for causal analysis. We construct **CausalToM**, consisting of two characters each acting on one object, e.g.: *"Carla grabs an opaque cup and fills it with coffee. Bob grabs a bottle and fills it with beer."* We then ask: *"What does Carla believe the cup contains?"* Refer to Appendix A & B for the full prompt and additional dataset details. Our experiments analyze the Llama-3-70B-Instruct model in half-precision, using NNsight [Fiotto-Kaufman et al., 2025].

We analyze model internals using *interchange interventions*: running the LM on paired original and counterfactual stories and swapping specific residual stream activations. If replacing a representation changes the answer as predicted by a causal model, we infer alignment between the hypothesized variable and the LM's computation. Refer to Appendix C for more details.

4 Belief Tracking via Ordering IDs and Lookback Mechanisms

4.1 Ordering IDs

The LM assigns an Ordering ID (OI; [Dai et al., 2024]) to the character, object, and state tokens. These OIs, encoded in a low-rank subspace of the internal activation, serve as a reference that indicates whether an entity is the first or second of its type independent of its token value. For example, in Fig.2a, **Bob** is assigned the first character OI, while **Carla** receives the second.

4.2 Binding Lookback

Character and object OIs are stored as addresses alongside the state OI payload within the corresponding state token. For instance, in A, the character and object OIs of **Bob** and **bottle** are copied into the residual stream of the **beer** token, along with its own state OI. When they appear in the question sentence, the corresponding pointer copies are carried forward to the final token, where attention dereferences them to retrieve the answer state OI , as shown in Fig.2a. Refer to Appendix E for more details on the causal intervention experiments and results.

4.3 Answer Lookback

In the second lookback, the answer state OI acts as the source: its address copy is stored in the state token's residual stream, while its pointer, propagated through the binding lookback, is dereferenced at the final token to retrieve the correct state, as shown in Fig.2a. For example, in A, the state OI of the **beer** token is dereferenced to recover its token value, which is then predicted as the final answer. This is verified using causal intervention experiments shown in Fig.2b. Refer to Appendix F for more details on the causal intervention experiments and results.

4.4 Visibility Lookback

When the input prompt specifies that one character observes another, the LM generates a *visibility ID* at the visibility sentence 3. Its address copy is retained on the corresponding tokens, while its pointer copy is propagated to the question sentence, as illustrated in Fig. 3. In later layers, this pointer is

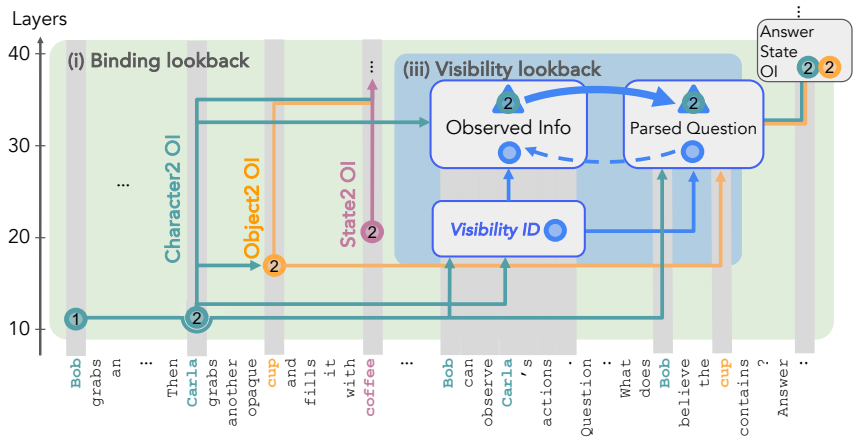


Figure 3: **Visbility Lookback** When one character (the observing character) can see another (the observed character), the LM assigns a visibility ID ( ) to the visibility sentence (where this relation is defined). An address copy of this visibility ID remains in the visibility sentence's residual stream. A pointer copy of the visibility ID is transferred to the subsequent tokens' residual stream (lookback tokens). During processing, the model dereferences this pointer through a QK-circuit, bringing forward the payload ( ). Based on initial evidence, this payload contains the observed character's OI( ). Refer to Appendix J for more details. This mechanism allows the model to incorporate the observed character's knowledge into the observing character's belief state, enabling more complex belief reasoning.

dereferenced, allowing the observing character's awareness to incorporate the observed character's OI, thereby enabling reasoning about nested beliefs. Refer to Appendix G for more details on the causal intervention experiments and results.

5 Discussion and Future work

Our analysis, as described in Appendix M & L, reveals that the identified mechanism is not idiosyncratic to a single model or dataset, but rather appears to be a general computation. We identified the same mechanism in Llama-3.1-405B-Instruct and other benchmarks such as BigToM [Gandhi et al., 2024]. Importantly, the presence of structured operations such as binding, answer, and visibility lookbacks indicates that the model is not simply memorizing training examples. Instead, it has learned a genuine algorithmic strategy for ToM reasoning, in which abstract information is generated, stored, and later retrieved in a structured way. Particularly striking is the finding that the model systematically converts vital tokens, characters, objects, and states, into abstract ordering identifiers and then employs lookbacks to dereference these identifiers during reasoning. This shows that transformers can construct intermediate abstractions that go beyond surface text, enabling them to solve tasks requiring complex logical reasoning.

We see several directions as promising future works. First, we plan to investigate the ubiquity of ordering information: do abstract OIs arise for all tokens, or only a select subset that the model deems relevant for reasoning? Second, given our evidence across models and datasets, we hypothesize that lookback is a generic mechanism that may underpin many reasoning tasks beyond ToM. Exploring this possibility could help unify mechanistic explanations of diverse LM capabilities. Finally, extending our analysis to scenarios with more than two interacting characters would test whether lookbacks scale to richer, more naturalistic conversations, and whether new variants of the mechanism emerge in such settings. Together, these directions point toward building a broader catalog of algorithmic motifs that constitute the foundations of reasoning in LMs.

References

S. Baron-Cohen, A. M. Leslie, and U. Frith. Does the autistic child have a "theory of mind"? *Cognition*, 21(1):37–46, 1985.

Q. Dai, B. Heinzerling, and K. Inui. Representational analysis of binding in language models. In Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17468–17493, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.967. URL https://aclanthology.org/2024.emnlp-main.967/.

X. Davies, M. Nadeau, N. Prakash, T. R. Shaham, and D. Bau. Discovering variable binding circuitry with desiderata, 2023. URL https://arxiv.org/abs/2307.03637.

N. De Cao, M. S. Schlichtkrull, W. Aziz, and I. Titov. How do decisions emerge across layers in neural models? interpretation with differentiable masking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3243–3255, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.262. URL https://aclanthology.org/2020.emnlp-main.262.

J. F. Fiotto-Kaufman, A. R. Loftus, E. Todd, J. Brinkmann, K. Pal, D. Troitskii, M. Ripa, A. Belfki, C. Rager, C. Juang, A. Mueller, S. Marks, A. S. Sharma, F. Lucchetti, N. Prakash, C. E. Brodley, A. Guha, J. Bell, B. C. Wallace, and D. Bau. NNsight and NDIF: Democratizing access to open-weight foundation model internals. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=MxbEiFRf39.

K. Gandhi, J.-P. Fränken, T. Gerstenberg, and N. Goodman. Understanding social reasoning in language models with language models. *Advances in Neural Information Processing Systems*, 36, 2024.

A. Geiger, K. Richardson, and C. Potts. Neural natural language inference models partially embed theories of lexical entailment and negation. In A. Alishahi, Y. Belinkov, G. Chrupała, D. Hupkes, Y. Pinter, and H. Sajjad, editors, *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.16. URL https://aclanthology.org/2020.blackboxnlp-1.16.

A. Geiger, D. Ibeling, A. Zur, M. Chaudhary, S. Chauhan, J. Huang, A. Arora, Z. Wu, N. Goodman, C. Potts, and T. Icard. Causal abstraction: A theoretical foundation for mechanistic interpretability, 2024. URL https://arxiv.org/abs/2301.04709.

M. Geva, J. Bastings, K. Filippova, and A. Globerson. Dissecting recall of factual associations in auto-regressive language models, 2023. URL https://arxiv.org/abs/2304.14767.

J. Hu, F. Sosa, and T. Ullman. Re-evaluating theory of mind evaluation in large language models. *arXiv preprint arXiv:2502.21098*, 2025.

M. Kosinski. Evaluating large language models in theory of mind tasks. *Proceedings of the National Academy of Sciences*, 121(45), Oct. 2024. ISSN 1091-6490. doi: 10.1073/pnas.2405460121. URL http://dx.doi.org/10.1073/pnas.2405460121.

A. Mueller, J. Brinkmann, M. Li, S. Marks, K. Pal, N. Prakash, C. Rager, A. Sankaranarayanan, A. S. Sharma, J. Sun, E. Todd, D. Bau, and Y. Belinkov. The quest for the right mediator: A history, survey, and theoretical grounding of causal interpretability, 2024. URL https://arxiv.org/abs/2408.01416.

N. Prakash, T. R. Shaham, T. Haklay, Y. Belinkov, and D. Bau. Fine-tuning enhances existing mechanisms: A case study on entity tracking. In *Proceedings of the 2024 International Conference on Learning Representations*, 2024. arXiv:2402.14811.

D. Premack and G. Woodruff. Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1(4):515–526, 1978. doi: 10.1017/S0140525X00076512.

P. Smolensky. Neural and conceptual interpretation of PDP models. In J. L. McClelland, D. E. Rumelhart, and the PDP Research Group, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Psychological and Biological Models*, volume 2, pages 390–431. MIT Press, 1986.

