Competition of Mechanisms: Tracing How Language Models Handle Facts and Counterfactuals

Anonymous ACL submission

Abstract

Interpretability research aims to bridge the gap between the empirical success and our scientific understanding of the inner workings of large language models (LLMs). However, most existing research in this area focused on analyzing a single mechanism, such as how models copy or recall factual knowledge. In this work, we propose a formulation of *competition of mechanisms*, which instead of individual mechanisms focuses on the interplay of multiple mechanisms, and traces how one of them becomes dominant in the final prediction. We uncover how and where the competition of mechanisms happens within LLMs using two interpretability methods, logit inspection and attention modification. Our findings show traces of the mechanisms and their competition across various model components, and reveal attention positions that effectively control the strength of certain mechanisms.[1]

1 Introduction

Recent advancements in large language models (LLMs) have brought unprecedented performance improvements to NLP (Brown et al., 2020; Touvron et al., 2023; OpenAI, 2023; Anil et al., 2023, *inter alia*). However, the black-box nature of these models obfuscates our scientific understanding of *how these models achieve certain capabilities*, and *how can we trace the problem when they fail at other tasks*. This has brought an increasing focus on interpretability research to help us understand the inner workings of LLMs.

Existing interpretability research has been largely focused on discovering the *existence* of single mechanisms, such as the copy mechanism in induction heads of LLMs (Elhage et al., 2021; Olsson et al., 2022), and factual knowledge recall in the MLP layers (Geva et al., 2021; Meng et al., 2022; Geva et al., 2023). However, different from discovering *what mechanisms exist in LLMs*, we propose a more fundamental question: *how do different mechanisms interact in the decision-making of LLMs?* We show a motivating example in Figure 1, where the model fails to recognize the correct mechanism when it needs to judge between two possible mechanisms: whether to recall the factual knowledge on who developed the iPhone (i.e., Mechanism 1) or to follow its counterfactual redefinition in the new given context (i.e., Mechanism 2).

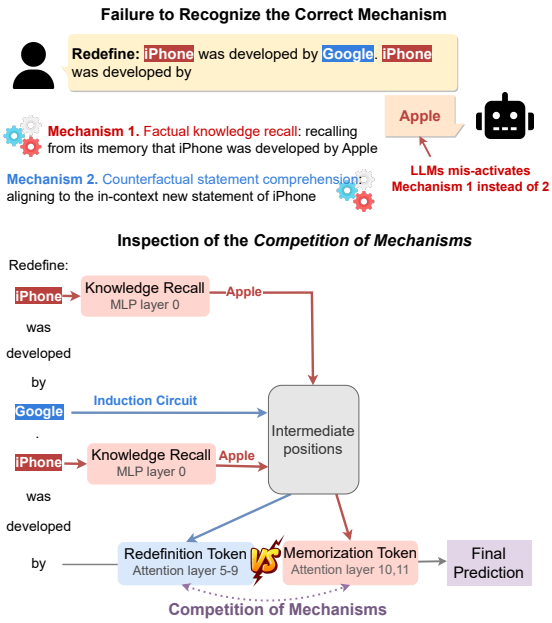

Figure 1: Top: An example showing that LLMs can fail to recognize the correct mechanism when multiple possible mechanisms exist. Bottom: Our mechanistic inspection of where and how the competition of mechanisms takes place within the LLMs.

We propose a novel formulation of *competition of mechanisms*, which focuses on tracing each mechanism in the model, and understanding how one of them becomes dominant in the final prediction by winning the "competition". Specifically, we build our work on two single mechanisms that are

[1] Our code and data have been uploaded to the submission system and will be open-sourced upon acceptance.

well-studied separately in literature: (1) the factual knowledge recall mechanism, which can be located in the MLP layers (Geva et al., 2021; Meng et al., 2022; Geva et al., 2023); and (2) the in-context adaptation to a counterfactual statement, which is enabled by the copy mechanism conducted by induction heads of attention layers (Elhage et al., 2021; Olsson et al., 2022). Based on the latest tools to inspect each of these two mechanisms (Nostalgebraist, 2020; Wang et al., 2023; Geva et al., 2023), we then unfold *how and where* the competition of the two mechanisms happen, and how it leads to the overall success or failure of LLMs.

Technically, we deploy two main methods: logit inspection (Nostalgebraist, 2020; Geva et al., 2022) by projecting the outputs of each model components by an unembedding matrix, and attention modification (Geva et al., 2023; Wang et al., 2023). Using these methods, we assess the contributions of various model components, both from a macroscopic view (e.g., each layer) as well as a microscopic view (e.g., attention heads), and identify critical positions and attention heads involved in the competition of the two mechanisms. Moreover, we locate a few, localized positions of some attention head matrices that can significantly control the strength of the factual mechanism. We summarize our main findings as follows:

1. In early layers, the factual attribute is encoded in the subject position, while the counterfactual is in the attribute position (Sec. 6.1);
2. The attention blocks write most of the factual and counterfactual information to the last position (Sec. 6.2);
3. All the highly activated heads attend to the attribute position regardless of the specific type of information they promote. The factual information flows by penalizing the counterfactual attribute rather than promoting the factual one (Sec. 6.3);
4. We find that we can up-weight the value of a few, very localized values of the attention head matrix to strengthen factual mechanisms substantially (Sec. 6.4).

2 Related Work on Interpretability

As deep learning approaches show increasingly impressive performance in NLP, their black-box nature has hindered the scientific understanding of these models and their effective future improvements. To this end, interpretability research has been a rising research direction to understand the internal workings of these models.

Interpreting the Representations. One major type of work in interpretability has focused on understanding what has been encoded in the representations of deep learning models. This is usually achieved by a *probe*, namely by training a supervised classifier to predict features from its representations (Alain and Bengio, 2016; Conneau et al., 2018; Hupkes et al., 2018; Hewitt and Liang, 2019; Tenney et al., 2019; Jiang et al., 2020; Elazar et al., 2021, *inter alia*). Example features of interest include part of speech (Belinkov et al., 2017), verb tense (Conneau et al., 2018), syntax (Hewitt and Manning, 2019), and factual knowledge (Petroni et al., 2019).

