where h_o is the full residual stream of the original run, h_c is the full residual stream of the counterfactual run, and h_{new} is the intervened vector where the chosen subspace of h_o is replaced with that of h_c.

The core idea is to first remove the existing information from the subspace defined by the projection matrix and then insert the counterfactual information into that same subspace using the same projection matrix.

In order to find the optimal subspace, we optimize $\mathbf{m}$ to maximize the agreement between the causal model output and the LM's output. To do so, we train the mask for each experiment on 80 examples of the same counterfactual datasets specified in the main text and use another 80 samples as the validation set. We use the following objective function, which maximizes the logit of the causal model output token:

$$\mathcal{L} = -\text{logit}_{\text{causal\_model\_output\_under\_intervention}} + \lambda \sum \mathbf{m} \tag{6}$$

Where λ is a hyperparameter used to control the rank of the subspace and $\mathbf{m}$ is the learnable mask. See Appendix D for details on how the causal model output under intervention are computed. We trained $\mathbf{m}$ for one epoch with ADAM optimizer, on batches of size 4 and a learning rate of 0.01. During training, the parameters of $\mathbf{m}$ are continuous and constrained to lie within the range $[0, 1]$. To enforce this constraint, we clamp their values after each gradient update. During evaluation, we binarize the mask by rounding each parameter to the nearest integer, i.e., 0 or 1.

G Aligning Character and Object OIs

As mentioned in section 5.2, the source reference 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. 6, 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.

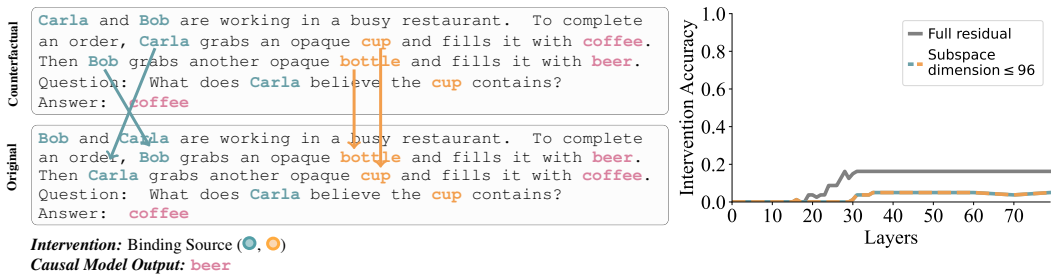

Figure 13: **Source Reference Information** of Binding lookback: In this interchange intervention experiment, the source information, i.e., the character and object OIDs (●, ●), is modified, while the address and payload (●, ●, ▲) are recomputed based on the modified source. Since both the address and 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.

In section 5.2, 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.14 and Fig.15.

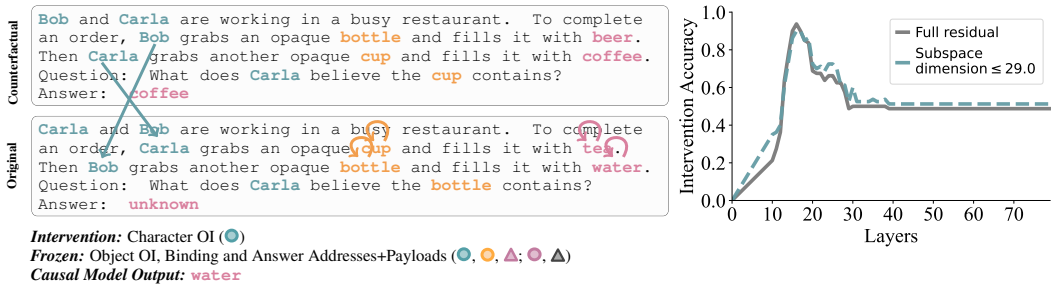

Figure 14: **Character OI**: This interchange intervention experiment swaps the character OI (), while freezing the object OI as well as binding lookback address and payload (, ,). Swapping the character OIs in the story tokens changes the queried character OI to the other one. Hence, the final output changes from *unknown* to **water**.