W. Street, J. O. Siy, G. Keeling, A. Baranes, B. Barnett, M. McKibben, T. Kanyere, A. Lentz, B. A. y Arcas, and R. I. M. Dunbar. Llms achieve adult human performance on higher-order theory of mind tasks, 2024. URL https://arxiv.org/abs/2405.18870.

J. Vig, S. Gehrmann, Y. Belinkov, S. Qian, D. Nevo, Y. Singer, and S. Shieber. Investigating gender bias in language models using causal mediation analysis. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12388–12401. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/92650b2e92217715fe312e6fa7b90d82-Paper.pdf.

References

S. Baron-Cohen, A. M. Leslie, and U. Frith. Does the autistic child have a "theory of mind"? *Cognition*, 21(1):37–46, 1985.

Q. Dai, B. Heinzerling, and K. Inui. Representational analysis of binding in language models. In Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17468–17493, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.967. URL https://aclanthology.org/2024.emnlp-main.967/.

X. Davies, M. Nadeau, N. Prakash, T. R. Shaham, and D. Bau. Discovering variable binding circuitry with desiderata, 2023. URL https://arxiv.org/abs/2307.03637.

N. De Cao, M. S. Schlichtkrull, W. Aziz, and I. Titov. How do decisions emerge across layers in neural models? interpretation with differentiable masking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3243–3255, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.262. URL https://aclanthology.org/2020.emnlp-main.262.

J. F. Fiotto-Kaufman, A. R. Loftus, E. Todd, J. Brinkmann, K. Pal, D. Troitskii, M. Ripa, A. Belfki, C. Rager, C. Juang, A. Mueller, S. Marks, A. S. Sharma, F. Lucchetti, N. Prakash, C. E. Brodley, A. Guha, J. Bell, B. C. Wallace, and D. Bau. NNsight and NDIF: Democratizing access to open-weight foundation model internals. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=MxbEiFRf39.

K. Gandhi, J.-P. Fränken, T. Gerstenberg, and N. Goodman. Understanding social reasoning in language models with language models. *Advances in Neural Information Processing Systems*, 36, 2024.

A. Geiger, K. Richardson, and C. Potts. Neural natural language inference models partially embed theories of lexical entailment and negation. In A. Alishahi, Y. Belinkov, G. Chrupała, D. Hupkes, Y. Pinter, and H. Sajjad, editors, *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.16. URL https://aclanthology.org/2020.blackboxnlp-1.16.

A. Geiger, D. Ibeling, A. Zur, M. Chaudhary, S. Chauhan, J. Huang, A. Arora, Z. Wu, N. Goodman, C. Potts, and T. Icard. Causal abstraction: A theoretical foundation for mechanistic interpretability, 2024. URL https://arxiv.org/abs/2301.04709.

M. Geva, J. Bastings, K. Filippova, and A. Globerson. Dissecting recall of factual associations in auto-regressive language models, 2023. URL https://arxiv.org/abs/2304.14767.

J. Hu, F. Sosa, and T. Ullman. Re-evaluating theory of mind evaluation in large language models. *arXiv preprint arXiv:2502.21098*, 2025.

M. Kosinski. Evaluating large language models in theory of mind tasks. *Proceedings of the National Academy of Sciences*, 121(45), Oct. 2024. ISSN 1091-6490. doi: 10.1073/pnas.2405460121. URL http://dx.doi.org/10.1073/pnas.2405460121.

A. Mueller, J. Brinkmann, M. Li, S. Marks, K. Pal, N. Prakash, C. Rager, A. Sankaranarayanan, A. S. Sharma, J. Sun, E. Todd, D. Bau, and Y. Belinkov. The quest for the right mediator: A history, survey, and theoretical grounding of causal interpretability, 2024. URL https://arxiv.org/abs/2408.01416.

N. Prakash, T. R. Shaham, T. Haklay, Y. Belinkov, and D. Bau. Fine-tuning enhances existing mechanisms: A case study on entity tracking. In *Proceedings of the 2024 International Conference on Learning Representations*, 2024. arXiv:2402.14811.

D. Premack and G. Woodruff. Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1(4):515–526, 1978. doi: 10.1017/S0140525X00076512.

P. Smolensky. Neural and conceptual interpretation of PDP models. In J. L. McClelland, D. E. Rumelhart, and the PDP Research Group, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Psychological and Biological Models*, volume 2, pages 390–431. MIT Press, 1986.

W. Street, J. O. Siy, G. Keeling, A. Baranes, B. Barnett, M. McKibben, T. Kanyere, A. Lentz, B. A. y Arcas, and R. I. M. Dunbar. Llms achieve adult human performance on higher-order theory of mind tasks, 2024. URL https://arxiv.org/abs/2405.18870.

J. Vig, S. Gehrmann, Y. Belinkov, S. Qian, D. Nevo, Y. Singer, and S. Shieber. Investigating gender bias in language models using causal mediation analysis. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12388–12401. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/92650b2e92217715fe312e6fa7b90d82-Paper.pdf.

A Full prompt

No Visibility

```
Instruction: 1. Track the belief of each character as described in the
story. 2. A character's belief is formed only when they perform an action
themselves or can observe the action taking place. 3. A character does
not have any beliefs about the container and its contents which they cannot
observe. 4. To answer the question, predict only what is inside the queried
container, strictly based on the belief of the character, mentioned in the
question. 5. If the queried character has no belief about the container in
question, then predict 'unknown'. 6. Do not predict container or character
as the final output.
Story: Bob and Carla are working in a busy restaurant. To complete an
order, Bob grabs an opaque bottle and fills it with beer. Then Carla grabs
another opaque cup and fills it with coffee.
Question: What does Bob believe the bottle contains?
Answer:
```

Explicit Visibility

```
Instruction: 1. Track the belief of each character as described in the
story. 2. A character's belief is formed only when they perform an action
themselves or can observe the action taking place. 3. A character does
not have any beliefs about the container and its contents which they cannot
observe. 4. To answer the question, predict only what is inside the queried
container, strictly based on the belief of the character, mentioned in the
question. 5. If the queried character has no belief about the container in
question, then predict 'unknown'. 6. Do not predict container or character
as the final output.
Story: Bob and Carla are working in a busy restaurant. To complete an
order, Bob grabs an opaque bottle and fills it with beer. Then Carla
grabs another opaque cup and fills it with coffee. Bob can observe Carla's
actions. Carla cannot observe Bob's actions.
Question: What does Bob believe the cup contains?
Answer:
```

B The CausalToM Dataset

In total, there are 4 templates (one without and 3 with explicit visibility statements). Each template allows 4 different types of questions (CharacterX asked about ObjectY). We used lists of 103 characters, 21 objects, and 23 states. In our interchange intervention experiments, we randomly sample 80 pairs of original and counterfactual stories.

C Desiderate Based Patching Via Causal Abstraction

Causal Models and Interventions A deterministic causal model $\mathcal{M}$ has *variables* that take on *values*. Each variable has a *mechanism* that determines the value of the variable based on the values of *parent variables*. Variables without parents, denoted $\mathbf{X}$, can be thought of as inputs that determine the setting of all other variables, denoted $\mathcal{M}(\mathbf{x})$. A *hard intervention* $A \leftarrow a$ overrides the mechanisms of variable A, fixing it to a constant value a.

Interchange Interventions We perform *interchange interventions* [Vig et al., 2020, Geiger et al., 2020] where a variable (or set of features) A is fixed to be the value it would take on if the LM were processing *counterfactual input* $\mathbf{c}$. We write $A \leftarrow \mathsf{Get}(\mathcal{M}(\mathbf{c}), A)$ where $\mathsf{Get}(\mathcal{M}(\mathbf{c}), A)$ is the value of variable A when $\mathcal{M}$ processes input $\mathbf{c}$. In experiments, we will feed a *original input* $\mathbf{o}$ to a model under an interchange intervention $\mathcal{M}_{A \leftarrow \mathsf{Get}(\mathcal{M}(\mathbf{c}), A))}(\mathbf{o})$.

Featurizing Hidden Vectors The dimensions of hidden vectors are not an ideal unit of analysis [Smolensky, 1986], and so it is typical to *featurize* a hidden vector using some invertible function,

e.g., an orthogonal matrix, to project a hidden vector into a new variable space with more interpretable dimensions called "features"[Mueller et al., 2024]. A feature intervention $\mathbf{F}_\mathbf{h} \leftarrow \mathbf{f}$ edits the mechanism of a hidden vector $\mathbf{h}$ to fix the value of features $\mathbf{F}_\mathbf{h}$ to $\mathbf{f}$.

Alignment The LM is a *low-level causal model* $\mathcal{L}$ where variables are dimensions of hidden vectors and the hypothesis about LM structure is a *high-level causal model* $\mathcal{H}$. An *alignment* Π assigns each high-level variable A to features of a hidden vector $\mathbf{F}^A_\mathbf{h}$, e.g., orthogonal directions in the activation space of $\mathbf{h}$. To evaluate an alignment, we perform intervention experiments to evaluate whether high-level interventions on the variables in $\mathcal{H}$ have the same effect as interventions on the aligned features in $\mathcal{L}$.

Causal Abstraction We use interchange interventions to reveal whether the hypothesized causal model $\mathcal{H}$ is an abstraction of an LM $\mathcal{L}$. To simplify, assume both models share an input and output space. The high-level model $\mathcal{H}$ is an abstraction of the low-level model $\mathcal{L}$ under a given alignment when each high-level interchange intervention and the aligned low-level intervention result in the same output. For a high-level intervention on A aligned with low-level features $\mathbf{F}^A_\mathbf{h}$ with a counterfactual input $\mathbf{c}$ and original input $\mathbf{b}$, we write

$$\mathsf{GetOutput}(\mathcal{L}_{\mathbf{F}^A_\mathbf{h} \leftarrow \mathsf{Get}(\mathcal{L}(\mathbf{c}), \mathbf{F}^A_\mathbf{h}))}(\mathbf{o})) = \mathsf{GetOutput}(\mathcal{H}_{A \leftarrow \mathsf{Get}(\mathcal{H}(\mathbf{c}), A))}(\mathbf{o})) \tag{1}$$

If the low-level interchange intervention on the LM produces the same output as the aligned high-level intervention on the algorithm, this is a piece of evidence in favor of the hypothesis. This extends naturally to multi-variable interventions [Geiger et al., 2024].