Interpreting the Learned Mechanisms/Functions. Beyond interpreting the representations in the hidden states of the black-box models, another research direction is to interpret the mechanisms or functions that the models have learned, giving rise to the field of mechanistic interpretability (Olah et al., 2020; Elhage et al., 2021; Olsson et al., 2022; Nanda et al., 2023, *inter alia*). Some example mechanisms decoded in recent work include: mathematical operations such as modular addition (Nanda et al., 2023) and the greater-than operation (Hanna et al., 2023); natural language-related operations such as the copy mechanism achieved by induction heads in LLMs Olsson et al. (2022) and factual knowledge recall achieved by MLP layers (Geva et al., 2021; Meng et al., 2022; Geva et al., 2023), which we describe below.

The Single Mechanism of Copy: One of the basic actions in LLMs is the copy mechanism, which is found to be operationalized by attending to the copied token in the attention heads and passing it on to the next token prediction (Elhage et al., 2021; Olsson et al., 2022). This foundational mechanism enables further research to decode more complex mechanisms, such as indirect object identification (Wang et al., 2023).

The Single Mechanism of Factual Knowledge Recall: Another major direction is understanding how LLMs mechanistically recall factual information (Geva et al., 2021; Meng et al., 2022; Geva et al., 2023). For example, Meng et al. (2022) develop the *causal tracing* method to show that the factual

information is found in the mid-layer MLP units in GPT-2. A followup work (Geva et al., 2023) shows that MLPs of early layers enrich the subject embeddings with related attributes, and late attention blocks select and write the correct factual information to the sentence's last position.

Interplay of Multiple Mechanisms: In the final stage of our project in December 2023, we noticed a related study by Yu et al. (2023), which also investigates the role of two different mechanisms in LLMs. Specifically, they inspect a type of prompt whose subjects are the capital cities and whose attributes are the countries, examine the dynamics of the factual recall mechanism and the effect of the in-context counterfactual statement, and find that the subject and attribute frequency in the pretraining set can affect the ability of factual recall. Differently, the methods in our work are applied to a broader set of prompts; moreover, we also establish novel analyses of the underlying mechanistic details of the competition and precisely localize the path where the information flows at the level of single attention map activations, based on which we discover new findings that are unique to our study.

3 Problem Setup

Following the setup of many existing interpretability studies (Olah et al., 2020; Elhage et al., 2021; Olsson et al., 2022; Nanda et al., 2023, *inter alia*), we look into the next token prediction behavior of autoregressive LLMs in their inference mode, namely

$$P(t_k|t_{<k}), \tag{1}$$

which predicts the k-th token t_k given all the previous tokens in the context.

Next, we design the task to incorporate the competition of mechanisms as in Figure 1. Specifically, for each factual statement $\boldsymbol{f} := (t_1^f, \ldots, t_k^f)$ consisting of k tokens (e.g., "iPhone was developed by Apple"), we compose a corresponding counterfactual statement $\boldsymbol{c} := (t_1^c, \ldots, t_{k'}^c)$ (e.g., "iPhone was developed by Google"). Then, we compose a prompt connecting the two statements as "Redefine: $\boldsymbol{c}$. $\boldsymbol{f}_{1:k-1}$.", such as *"Redefine: iPhone was developed by Google. iPhone was developed by ___"*.

The two mechanisms can be traced by inspecting the rise and fall of the factual token t_k^f and the counterfactual token $t_{k'}^c$. For the simplicity of notation, we take the tokens out of the context of their exact position and denote them as t_{fact} and t_{cofa}, respectively, in the rest of the paper.

4 Method and Background

Method 1: Logit Inspection. To inspect the inner workings of the two mechanisms, we trace the *residual stream* (Elhage et al., 2021), or logits of each component in the LLM. Given a text sequence of k tokens, LLMs maps it into the residual stream, namely a matrix $\mathbf{x} \in \mathbb{R}^{d \times k}$, where d is the dimension of the internal states of the model. We use the term $\mathbf{x}_i^l$ to specify the residual stream at position i and layer l.

An LLM produces the initial residual stream $\mathbf{x}_i^0$ by applying an embedding matrix $W_E \in \mathbb{R}^{|V| \times d}$ to each token t_i, where $|V|$ is the size of the vocabulary. Then, it modifies the residual stream by a sequence of L layers, each consisting of an attention block $\mathbf{a}^l$ and MLP $\mathbf{m}^l$. Finally, after the last layer, it projects the internal state of the residual stream back to the vocabulary space with an unembedding matrix $W_U \in \mathbb{R}^{d \times |V|}$. Formally, the update of the residual stream at the lth layer is:

$$\mathbf{x}^l = \mathbf{x}^{l-1} + \mathbf{a}^l + \mathbf{m}^l \ , \tag{2}$$

where both the attention and the MLP block take as input the $\mathbf{x}$ after layer normalization norm:

$$\mathbf{a}^l = \mathbf{a}^l(\text{norm}(\mathbf{x}^{l-1})) \ , \tag{3}$$

$$\mathbf{m}^l = \mathbf{m}^l(\text{norm}(\mathbf{x}^{l-1} + \mathbf{a}^l)) \ . \tag{4}$$

To understand which token the residual stream $\mathbf{x}^l$ favors, we follow the common practice in previous work (Halawi et al., 2023; Geva et al., 2023; Dar et al., 2023; Geva et al., 2022) to project it to the vocabulary space using the aforementioned unembedding matrix W_U which maps the latent embeddings to actual tokens in the vocabulary, enabling us to obtain the logits of the factual t_{fact} and counterfactual token t_{cofa}.