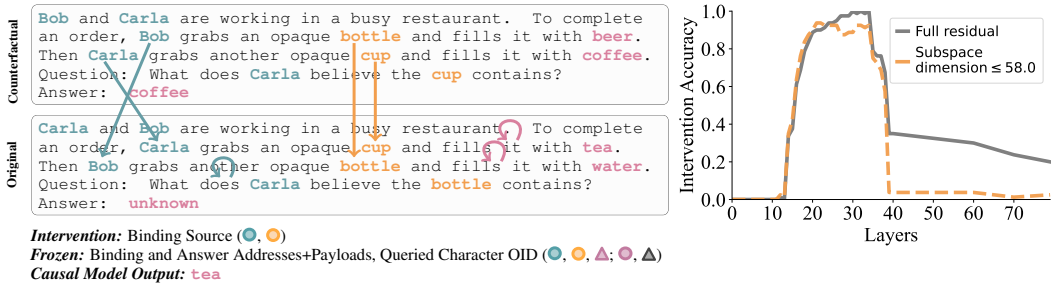

Figure 15: **Object OI**: This interchange intervention experiment swaps both the character and object OIs (,), while freezing the address and payload of binding lookback (, ,) as well as queried character OI (). Swapping both character and object OIs in the story tokens ensures that the queried object gets the other OI. Hence, the final output changes from *unknown* to **tea**.

H Aligning Query Character and Object OIs

In section 5.2, 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. 16 and Fig. 17.

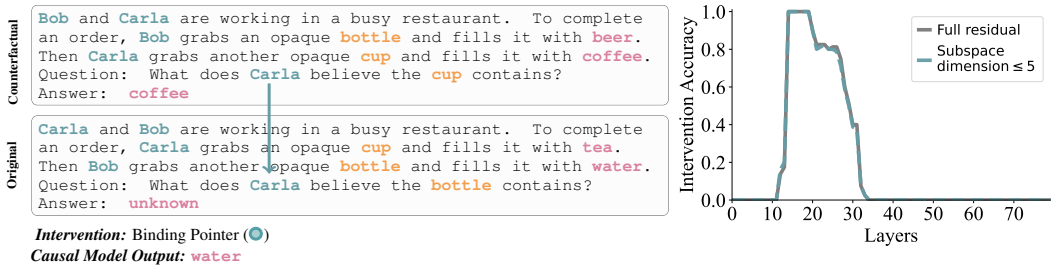

Figure 16: **Query Character OI**: This interchange intervention experiment alters the OI of the queried character () to the other one. Hence, the final output changes from *unknown* to **water**.

I Speculated Payload in Visibility Lookback

As mentioned in section 6, the payload of the Visibility lookback remains undetermined. In this section, we attempt to disambiguate its semantics using the Attention Knockout technique introduced

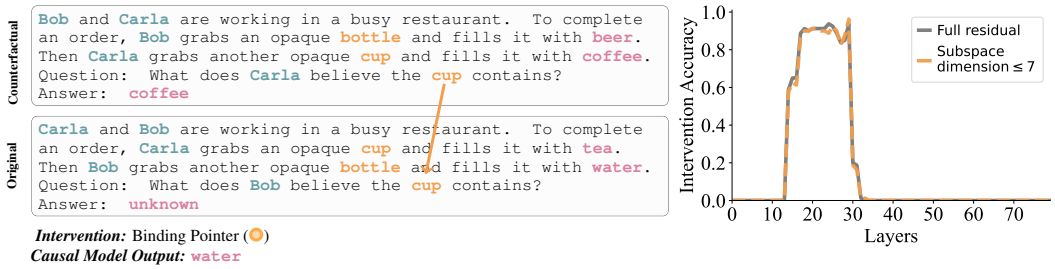

Figure 17: **Query Object OI**: This interchange intervention experiment alters the OI of the queried object ( ) to the other one. Hence, the final output changes from *unknown* to **water**.

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 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.

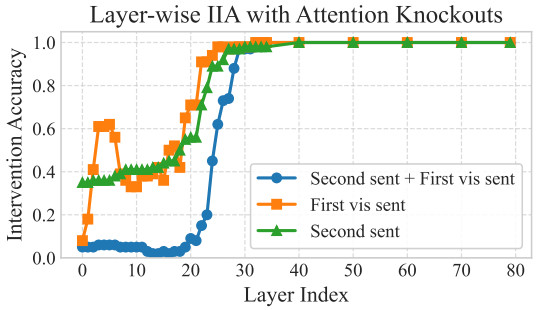

Figure 18: At the second visibility sentence, attention heads are restricted to retrieve information from one of three prior contexts: (1) both the second story sentence and the first visibility sentence (— line), (2) only the first visibility sentence (— line), or (3) only the second story sentence (— line).

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.18 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 G), 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.