Graded Faithfulness Metric We construct *counterfactual datasets* for each causal variable where an example consists of a base prompt and a counterfactual prompt . The *counterfactual label* is the expected output of the algorithm after the high-level interchange intervention, i.e., the right-side of Equation 1. The interchange intervention accuracy is the proportion of examples for which Equation 1 holds, i.e., the degree to which $\mathcal{H}$ faithfully abstracts $\mathcal{L}$.

Aligning Features to Causal Variables In our experiments, we use Singular Vector Decomposition (SVD) to featurize residual stream vectors, i.e., features are the orthogonal singular vectors. For a given transformer layer and token location, we collect the residual stream vectors across a large number of examples and compute the singular vectors. Given singular vector features $\mathbf{F}_\mathbf{h}$ of a hidden vector $\mathbf{h}$ in the residual stream of the LM $\mathcal{L}$, we select features to align with a causal variable A in causal model $\mathcal{H}$ using Desiderata-based Component Masking (DCM) [De Cao et al., 2020, Davies et al., 2023, Prakash et al., 2024]. Given original input $\mathbf{o}$ and counterfactual input $\mathbf{c}$, we train a mask $\mathbf{m} \in [0,1]^{|\mathbf{F}_\mathbf{h}|}$ on the following objective

$$\mathsf{CE}\Big(\mathsf{GetLogits}\big(\mathcal{L}_{\mathbf{F}_\mathbf{h} \leftarrow \mathbf{m} \circ \mathsf{Get}(\mathcal{L}(\mathbf{c}), \mathbf{F}_\mathbf{h}))}(\mathbf{b})\big), \mathsf{GetLogits}\big(\mathcal{H}_{A \leftarrow \mathsf{Get}(\mathcal{H}(\mathbf{c}), A))}(\mathbf{b})\big)\Big) \tag{2}$$

D Pseudocode for the Belief Tracking High-Level Causal Model

Algorithm 2 High-level causal model for the no visibility

procedure BELIEFTRACKING($c_1, o_1, s_1, c_2, o_2, s_2, q_c, q_o$)
    Ordering ID assignment
    $c_1^{OI}, o_1^{OI}, s_1^{OI} \leftarrow$ AssignOIs($[c_1, o_1, s_1], 1$)
    $c_2^{OI}, o_2^{OI}, s_2^{OI} \leftarrow$ AssignOIs($[c_2, o_2, s_2], 2$)

    Binding lookback mechanism
    binding_address$_1 \leftarrow$ (copy(c_1^{OI}), copy(o_1^{OI}))
    binding_address$_2 \leftarrow$ (copy(c_2^{OI}), copy(o_2^{OI}))

    $q_c^{OI} \leftarrow$ copy($\{c_1 : c_1^{OI}, c_2 : c_2^{OI}\}[q_c]$)
    $q_o^{OI} \leftarrow$ copy($\{o_1 : o_1^{OI}, o_2 : o_2^{OI}\}[q_o]$)
    binding_pointer $\leftarrow (q_c^{OI}, q_o^{OI})$

    if binding_address$_1$ = binding_pointer **then**
        binding_payload $\leftarrow$ copy(s_1^{OI})
    else if binding_address$_2$ = binding_pointer **then**
        binding_payload $\leftarrow$ copy(s_2^{OI})
    end if

    Answer lookback mechanism
    answer_pointer $\leftarrow$ binding_payload
    answer1_address $\leftarrow s_1^{OI}$
    answer2_address $\leftarrow s_2^{OI}$
    if answer1_address = answer_pointer **then**
        answer_payload $\leftarrow s_1$
    else if answer2_address = answer_pointer **then**
        answer_payload $\leftarrow s_2$
    end if
    return answer_payload
end procedure

E Uncovering the Binding Lookback Mechanism

The *Binding Lookback* is the first operation applied to these OIs. The character and object OIs, serving as the source information, are each copied twice. One copy, referred to as the address, is placed in the residual stream of the state token (recalled token), alongside the state OI as the payload to transfer. The other copy, referred to as the pointer, is moved in the residual stream of the final token (lookback token). These pointer and address copies are then used to form the QK-circuit at the lookback token, which dereferences the state OI payload, transferring it from the state token to the final token. See Fig.2a (i) for a schematic of this lookback and see Fig.1 for the general mechanism.

The Hypothesized Address and Payload. In our first experiment, we localize the address copies of the character and object OIs and the state OI payload to the residual stream of the state token (recalled token, Fig. 2a). We sampled a counterfactual dataset where each example consists of an original input **o** with an answer that isn't *unknown* and a counterfactual input **c** where the character, object, and state tokens are identical, except the ordering of the two sentences is swapped while the question remains unchanged, as illustrated in Fig. 4. The expected outcome predicted by our high-level causal model under intervention is the other state token from the original example, e.g., **beer**, because reversing the address and payload values without changing the pointer flips the output.

Testing Address and Payload Hypothesis. We perform an interchange intervention experiment layer-by-layer, where we replace the residual stream vectors at the first state token in the original run with that of the second state token in the counterfactual run and vice versa for the other state token. It is important to note that if the intervention targets state token values instead of their OIs, it should not produce the expected output. (This happens in the earlier layers.)

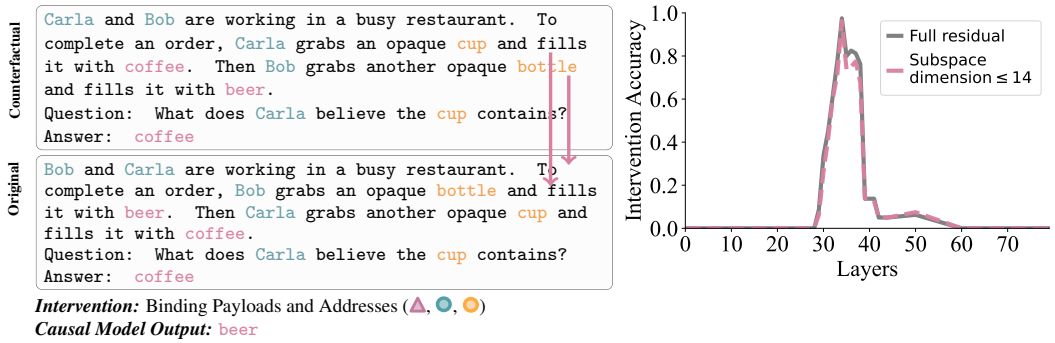


Figure 4: **Binding lookback payload and address:** We intervened on both the high-level causal model and the LM running on the original story, modifying their variables and internal activations respectively, to match those from a counterfactual scenario. In the causal model intervention, we update the addresses (character and object OIs; and) and the payloads (state OIs;). This causes the binding lookback mechanism to attend to and retrieve the state OI corresponding to the alternate state token, which is then dereferenced by answer lookback to yield the alternate state token (e.g., **beer** instead of **coffee**). In the LM interchange intervention, modifying the residual stream at the state token results in identical outputs between layers 33 and 38. This confirms our hypothesis that both the address and payload information are represented in the residual stream of state tokens.

As shown in Fig. 4, the strongest alignment occurs between layers 33 and 38, supporting our hypothesis that the state token's residual stream contains both the address information (character and object OIs) and the payload information (state OI). These components are subsequently used to form a QK-circuit between the pointer at the lookback token and the address at the other state token and OV-circuit that retrieves its state OI as the payload.

Localizing the Source Information As shown in Fig. 2a, the source information is copied as both the address and the pointer at different token positions. To localize the source information, we conduct intervention experiments with a dataset where the counterfactual example, $\mathbf{c}$, swaps the order of the characters and objects as well as replaces the state tokens with entirely new ones while keeping the question the same as in $\mathbf{o}$.

With this dataset, an interchange intervention on the high-level causal model that targets the source information will have downstream effects on both the address and the pointer, so no change in output occurs. However, if we additionally freeze the payloads and addresses, the causal model outputs the other state token, e.g., **beer** in Fig. 5, due to the mismatch between address and pointer.

In the LM, we interchange the residual streams of the character and object tokens while keeping the residual stream of the state token fixed. When the output of the intervened LM aligns with that of the intervened causal model, it indicates that the QK-circuit at the final token is attending to the alternate state token. As shown in Fig. 5, the second experiment reveals alignment between layers 20 and 34. This suggests that source information—specifically, the character and object OIs—is represented in their respective token residual streams within this layer range.

We provide more experimental results in Appendix H where we show in Fig. 7 that freezing the residual stream of the state token is necessary. In sum, these results not only provide evidence for the presence of source information but also establish its transfer to the recalled and lookback tokens as addresses and pointers, respectively.

Localizing the Pointer Information The pointer copies of the character and object OI are first formed at the character and object tokens in the question before being moved again to the final token for dereferencing (see Appendix I for experiments and more details).