Method 2: Attention Modification. Modifying or ablating the activation of a specific model component is also a strategy used to improve the understanding of the information flow within LLMs, including techniques such as causal tracing (Meng et al., 2022) and attention knockout (Wang et al., 2023; Geva et al., 2023). In our work, we mainly

modify the attention matrix A^{hl} of a certain attention head of the attention layer $\mathbf{a}^l$. Formally, in the attention head h at layer l, we can force-set the attention weight of a token $\mathbf{x}_i^l$ to one of its earlier tokens $\mathbf{x}_j^l$ to be α:

$$A_{ij}^{hl} = \alpha, \quad \text{where } j < i\,. \tag{5}$$

5 Experimental Setup

Data Creation To compose the factual and counterfactual statements as introduced in Sec. 3, we adopt COUNTERFACT (Meng et al., 2022), commonly used dataset to interpret models' ability of factual knowledge recall. We select 10K data points by considering only examples where the attributes are represented by a single token and where the model completes the sentence correctly. See our data composition details in Appendix A.

Models We follow most existing interpretability studies (e.g., Meng et al., 2022; Wang et al., 2023; Conmy et al., 2023; Hanna et al., 2023) to use the autoregressive transformer, GPT-2 small (Radford et al., 2019) has 117M parameters, consisting of 12 layers with 12 self-attention heads each and a residual stream of 768 dimensions. We present our main results using GPT-2 small and also provide supplemental results of a larger model, Pythia-6.9B (Biderman et al., 2023) in Appendix C, to show the robustness of our findings across LLMs of different scale. In particular, since Pythia-6.9B has 32 layers with 32 self-attention heads each and a model dimension of 4096, results for this model reflect the consistency of our findings over a 30x increase in the number of parameters. We introduce our experimental details in Appendix B.

6 Results and Findings

In this section, we trace the competition of the mechanisms within the LLM via the two methods introduced in Sec. 4, i.e., inspecting the residual stream and intervening on the attention. We provide mechanistic analyses on five research questions in the following subsections:

1. Macroscopic view: In which layers do the competition of the mechanisms take place? (Sec. 6.1)
2. Intermediate view: How do we attribute the prediction to attention and MLP blocks? (Sec. 6.2)
3. Microscopic view: How do individual attention heads contribute to the prediction? (Sec. 6.3)
4. Intrinsic intervention: Can we modify the strength of a certain mechanism? (Sec. 6.4)
5. Behavioral intervention: What word choice varies the strength of the counterfactual mechanism in the given context? (Sec. 6.5)

6.1 Where Do the Two Mechanisms Take Place?

In the main model that we inspect, GPT-2, we find that it can usually identify the correct mechanism, the counterfactual mechanism, in 96% of the 10K test examples. This means that, in the last sequence position, at the output of the network, the counterfactual token, t_{cofa}, gets most of the times a higher probability than t_{fact}. In the following, we will inspect how the "winning" of the counterfactual mechanism happens across the layers of the LLM.

Methods. We study how t_{fact} and t_{cofa} are encoded in the residual stream using the first method described previously in Sec. 4. Specifically, for a given position i and a layer l, we project the embedding $\mathbf{x}_i^l$ (see Eq.(2)) to the vocabulary space using the last layer unembedding matrix $\tilde{\mathbf{x}}_i^l = W_U \mathbf{x}_i^l$. By varying l, we measure the value of the logit of t_{fact} and t_{cofa} as they evolve in the residual stream after the first attention block.

Results. The upper panel of Fig. 2-left shows the logit of the factual token, t_{fact}. In the early layers, Logit(t_{fact}) is higher in the subject positions, and in late layers, its prevalence shifts to the last token of the sentences. A high logit value in the last position of late layers is expected, as this position is used to predict the subject attribute. Consistently with Geva et al. (2023), we also found that the factual attribute is written in the subject positions by the first MLP, which increases on average the value of t_{fact} from 3.3 to 13.1 in the premise and from 9.7 to 13.5 in the second sentence. The evolution of t_{cofa}'s logits is shown in the lower panel. In this case, t_{cofa} is best encoded in the attribute position in early layers. Indeed, after the first attention block, the average logit value of t_{cofa} is 19.8, and it is, in most of the examples, the highest in the $50k$-dimensional vocabulary of GPT-2.

As we proceed towards the output, the rank of t_{cofa} in the projected logit distribution remains very low: t_{cofa} is among the 20 most likely tokens in the first

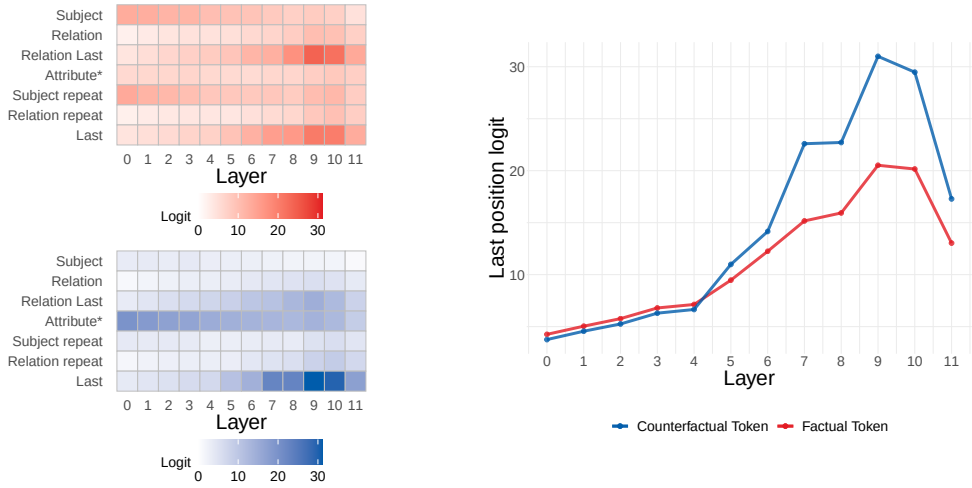

Figure 2: **Analysis of the key positions of the factual and counterfactual in GPT-2**. This figure displays the logit values for two significant tokens across different positions and layers. The upper left panel shows the logit values for the factual token t_{fact}. Logit(t_{fact}) is higher in the subject position in the initial layers and in the last position of the premise and second sentence in the final layers. The lower right panel shows the logit of the counterfactual token t_{cofa}. Logit(t_{cofa}) is higher in the attribute position in the first layers and in the last position of the second sentence at the end of the network. The right panel shows the average logits of t_{fact} (red) and of t_{cofa} (blue) in the last sequence position. At the output of the network Logit(t_{cofa}) > Logit(t_{fact}) in 96% of the examples.

