A Causal Tracing

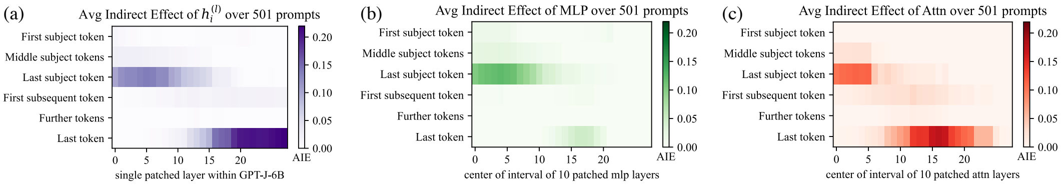

Figure 8: **Causal Tracing** (using the method of Meng et al. 2022). Each grid cell's intensity reflects the average causal indirect effect of a hidden state on the expression of a factual association, with strong causal mediators highlighted with darker colors. We find that MLPs at the last subject token and attention modules at the last token are important. The presence of influential attention activations at the earliest layers of the last subject token is investigated with additional path dependent experiments (Figure 3).

MEMIT begins by identifying MLP layers that are causal mediators for recall of factual associations in the model. To do so in GPT-J, we use code provided by Meng et al. (2022): beginning with a sample of 501 true statements of facts that are correctly predicted by GPT-J, we measure baseline predicted probabilities of each true fact when noise is introduced into encoding of the subject tokens to degrade the accuracy of the model. Then in Figure 8 (a) for each individual h_t^l, we restore the state to the value that it would have had without injected noise, and we plot the average improvement of predicted probability. As in Meng et al. (2022), we use Gaussian noise with standard deviation 3σ (σ^2 is the empirically observed variance of embedding activations) and plot averages for all 501 statements over 10 noise samples. For (b) and (c) we use the same procedure, except we restore runs of 10 layers of MLP outputs m_t^l and 10 layers of Attn a_t^l, instead of full hidden states.

These measurements confirm that GPT-J has a causal structure that is similar to the structure reported by Meng et al. (2022) in their study of GPT2-XL. Unlike with GPT-XL, a strong causal effect is observed in the earliest layers of Attention at the last subject token, which likely reflects a concentrated attention computation when GPT-J is recognizing and chunking the n-gram subject name, but the path-dependent experiment (Figure 3) suggests that Attention is not an important mediator of factual recall of memories about the subject.

In the main paper, Figure 3 plots the same data as Figure 8 (a) as a bar graph, focused on only the last subject token, and it adds two additional measurements. In red bars, it repeats the measurement of causal effects of states with Attention modules at the last subject token frozen in the corrupted state, so that cannot be influenced by the state being probed, and in green bars it repeats the experiment with the MLP modules at the last subject token similarly frozen, so they cannot be influenced by the causal probe. Severing the Attention modules does not shift the curve, which suggests that Attention computations do not play a decisive mediating role in knowledge recall at the last subject token. In contrast, severing the MLP modules reveals a large gap, which suggests that, at layers where the gap is largest, the role of the MLP computation is important. We select the layers where the gap is largest as the range $\mathcal{R}$ to use for the intervention done by MEMIT.

B Implementation Details

B.1 Fine-Tuning with Weight Decay

Our fine-tuning baseline updates layer 21 of GPT-J, which Meng et al. (2022) found to provide the best performance in the single-edit case. Rather than using a hard L_∞-norm constraint, we use a soft weight decay regularizer. However, the optimal amount of regularization depends strongly on the number of edits (more edits require higher-norm edits), so we tune this hyperparameter for the $n = 10{,}000$ case. Figure 9 shows that 5×10^{-4} selects for the optimal tradeoff between generalization and specificity. FT-W optimization proceeds for a maximum of 25 steps with a learning rate of 5×10^{-4}. To prevent overfitting, early stopping is performed when the loss reaches 10^{-2}. Regarding runtime, FT takes $1{,}716.21\,\text{sec} \approx 0.48\,\text{hr}$ to execute $10{,}000$ edits on GPT-J.