F Uncovering the Answer Lookback Mechanism

The LM answers the question using the *Answer Lookback*. The state OI of the correct answer serves as the source information, which is copied into two instances. One instance, the address copy of the state OI, is in the residual stream of the state token (recalled token) with the state token itself as the payload. The other instance, the pointer copy of the state OI, is transferred to the residual stream of the final token (lookback token) as the payload of the binding lookback. This pointer is

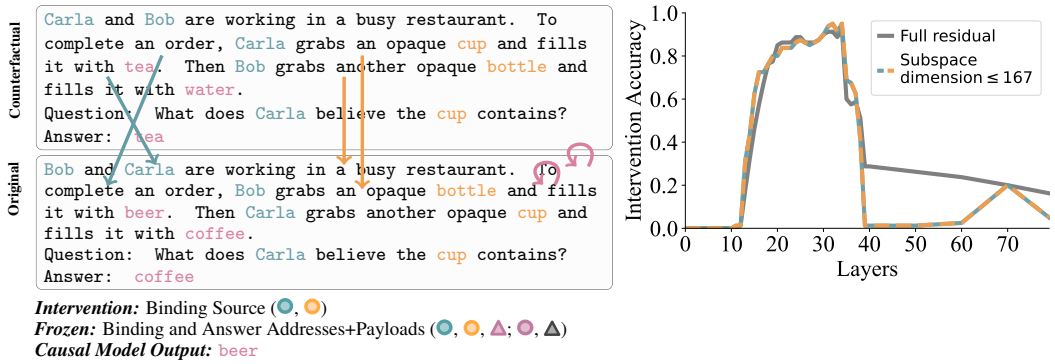


Figure 5: **Source Information of Binding lookback**: We run the causal model and the LM on an original story, then update their variables and activations, respectively, to the values they would take on for a counterfactual story with swapped characters and objects and new states. The interchange intervention on the high-level causal model swaps the sources of the binding lookback (character and object OIs; ●, ●) while freezing the addresses and payloads of the binding lookback (character, object, and state OIs; ●, ●, ▲) and the answer lookback (state OI and token; ●, ▲). By altering the sources, but freezing the addresses and payloads, only the pointer is changed so the binding lookback retrieves the other state OI which is dereferenced by the answer lookback to the other state token (e.g., **beer** instead of **coffee**). We perform the same interchange intervention on the LM and measure the agreement with the intervened causal model. Our results localize the source to the character and object token residual streams between layers 20 and 34.

then dereferenced, bringing the state token as the payload into the residual stream of the final token, which is predicted as the final output. See Fig. 2a (ii) for the answer lookback and Fig. 1 for the general mechanism.

Localizing the Pointer Information We first localize the pointer of the answer lookback, which is the payload of the binding lookback. To do this, we conduct an interchange intervention experiment where the residual vectors at the final token position in the original run are replaced with those from the counterfactual run, one layer at a time. The counterfactual inputs have swapped objects and characters and randomly sampled states. If the answer pointer is targeted for intervention in the high-level causal model, the output is the other state in the original input, e.g., **beer**. As shown in Fig. 2b, alignment begins at layer 34, indicating that this layer contains pointer information, in low-rank subspace, which remains causally relevant until layer 52.

Localizing the Payload To determine where the model uses the state OI pointer to retrieve the state token, we use the same interchange intervention experiment. However, if the answer payload is targeted for intervention in the high-level causal model, the output is the correct state token from the counterfactual example, e.g., **tea**, rather than the state token from the original example, as illustrated in Fig. 2b. The alignment occurs after layer 56, indicating that the model retrieves the correct state token (payload) into the final token's residual stream by 56, where it is subsequently used to generate the final output.

G Uncovering the Visibility Lookback Mechanism

Localizing the Source Information To localize the source information, we conduct an interchange intervention experiment where the counterfactual is a different story with altered visibility information. In the original example, the first character cannot observe the second character's actions, whereas in the counterfactual example, the first character can observe them (Fig. 6). The causal model outcome of this intervention is a change in the final output of the original run from "unknown" to the state token associated with the queried object. The interchange intervention is executed on visibility sentence tokens. As shown in Fig. 6 (— line), alignment occurs between layers 10 and 23, indicating that the visibility ID remains encoded in the visibility sentence until layer 23, after which it is duplicated into address and pointer copies on visibility sentence and subsequent tokens respectively.

Localizing the Payload To localize the payload information, we use the same counterfactual dataset. However, instead of intervening on the source or recalled tokens, we intervene on the lookback tokens, specifically the question and answer tokens. As in the previous experiment, we

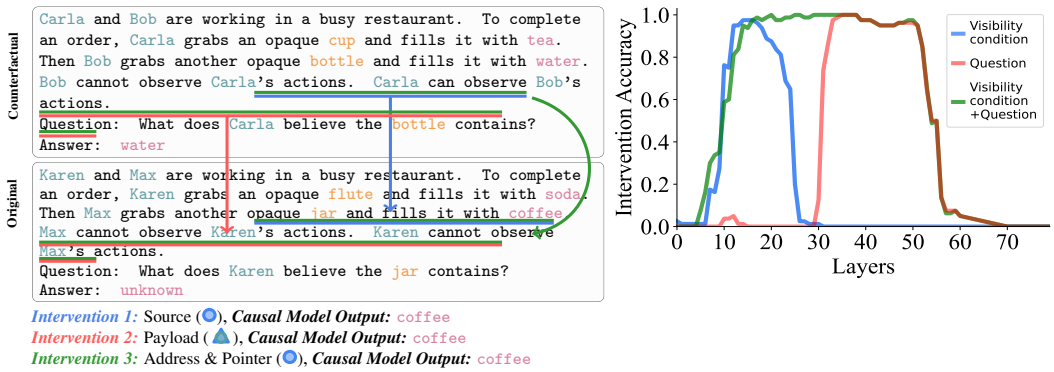


Figure 6: **Visibility Lookback**: We conduct three interchange intervention experiments to support the Visibility Lookback hypothesis: (1) *Source Alignment*: We align the source information ( ) by intervening on the visibility sentence—replacing it with its representation from a counterfactual run where the visibility sentence causes the queried character to become aware of the queried object's contents. We observe that source information aligns between layers 10 and 23, after which it splits into separate address and pointer components. (2) *Payload Alignment*: To align the payload ( ), we intervene on all subsequent tokens and observe alignment only after layer 31. (3) *Address and Pointer Alignment*: When intervening on both the address and pointer information ( ), we observe alignment across a broader range of layers, particularly between layers 24 and 31, because of the enhanced alignment between the address and pointer copies at the recalled and lookback tokens.

replace the residual vectors of these tokens in the original run with those from the counterfactual run. As shown in Fig. 6 (— line), alignment occurs only after layer 31, indicating that the information enhancing the queried character's awareness is present in the lookback tokens only after this layer.

Localizing the Address and Pointer The previous two experiments suggest the presence of a lookback mechanism, as there is no signal indicating that the source or payload has been formed between layers 24 and 31. We hypothesize that this lack of signal is due to a mismatch between the address and pointer information at the recalled and lookback tokens. Specifically, when intervening only on the recalled token after layer 25, the pointer is not updated, whereas intervening only on the lookback tokens leaves the address unaltered, leading to the mismatch. To test this hypothesis, we conduct another intervention using the same counterfactual dataset, but this time, we intervene on the residual vectors of both the recalled and lookback tokens, i.e., the visibility sentence, as well as the question and answer tokens. As shown in Fig. 6 (— line), alignment occurs after layer 10 and remains stable, supporting our hypothesis. This intervention replaces both the address and pointer copies of the visibility IDs, enabling the LM to form a QK-circuit and retrieve the payload.

H Aligning Character and Object OIs

As mentioned in section E, the source information, consisting of character and object OI, is duplicated to form the address and pointer of the binding lookback. Here, we describe another experiment to verify that the source information is copied to both the address and the pointer. More specifically, we conduct the same interchange intervention experiment as described in Fig. 5, but without freezing the residual vectors at the state tokens. Based on our hypothesis, this intervention will not be able to change the state of the original run, since the intervention at the source information will affect both address and pointer, hence making the model form the original QK-circuit.

In section E, we identified the source of the information but did not fully determine the locations of each character and object OI. To address this, we now localize the character and object OIs separately to gain a clearer understanding of the layers at which they appear in the residual streams of their respective tokens, as shown in Fig.8 and Fig.9.

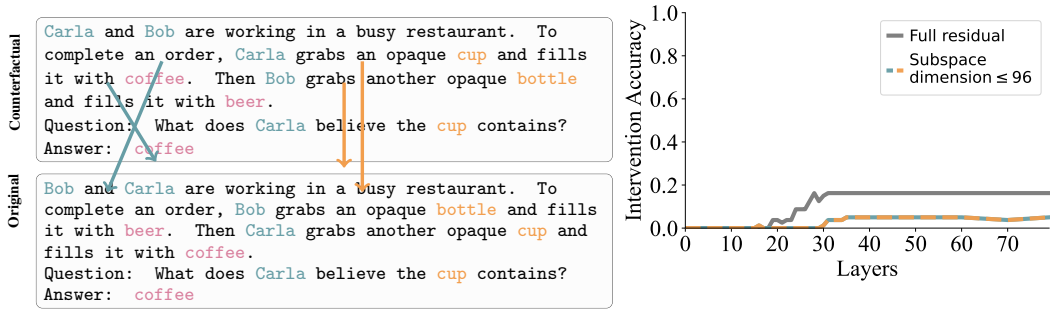


Figure 7: **Source Information** of Binding lookback: In this interchange intervention experiment, the source information—i.e., the character and object OIDs (,)—is modified, while the address and payload (, ,) are recomputed based on the modified source. Since both the address and pointer information are derived from the altered source, the binding lookback ultimately retrieves the same original state token as the payload. As a result, we do not observe high intervention accuracy.