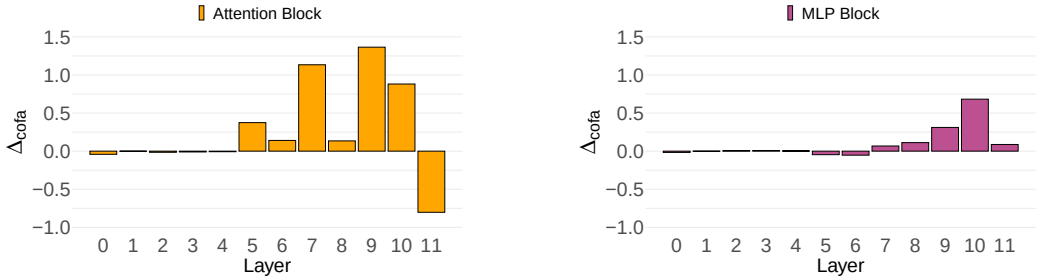

Figure 3: **Contributions of attention and MLP blocks to the logits of t_{fact} and t_{cofa} in the last sequence position.** The figure shows the contribution of attention and MLP blocks to the logit difference $\Delta_{\text{cofa}} = \text{Logit}(t_{\text{cofa}}) - \text{Logit}(t_{\text{fact}})$ at $W_U \mathbf{a_N}^l$ (resp. $W_U \mathbf{m_N}^l$) for the input's last position across layers. The attention blocks (left) contribute more significantly than the MLP blocks (right). Only the attention block at layer 11 favors t_{fact}.

five layers and between the 20th and the 70th in the last part of the network (see Appendix, Fig. S5). Figure 2-right shows that after four attention blocks, the logit of the counterfactual attribute grows markedly in the last position of the second sentence, becoming higher than Logit(t_{cofa}) and reaching its peak value of 30 at layer 10. In the last layer, the logit of t_{cofa} slightly decreases but remains larger than that of t_{fact} in 96% of the examples.

The pattern we have described so far shows that the information about t_{fact} and t_{cofa} is brought to the last position after 6 or 7 layers (see Fig. 2-right). We have also seen that the most activated positions at early layers are the subjects for t_{fact} and the attribute for t_{cofa}. The information flow between different sequence positions suggests a major role played by the attention mechanism in moving such information to the last position, as observed in Geva et al. (2023). We continue our analysis studying the role of the self-attention and MLP blocks in writing the information about t_{fact} and t_{cofa}, and by understanding which, between subject and attribute positions, is the main source of factual information written to the last position in the final layers.

6.2 Counterfactual Contributions of Attention and MLP Blocks

Methods. To understand the main source of factual and counterfactual information introduced in the last position, we apply the logit inspection method to the output of the attention and MLP

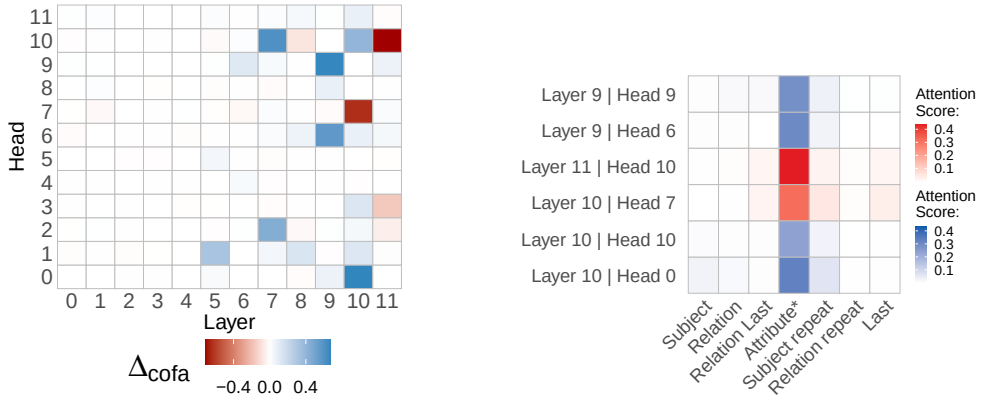

Figure 4: **Attention pattern for relevant attention heads.** Left: Direct contribution to Δ_{cofa} of all heads in GPT-2. Heads favoring t_{fact} are colored in red, and those favoring t_{cofa} in blue. Right: attention scores of relevant heads for the last position. A large attention score in the attribute position is found in all highly activated heads.

blocks before they are added to the residual stream. We analyze the values of t_{fact} and t_{cofa} in the the logit distribution at $W_U \mathbf{a_N}^l$ and $W_U \mathbf{m_N}^l$, where N denotes the last token in the sequence. A positive value of $\Delta_{\text{cofa}} = \text{Logit}(t_{\text{cofa}}) - \text{Logit}(t_{\text{fact}})$ at $W_U \mathbf{a_N}^l$ (resp. $W_U \mathbf{m_N}^l$) means that the attention map (resp. MLP) is increasing the likelihood of the counterfactual attribute in the last position at layer l relative to the factual one.