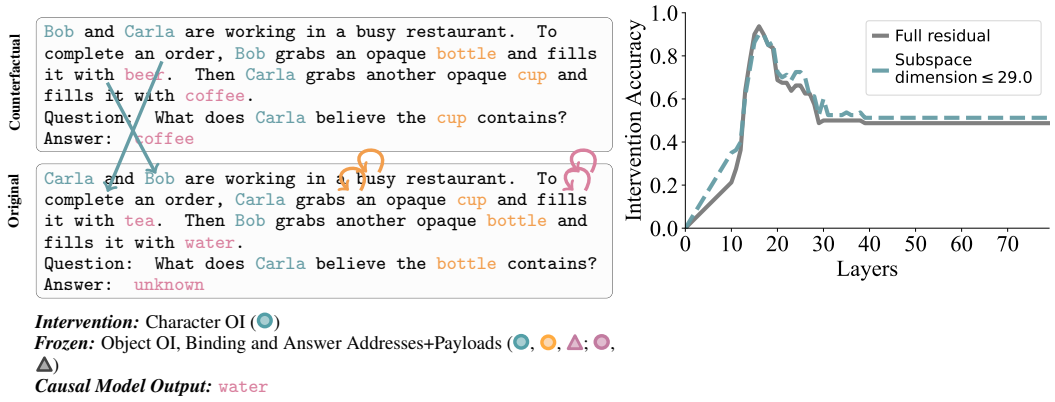


Figure 8: **Character OI**: This interchange intervention experiment swaps the character OI ( ), while freezing the object OI as well as binding lookback address and payload (, ,). Swapping the character OIs in the story tokens changes the queried character OI to the other one. Hence, the final output changes from *unknown* to **water**.

I Aligning Query Character and Object OIs

In section E, we localized the pointer information of binding lookback. However, we found that this information is transferred to the lookback token (last token) through two intermediate tokens: the queried character and the queried object. In this section, we separately localize the OIs of the queried character and queried object, as shown in Fig. 10 and Fig. 11.

J Speculated Payload in Visibility Lookback

As mentioned in section G, the payload of the Visibility lookback remains undetermined. In this section, we attempt to disambiguate its semantics using the Attention Knockout technique introduced in [Geva et al., 2023], which helps reveal the flow of crucial information. We apply this technique to understand which previous tokens are vital for the formation of the payload information. Specifically, we "knock out" all attention heads at all layers of the second visibility sentence, preventing them from attending to one or more of the previous sentences. Then, we allow the attention heads to attend to the knocked-out sentence one layer at a time.

If the LM is fetching vital information from the knocked-out sentence, the interchange intervention accuracy (IIA) post-knockout will decrease. Therefore, a decrease in IIA will indicate which attention

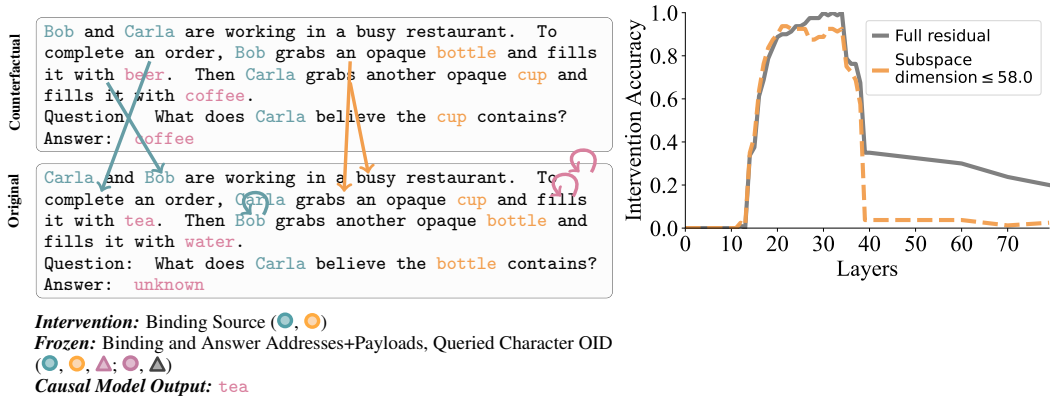


Figure 9: **Object OI**: This interchange intervention experiment swaps both the character and object OIs, while freezing the address and payload of binding lookback as well as queried character OI. Swapping both character and object OIs in the story tokens ensures that the queried object gets the other OI. Hence, the final output changes from *unknown* to **tea**.

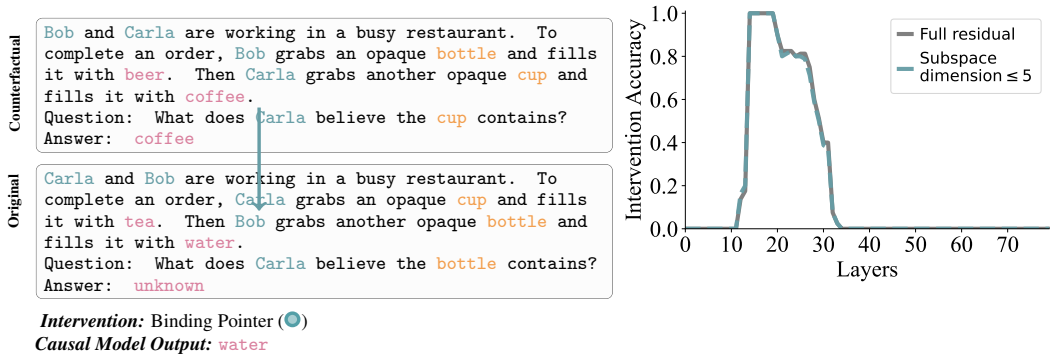


Figure 10: **Query Character OI**: This interchange intervention experiment alters the OI of the queried character to the other one. Hence, the final output changes from *unknown* to **water**.

heads, at which layers, are bringing in the vital information from the knocked-out sentence. If, however, the model is not fetching any critical information from the knocked-out sentence, then knocking it out should not affect the IIA.

To determine if any vital information is influencing the formation of the Visibility lookback payload, we perform three knockout experiments: 1) Knockout attention heads from the second visibility sentence to both the first visibility sentence and the second story sentence (which contains information about the observed character), 2) Knockout attention heads from the second visibility sentence to only the first visibility sentence, and 3) Knockout attention heads from the second visibility sentence to the second story sentence. In each experiment, we measure the effect of the knockout using IIA.

Fig.12 shows the experimental results. Knocking out any of the previous sentences affects the model's ability to produce the correct output. The decrease in IIA in the early layers can be explained by the restriction on the movement of character OIs. Specifically, the second visibility sentence mentions the first and second characters, whose character OIs must be fetched before the model can perform any further operations. Therefore, we believe the decrease in IIA until layer 15, when the character OIs are formed (based on the results from Section H), can be attributed to the model being restricted from fetching the character OIs. However, the persistently low IIA even after this layer—especially when both the second and first visibility sentences are involved—indicates that some vital information is being fetched by the second visibility sentence, which is essential for forming the coherent Visibility lookback payload. Thus, we speculate that the Visibility payload encodes information about the observed character, specifically their character OI, which is later used to fetch the correct state OI.

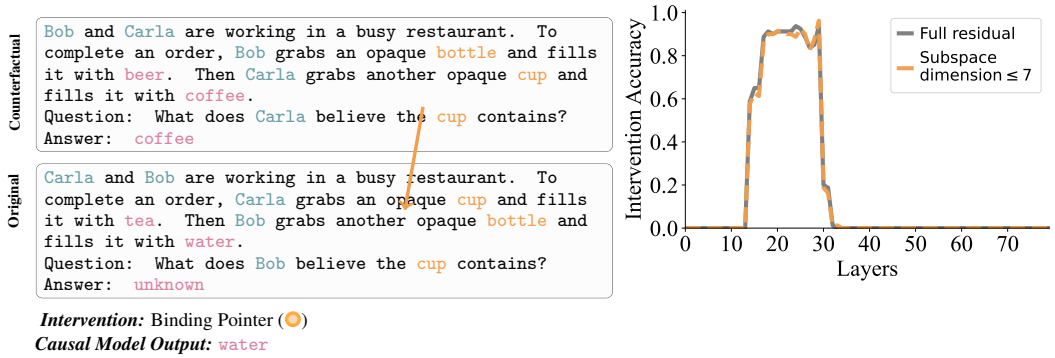


Figure 11: **Query Object OI**: This interchange intervention experiment alters the OI of the queried object ( ) to the other one. Hence, the final output changes from *unknown* to **water**.

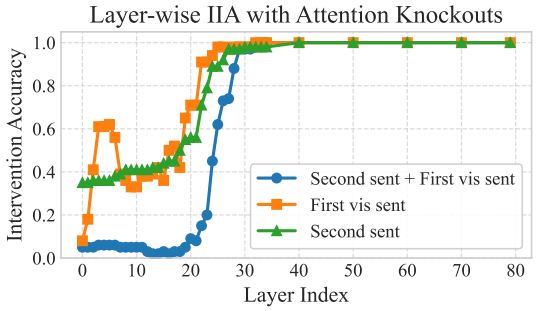


Figure 12: At the second visibility sentence, attention heads are restricted to retrieve information from one of three prior contexts: (1) both the second story sentence and the first visibility sentence (— line), (2) only the first visibility sentence (— line), or (3) only the second story sentence (— line).

K Correlation Analysis of Causal Subspaces and Attention Heads

This section identifies the attention heads that align with the causal subspaces discovered in the previous sections. Specifically, first we focus on attention heads whose query projections are aligned with the subspaces—characterized by the relevant singular vectors—that contain the correct answer state OI. To quantify this alignment between attention heads and causal subspaces, we use the following computation.