Results. Figure 3 compares the Δ_{cofa} values added to the residual stream by the attention and MLP blocks at all the GPT-2-small layers averaged on the whole dataset. The left panel shows the contribution to Δ_{cofa} of the attention blocks summing over all the attention heads. Until layer 4, Δ_{cofa} is approximately zero, meaning that the attention maps do not add information about either t_{fact} or t_{cofa}. From layers 5 to 10, the value of Δ_{cofa} increases, reaching a peak of about 1.4 at layer 9. The last layer is the only one that favors t_{fact} with a sizeable negative value of Δ_{cofa} around -0.8. In the next section, we will see that the decrease of Δ_{cofa} in the last two layers is mainly due to the contribution of two attention heads, L10H7 and L11H10, which explain most of the total negative contribution to Δ_{cofa}.

The right panel shows the evolution of Δ_{cofa} for the MLP blocks. Similar to the trend observed for the attention blocks, this contribution remains close to zero in the early layers of the networks and grows only after layer 8. The total contribution of the MLPs to Δ_{cofa} is positive but much smaller than the one given by the attention blocks and significantly different from zero only at layers 9 and 10, where it reaches its maximum value around 0.75. Attention blocks and MLPs behave consistently across the network, favoring the counterfactual attribute in all the layers except for the attention contribution in the last layer. The attention blocks are in charge of adding most of the information about t_{cofa} to the residual stream of the last position, consistently with the findings of Geva et al. (2023). The next section delves into the analysis of the attention heads to better understand their behavior. In particular, we will investigate if some heads have a special role in favoring factual or counterfactual tokens and identify the position from which they primarily extract information.

6.3 Effects of Attention Heads

In this section, we study the role played by each attention head in promoting the factual vs counterfactual token. We will find that the behavior of the attention heads of the final layers is highly specialized, and the inspection of the heads that contribute most to Δ_{cofa} allows finding from which position most of the information is brought to the final one. We finally show that we can significantly increase the factual predictions by simply up-weighting the few specialized attention weights where most factual information flows.

Methods. We analyze the attention heads separately with the logit inspection method by projecting the outputs of each attention head to the last sequence position N in the vocabulary space. More precisely, we consider $\Delta_{\text{cofa}} = \text{Logit}(t_{\text{cofa}}) - \text{Logit}(t_{\text{fact}})$ where the logits are taken from the projection $W_U \mathbf{a_N}^{h,l}$ for each head h.

Results. Figure 4 (left) shows on the y-axis the individual contribution of each head to Δ_{cofa} averaged over the dataset. On the x-axis, we plot the layers of the network. A blue cell indicates a positive Δ_{cofa} or that a given head is favoring t_{cofa}. Conversely, a red cell indicates a negative Δ_{cofa} or that a given head is favoring t_{fact}. Figure 4 (left) shows that few heads are responsible for promoting or suppressing the copy of t_{cofa} to the last position. For example, the sum of L7H2 and L7H10 equals 75% of the large positive Δ_{cofa} contribution of layer 7. The sum of L9H6 and L9H9 explains 65% of the Δ_{cofa} at layer 9. In layers 10 and 11, the decrease of Δ_{cofa} shown in Fig. 3 (left) is almost entirely due to the action of two heads: L10H7 and L11H10, which alone explain almost the 70% of the total negative contribution to Δ_{cofa} in the entire network (33% and 37% respectively). McDougall et al. (2023) showed that these two heads are responsible for suppressing the copy mechanisms in GPT-2-small. In our setting, the joint ablation of these two heads decreases the factual recall of GPT-2-small from 4.13% to 0.65%.

Most interestingly, all highly activated heads attend to the same position. Figure 4 (right) shows the last rows of the most activated attention maps (we report the full maps in Figure S6 of the Appendix). Even though each attention head may have a different role in supporting either t_{fact} or t_{cofa}, they all strongly attend to the attribute position. This is expected for the heads that support the copy mechanism since they read from the sequence position where t_{cofa} is better represented (see Sec. 6.1), while it is more surprising for the heads that favor the factual token. Indeed, an attention head can favor a factual response by increasing the logit of t_{fact} or decreasing the logit of t_{cofa} or both. In this case, L10H7 and L11H10 read from the attribute position because it is easier to give a lower value to the logit of t_{cofa} without at the same time penalizing too much t_{fact}. In these two heads, the logit of t_{fact} written to the last sequence position is smaller than the mean of the two layers, but the logit t_{cofa}, -1.13 for L10H7 and -1.05 for L11H10, are by far the lowest of all the heads in the network. This picture also holds true for larger GPT-2 transformers and Pythia: in all cases, the most activated heads have a strongly activated attention edge between the attribute and the last position, including those favoring t_{fact} (see Fig. S3 of the Appendix).

6.4 Improving the Factual Recall

Because the factual mechanism is promoted by only a few strongly activated heads (L10H7 and L11H10 in GPT-2, and mostly L17H28, L20H18, and L21H8 in Pythia, see Fig. S4) and because most of the information flows from the attribute position (see Fig. 4-right and Sec. 6.2), enhancing the weight of this path will likely improve the factual recall of the model. In this section, we show that enhancing the value of a few well-localized attention edges can largely improve the factual recall of the model.

Methods. To this end, we multiply by a common factor α the attribute position in the last row of L10H7 and L11H10 for GPT-2, and L17H28, L20H18, and L21H8 in Pythia (see Appendix, Fig. S4). We search the optimal value of α on the grid $[2, 5, 10, 100]$.

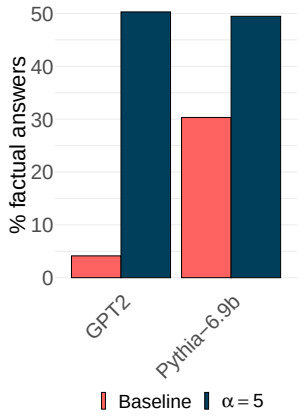

Figure 5: **Boosting factual recall with attention score modification.** The figure shows the increase in factual recall for GPT-2 and Pythia-6.9b, achieved by increasing the attention score of the attribute position in the last row of the attention map by a factor of 5 for the L10H7 and L11H10 in GPT-2 and L17H28, 498 L20H18, and L21H8 in Pythia.