Let $Q \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ denote the query projection weight matrix for a given layer:

We normalize Q column-wise:

$$\tilde{Q}_{:,j} = \frac{Q_{:,j}}{\|Q_{:,j}\|} \quad \text{for each column } j \tag{3}$$

Let $S \in \mathbb{R}^{d_{\text{model}} \times k}$ represent the matrix of k singular vectors (i.e., the causal subspace basis). We project the normalized query weights onto this subspace:

$$Q_{\text{sv}} = \tilde{Q} \cdot S \tag{4}$$

We then reshape the resulting projection into per-head components. Assuming $Q_{\text{sv}} \in \mathbb{R}^{d_{\text{model}} \times k}$, and each attention head has dimensionality d_h, we write:

$$Q_{\text{head}}^{(i)} = Q_{\text{sv}}^{(i)} \in \mathbb{R}^{d_h \times k} \quad \text{for } i = 1, \ldots, n_{\text{heads}} \tag{5}$$

Finally, we compute the norm of each attention head's projection:

$$\text{head\_norm}_i = \left\| Q_{\text{head}}^{(i)} \right\|_F \quad \text{for } i = 1, \ldots, n_{\text{heads}} \tag{6}$$

We compute the $head\_norm$ for each attention head in every layer, which quantifies how strongly a given head reads from the causal subspace present in the residual stream. The results are presented in Fig. 13, and they align with our previous findings: attention heads in the later layers form the QK-circuit by using pointer and address information to retrieve the payload during the Answer lookback.

We perform a similar analysis to check which attention heads' value projection matrix align with the causal subspace that encodes the payload of the Answer lookback. Results are shown in Fig. 14, indicating that attention heads at later layers primarily align with causal subspace containing the answer token.

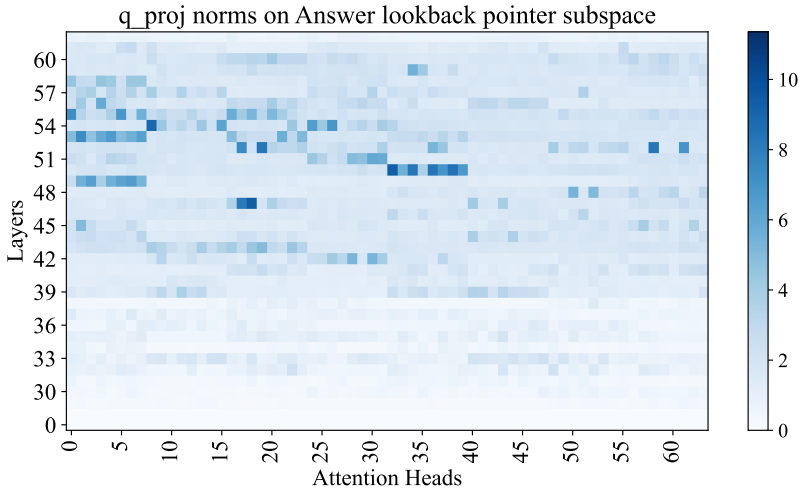


Figure 13: Alignment between the Answer lookback pointer causal subspace and the query projection matrix in Llama-3-70B-Instruct.

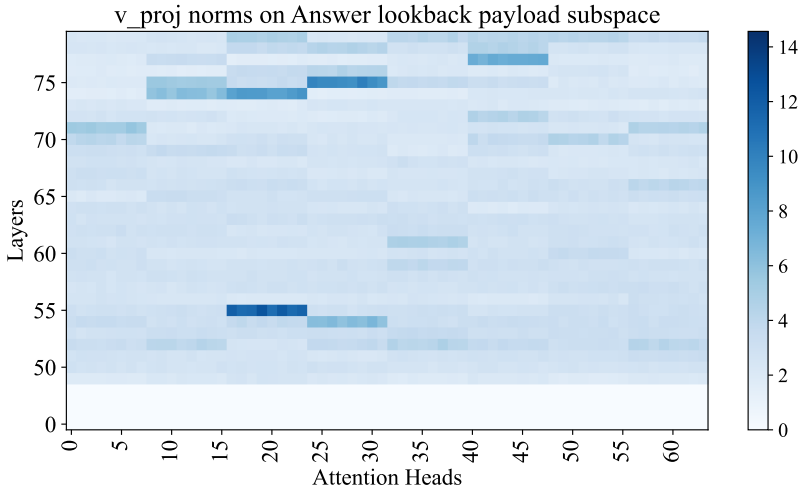


Figure 14: Alignment between the Answer lookback payload causal subspace and the value projection matrix in Llama-3-70B-Instruct.

L Belief Tracking Mechanism in BigToM Benchmark

This section presents preliminary evidence that the mechanisms outlined in Sections 4 and G generalize to other benchmark datasets. Specifically, we demonstrate that Llama-3-70B-Instruct answers the

belief questions (true belief and false belief) in the BigToM dataset Gandhi et al. [2024] in a manner similar to that observed for CausalToM: by first converting token values to their corresponding OIs and then performing logical operations on them using lookbacks. However, as noted in Section 3, BigToM—like other benchmarks—lacks the coherent structure necessary for causal analysis. As a result, we were unable to replicate all experiments conducted on CausalToM. Thus, the results reported here provide only preliminary evidence of a similar underlying mechanism.

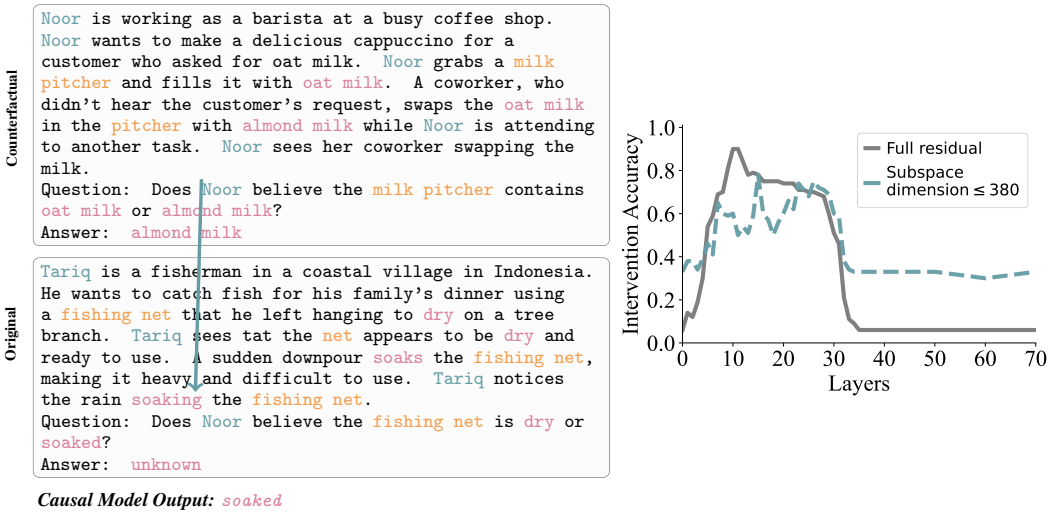


Figure 15: **Query Character OI in BigToM**: This interchange intervention experiment inserts the first character's OI into the residual stream at the queried character token (), resulting in the movement of pointer information to the last token that aligns with the address information of binding lookback mechanism. Consequently, the model is able to form the appropriate QK-circuit from the last token to predict the correct state answer token(s) as the final output, instead of unknown.

To justify the presence of OIs, we conduct an interchange intervention experiment, similar to the one described in Section I, aiming to localize the character OI at the character token in the question sentence. We construct an original sample by replacing its question sentence with that of a counterfactual sample, selected directly from the unaltered BigToM dataset. Consequently, when processing the original sample, the model has no information about the queried character and, as a result, produces unknown as the final output. However, if we replace the residual vector at the queried character token in the original sample with the corresponding vector from the counterfactual sample (which contains the character OI), the model's output changes from unknown to the state token(s) associated with the queried object. This is because inserting the character OI at the queried token provides the correct pointer information, aligning with the address information at the correct state token(s), thereby enabling the model to form the appropriate QK-circuit and retrieve the state's OI. As shown in Fig. 15, we observe a high IIA between layers $9 - 28$—similar to the pattern seen in CausalToM—suggesting that the queried character token encodes the character OI in its residual vector within these layers.

Next, we investigate the Answer lookback mechanism in BigToM, focusing specifically on localizing the pointer and payload information at the final token position. To localize the pointer information, which encodes the correct state OI, we construct original and counterfactual samples by selecting two completely different examples from the BigToM dataset, each with different ordered states as the correct answer. For example, as illustrated in Fig.16, the counterfactual sample designates the first state as the answer, **thrilling plot**, whereas the original sample designates the second state, **almond milk**. We perform an intervention by swapping the residual vector at the last token position from the counterfactual sample into the original run. The causal model outcome of this intervention is that the model will output the alternative state token from the original sample, **oat milk**. As shown in Fig.16, this alignment occurs between layers 33 and 51, similar to the layer range observed for the pointer information in the Answer lookback of CausalToM.