Results. Figure 5 shows the gain of factual recall obtained using $\alpha = 5$ for both GPT-2 for Pythia. The factual error rate decreases from 96% to 50% in GPT-2 and from 70% to 50% in Pythia. This result is remarkable since we modify only two entries in the attention map out of the 117M parameters of GPT-2 and three entries out of the 6.9B parameters of Pythia. This highlights the importance of the analysis of Sec. 6.3 and Sec. 6.2 in finding the detailed role played by the individual units of each transformer layer.

6.5 What Word Choices Intensify the Competition?

We next focus on understanding how the similarity between t_{fact} and t_{cofa} in our dataset affects the mechanism described in the previous sections.

Methods. We measure the similarity of the attributes using a pre-trained Word2Vec model (Mikolov et al., 2013) from Gensim Library (Řehůřek and Sojka, 2010), which generates 300-dimensional word vectors.

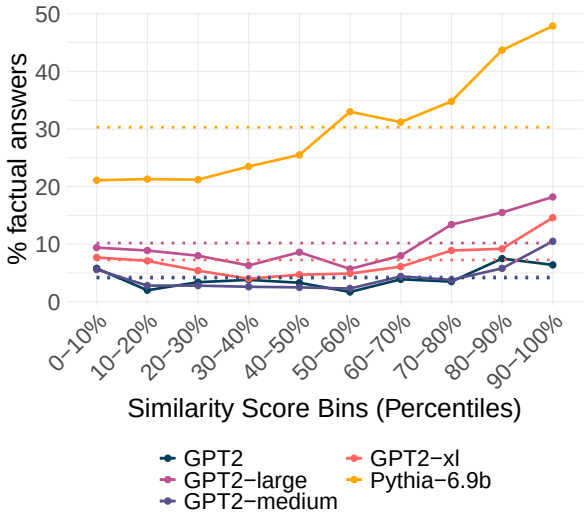

Figure 6: **Prediction Frequency of Factual Token by Similarity Level**. We show the percentage of t_{fact} predictions within each bin compared to the entire dataset (represented by a dotted line) across various model sizes. Each bin has a different range of cosine similarity between t_{fact} and t_{cofa}. We can notice that more similar t_{fact} and t_{cofa} are, and the factual mechanism is stronger.

Results. We divide the dataset into ten equally sized subsets according to the cosine similarity between the vectors representing t_{fact} and t_{cofa}. We then measure the frequency of predicting t_{fact} within each subset.

Figure 6 shows the outcomes across various model sizes. We can see that the factual recall is higher when t_{fact} and t_{cofa} are more similar. This trend is observed across all model sizes, with larger models generally exhibiting a stronger correlation and a higher overall propensity for predicting the factual token. This suggests that larger models are more sensitive to the similarity between t_{cofa} and t_{fact} and are more likely to rely on factual recall, even when the similarity is not exceptionally high. This finding aligns with the notion that larger models have a greater capacity to store and retrieve factual information.

7 Discussion and Future Work

To this point, we have analyzed in detail the mechanism by which LLMs handle the competition between facts learned during training and counterfactual information found in-context. Our findings are consistent with Geva et al. (2023), showing that late attention blocks write most of the information to the last layer when adding a counterfactual premise. Surprisingly, however, we found that the largest contribution to the factual prediction of the network mostly comes from the suppression of the counterfactual token read from the attribute position rather than the promotion of the factual token from the subject position. Consistently with McDougall et al. (2023), we found that few highly specialized heads suppress the counterfactual information. Moreover, we also found that up-weighting only two or three attention entries of these heads is enough to increase substantially the number of factual responses of the network. With an approach similar to ours, Yu et al. (2023) found that more heads can promote the factual mechanism, also in early layers, but found it challenging to improve the factual responses by scaling up the weights of the attention maps. This discrepancy can be due to the broader set of topics we include in our prompts, which allowed us to select fewer, more specialized heads, to the different ways the prompts are framed, or also to our more focused modification of the attention maps. As future research directions, we aim to analyze more in-depth how our findings depend on the prompt structure and whether the promotion of factual responses by suppressing the counterfactuals generalizes to larger models and a more comprehensive variety of datasets.

8 Conclusion

In this work, we have proposed the formulation of the *competition of algorithms* as a powerful interpretation when LLMs need to handle multiple mechanisms, only one of which leads to the correct answer. We deployed two mechanistic interpretability tools, logit inspection and attention modification, and identified critical positions and model components involved in competing for the mechanisms. Finally, we discovered a few localized positions in the attention map, which largely control the strength of the factual mechanism. Our study sheds light on future work on interpretability research for LLMs.

Limitations

Our focus was mainly on GPT-2 models, particularly GPT-2-small, which possess considerably fewer parameters than current state-of-the-art models. Furthermore, our experiments and insights are heavily grounded in the interpretability within the embedding space of the model's inner components. This approach is reliable and extensively employed in mechanistic interpretability research (Dar et al., 2023; Geva et al., 2022; Halawi et al., 2023). The logit inspection method, although commonly employed in previous work, can occasionally fail to reflect the actual importance of some vocabulary items, especially in the early layers of the network (Belrose et al., 2023). Finally, our research notes that the competitive dynamics intensify with increasing model size. However, as Wei et al. (2023) points out, this trend may exhibit a U-shaped curve, decreasing beyond a specific model size threshold.

Ethical Considerations

The aim of our study is to enhance comprehension of the interplay among mechanisms within language models that may yield unforeseen and undesirable outcomes. Additionally, our research serves as a conceptual demonstration of methods to guide model behavior under such conditions. We believe that recognizing and dissecting the mechanisms by which LLMs produce unpredictable responses is crucial for mitigating biases and unwanted results. Moreover, understanding the competitive dynamics under investigation is critical for improving the safety of LLMs. Specifically, inputting a prompt with an inaccurate redefinition may lead the model to inadvertently reveal sensitive factual information.