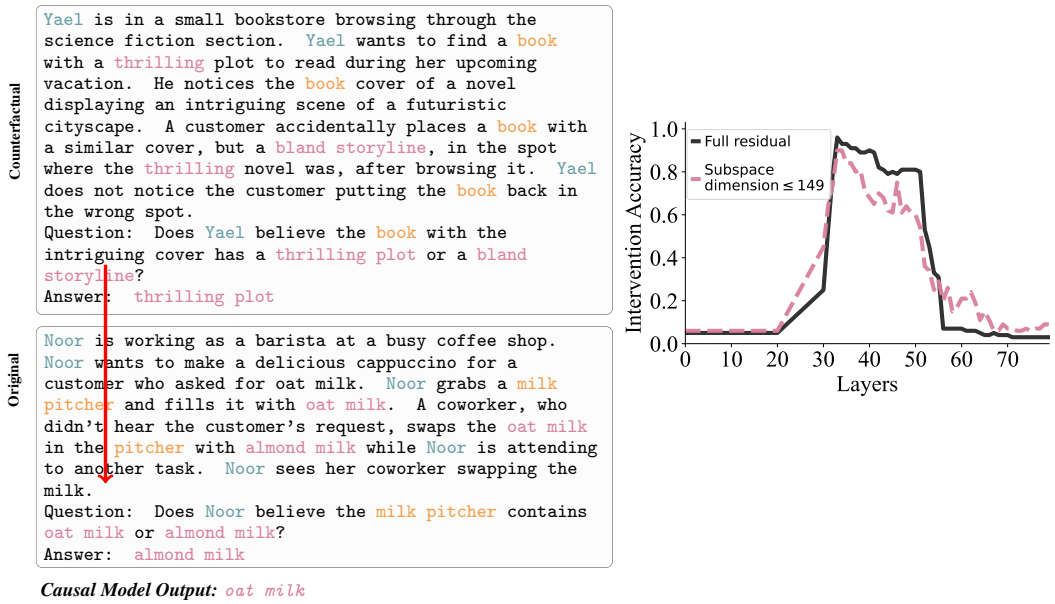


Figure 16: **Answer Lookback Pointer in BigToM**: This interchange intervention experiment modifies the pointer information (●) of the Answer lookback, thereby altering the subsequent QK-circuit to attend to the other state (e.g., **oat milk**) instead of the original one (e.g., **almond milk**). As a result, the model retrieves the token value corresponding to the other state to answer the question.

Further, to localize the payload of the Answer lookback in BigToM, we perform an interchange intervention experiment using the same original and counterfactual samples as mentioned in the previous experiment, but with a different expected output—namely, the correct state from the counterfactual sample instead of the other state from the original sample. As shown in Fig. 17, alignment emerges after layer 59, consistent with the layer range observed for the Answer lookback payload in CausalToM.

Finally, we investigate the impact of the visibility condition on the underlying mechanism and find that, similar to CausalToM, the model uses the Visibility lookback to enhance the observing character's awareness based on the observed character's actions. To localize the effect of the visibility condition, we perform an interchange intervention in which the original and counterfactual samples differ in belief type—that is, if the original sample involves a false belief, the counterfactual involves a true belief, and vice versa. The expected output of this experiment is the other (incorrect) state of the original sample. Following the methodology in Section G, we conduct three types of interventions: (1) only at the visibility condition sentence, (2) only at the subsequent question sentence, and (3) at both the visibility condition and the question sentence. As shown in Fig. 18, intervening only at the visibility sentence results in alignment at early layers, up to layer 17, while intervening only at the subsequent question sentence leads to alignment after layer 26. Intervening on both the visibility and question sentences results in alignment across all layers. These results align with those found in the CausalToM setting shown in the Fig. 6.

Previous experiments suggest that the underlying mechanisms responsible for answering belief questions in BigToM are similar to those in CausalToM. However, we observed that the subspaces encoding various types of information are not shared between the two settings. For example, although the pointer information in the Answer lookback encodes the correct state's OI in both cases, the specific subspaces that represent this information at the final token position differ significantly. We leave a deeper investigation of this phenomenon—shared semantics across distinct subspaces in different distributions—for future work.

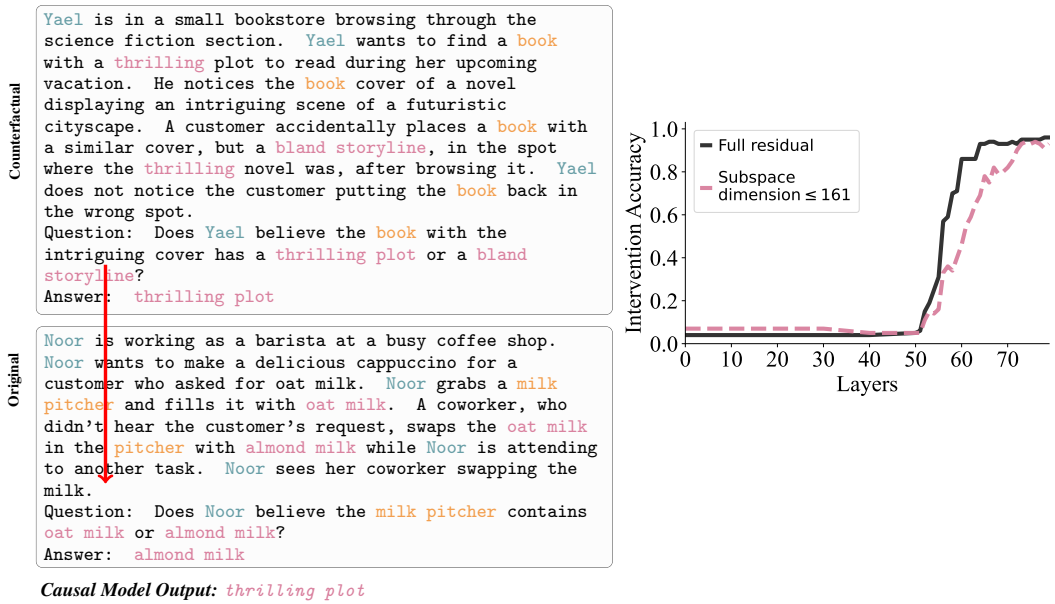
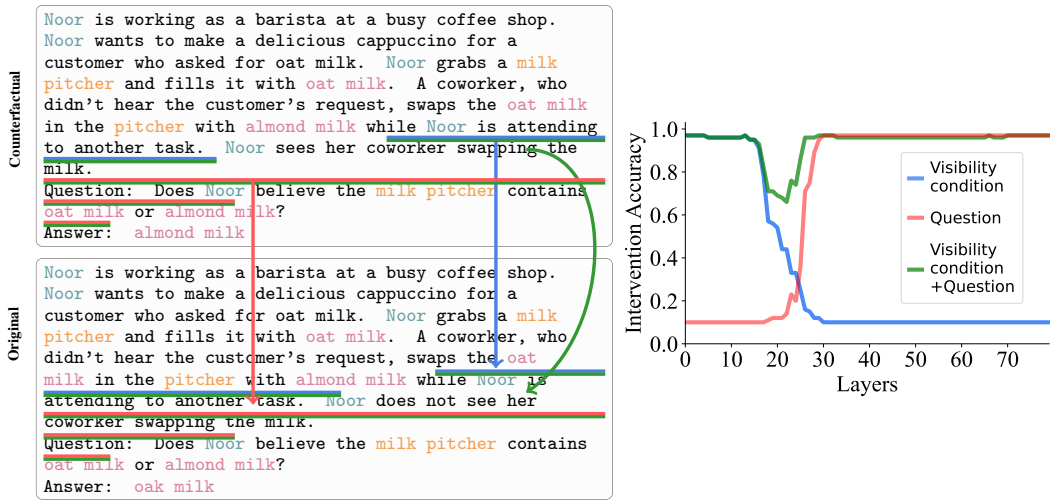


Figure 17: **Answer Lookback Payload in BigToM**: This interchange intervention experiment directly modifies the payload information (▲) of the Answer lookback, which is fetched from the corresponding state tokens and predicted as the next token(s). Thus, replacing its value in the original run, e.g. **almond milk**, with that from the counterfactual run, e.g. **thrilling plot**, causes the model's next predicted tokens to correspond to the correct answer of the counterfactual sample.

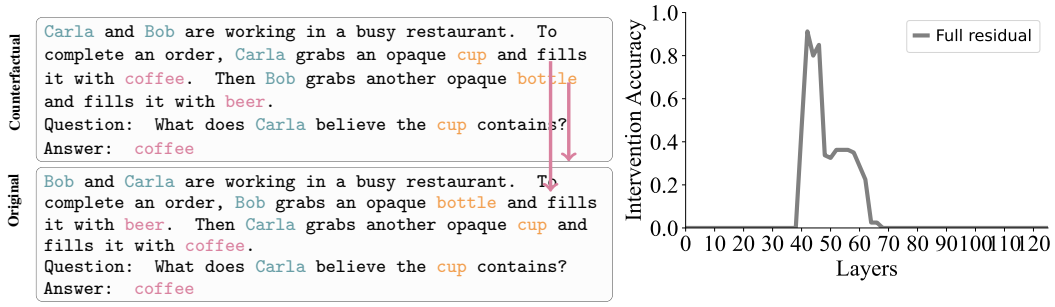
M Generalization of Belief Tracking Mechanism on CausalToM to Llama-3.1-405B-Instruct

This section presents all the interchange intervention experiments described in the main text, conducted using the same set of counterfactual examples on Llama-3.1-405B-Instruct, using NDIF Fiotto-Kaufman et al. [2025]. Each experiment was performed on 80 samples. Due to computational constraints, subspace interchange intervention experiments were not conducted. The results indicate that Llama-3.1-405B-Instruct employs the same underlying mechanism as Llama-3-70B-Instruct to reason about belief and answer related questions. This suggests that the identified belief-tracking mechanism generalizes to other models capable of reliably performing the task.



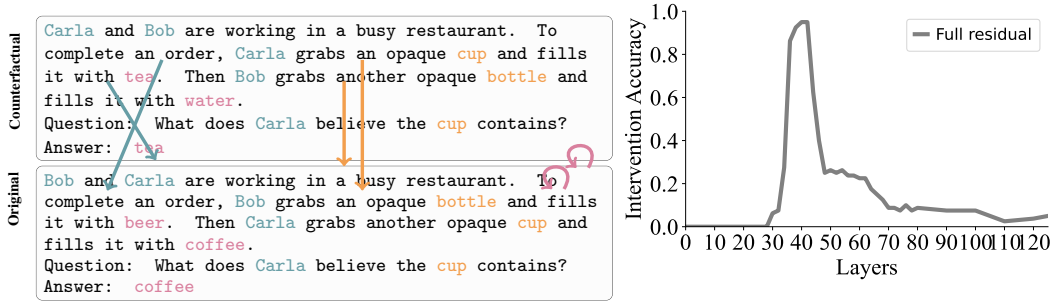
Expected Output: *almond milk*

Figure 18: **Visibility Lookback in BigToM**: We perform three interchange interventions to establish the presence of the Visibility ID, which serves as both address and pointer information. When intervening at the source (●)—i.e., the visibility sentence—both the address and pointer are updated, resulting in alignment across layers. Intervening only at the subsequent question tokens leads to alignment only at later layers, after the model has already fetched the payload (▲). However, intervening at both the visibility and question sentences results in alignment across all layers, as the address and pointer remain consistent throughout.



Causal Model Output: *beer*

Figure 19: **Payload and address of Binding lookback**



Causal Model Output: *beer*

Figure 20: **Source Information of Binding lookback**

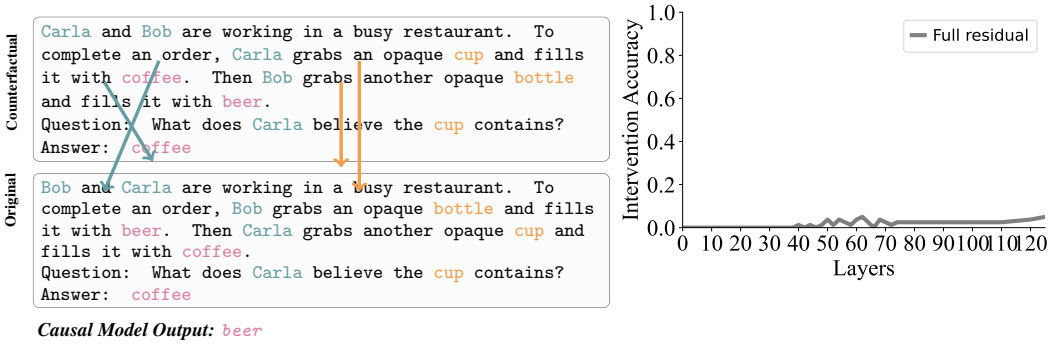


Figure 21: **Source Information of Binding lookback without freezing address and payload**

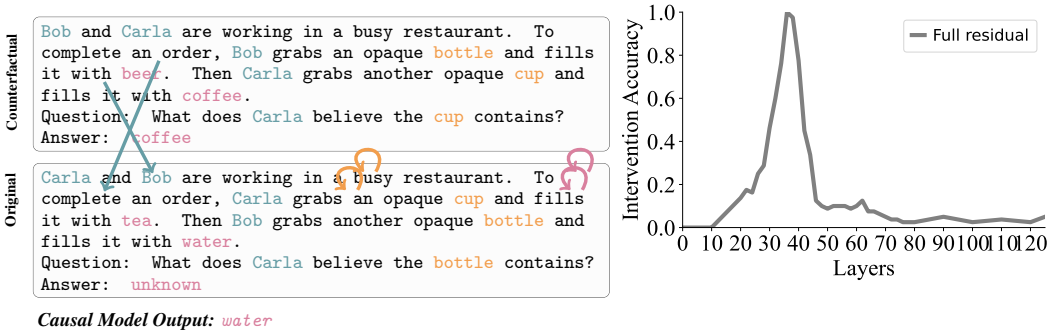


Figure 22: **Character OI**

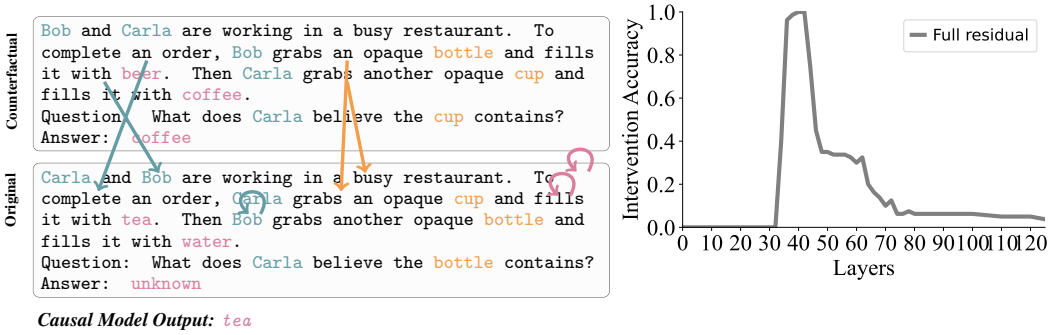


Figure 23: **Object OI**

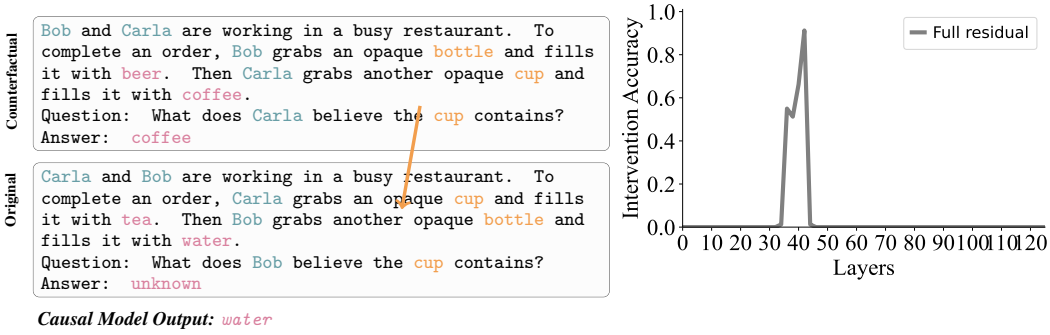


Figure 24: **Query Object OI**

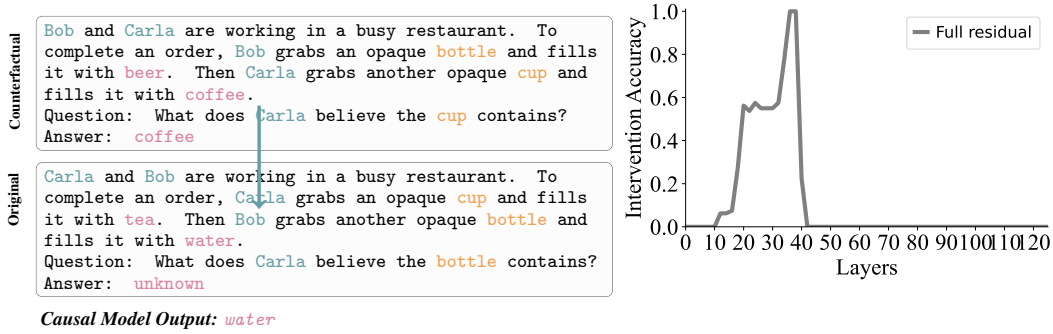


Figure 25: **Query Character OI**

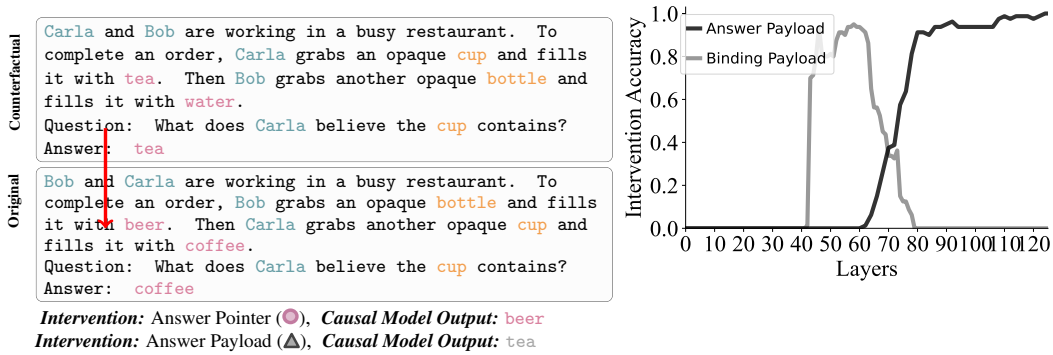


Figure 26: **Answer Lookback Pointer and Payload**

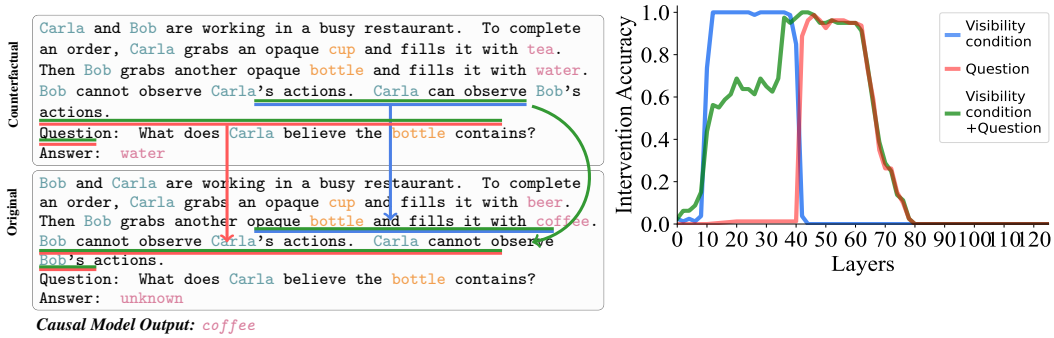


Figure 27: **Visibility Lookback**