References

Guillaume Alain and Yoshua Bengio. 2016. Understanding intermediate layers using linear classifier probes. *ArXiv*, abs/1610.01644. 2

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul Ronald Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, and et al. 2023. Gemini: A family of highly capable multimodal models. *CoRR*, abs/2312.11805. 1

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–872, Vancouver, Canada. Association for Computational Linguistics. 2

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *CoRR*, abs/2303.08112. 9

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 2397–2430. PMLR. 4

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. 1

Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. *CoRR*, abs/2304.14997. 4

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics. 2

Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. 2023. Analyzing transformers in embedding space. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

Papers), ACL 2023, Toronto, Canada, July 9-14, 2023, pages 16124–16170. Association for Computational Linguistics. 3, 9

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard H. Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Trans. Assoc. Comput. Linguistics*, 9:1012–1031. 2

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2021/framework/index.html. 1, 2, 3

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. *CoRR*, abs/2304.14767. 1, 2, 3, 4, 5, 6, 8

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics. 2, 3, 9

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. 1, 2

Danny Halawi, Jean-Stanislas Denain, and Jacob Steinhardt. 2023. Overthinking the truth: Understanding how language models process false demonstrations. *CoRR*, abs/2307.09476. 3, 9

Michael Hanna, Ollie Liu, and Alexandre Variengien. 2023. How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. *CoRR*, abs/2305.00586. 2, 4

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China. Association for Computational Linguistics. 2

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics. 2

Dieuwke Hupkes, Sara Veldhoen, and Willem H. Zuidema. 2018. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *J. Artif. Intell. Res.*, 61:907–926. 2

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438. 2

Callum McDougall, Arthur Conmy, Cody Rushing, Thomas McGrath, and Neel Nanda. 2023. Copy suppression: Comprehensively understanding an attention head. *CoRR*, abs/2310.04625. 7, 8

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *arXiv preprint arXiv:2202.05262*. 1, 2, 3, 4, 12

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119. 8

Neel Nanda and Joseph Bloom. 2022. Transformerlens. `https://github.com/neelnanda-io/TransformerLens`. 12

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net. 2, 3

Nostalgebraist. 2020. interpreting gpt: the logit lens. Accessed: Nov 2023. 2

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. 2020. Zoom in: An introduction to circuits. *Distill*, 5(3):e00024–001. 2, 3

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. 2022. In-context learning and induction heads. *CoRR*, abs/2209.11895. 1, 2, 3

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774. 1

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases?

In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics. 2

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8). 4

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en. 8

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. *ArXiv*, abs/1905.06316. 2

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971. 1

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2023. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net. 2, 3, 4

Jason Wei, Najoung Kim, Yi Tay, and Quoc V. Le. 2023. Inverse scaling can become u-shaped. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 15580–15591. Association for Computational Linguistics. 9

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics. 12

Qinan Yu, Jack Merullo, and Ellie Pavlick. 2023. Characterizing mechanisms for factual recall in language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9924–9959, Singapore. Association for Computational Linguistics. 3, 8

A Data Composition Details

We use the COUNTERFACT dataset (Meng et al., 2022) from https://rome.baulab.info/data/. An instance of COUNTERFACT expresses a relation r between a subject s and an attribute a: (s, r, a). For instance, in the sentence *"iPhone was developed by Apple"*, s = *"iPhone"*, r = *"was developed by"*, a = *"Apple"*. Each (s, r) pair is associated with a correct fact, e.g., a^c = "*Apple*", and a false fact, a^* = "*Google*". We aim to study how a prompt containing counterfactual information affects model prediction. For this purpose, we start by considering only triples (s, r, a) in COUNTERFACT whose correct fact a^c is represented by a single token, and for which the model correctly predicts a^c when asked to complete the sentence $(s, r, _)$. After selecting $10,000$ instances (s, r, a) with such properties among the 219180 triples of COUNTERFACT, we construct our final dataset that is composed of instances $(s, r, a^*, s, r, _)$. As a result, the dataset we consider is constituted of 10,000 counterfactual prompts. For example, the counterfactual prompt corresponding to the sentence *"iPhone was developed by Apple"* is *"Redefine: iPhone was developed by Google. iPhone was developed by ___."*.

B Experimental Details

We deploy the pre-trained GPT-2 and Pythia models from the Huggingface Hub (Wolf et al., 2020), and inspect the residual streams by the LogitLens tool in the TransformerLens library (Nanda and Bloom, 2022).

C Experiments for Pythia-6.9b

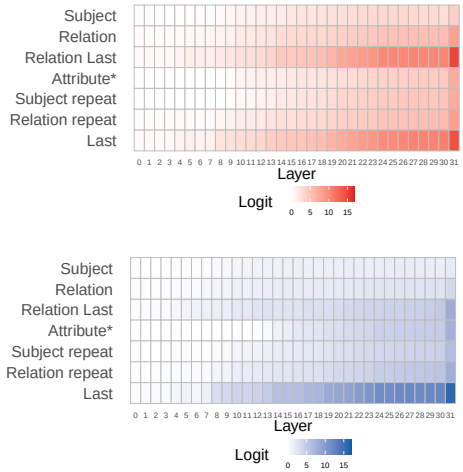

Figure S1: **Layer-wise Position Analysis of Relevant Tokens in GPT-2-small**. The figure presents the logit values for two pertinent tokens across various positions and layers. The left panel illustrates the logit values for the factual token t_{fact}, while the right panel illustrates the logit values for the counterfactual token t_{cofa}.

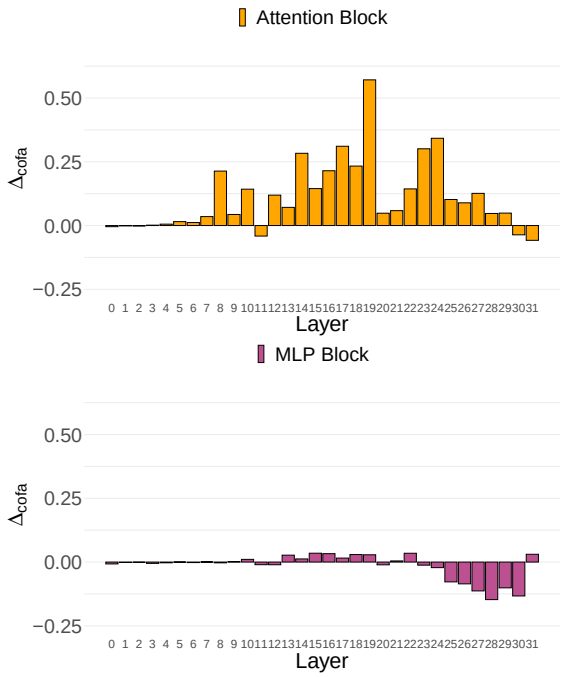

Figure S2: **Attribution of Logit Differences to Attention and MLP Blocks.** This figure depicts the role of Attention and MLP Blocks in influencing the logit difference $\Delta_{\text{cofa}} = \text{Logit}(t_{\text{cofa}}) - \text{Logit}(t_{\text{fact}})$ at $W_U \mathbf{a_N}^l$ for the final position of the input across various layers. It is observed that the Attention Blocks have a more pronounced impact compared to the MLP Blocks. Additionally, the right panel indicates a contribution from the MLP Blocks at the last layers towards the factual token t_{fact}.

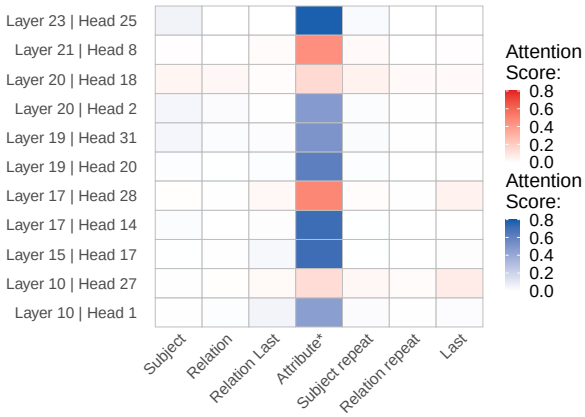

Figure S3: **Attention Pattern for Relevant Attention Heads.** The panel illustrates the attention patterns of relevant heads for the last position, demonstrating consistent attention to the attribute position by both red and blue heads.

D Other Experiment for GPT-2

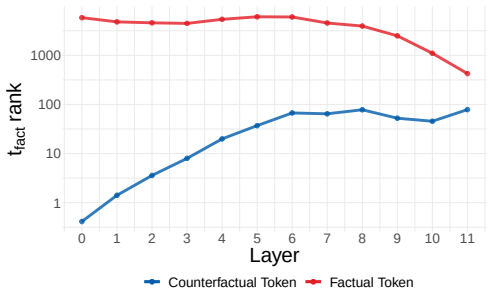

Figure S5: **Rank of Target Tokens for Attribute Position Across Layers in GPT-2.** This figure depicts the trend where the logit rank for the factual token t_{fact} decreases while the rank for the counterfactual token t_{cofa} increases at the attribute position. In the concluding layers, this pattern is evident as t_{fact} typically secures a lower rank, in contrast to t_{cofa}, which shows an upward trajectory in rank. However, it is important to note that t_{cofa}'s rank consistently remains lower than that of t_{fact}.

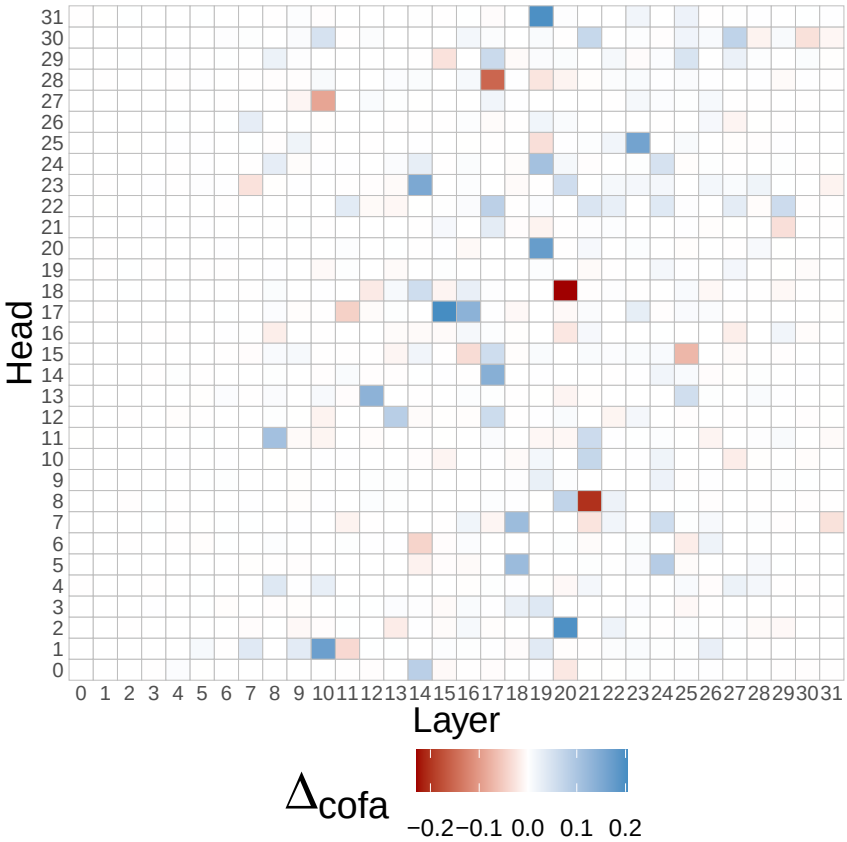

Figure S4: **Direct Contribution of Attention Heads.** The figure displays the direct contribution of all heads in Pythia-6.9b to the logit difference Δ_{cofa} with heads favoring t_{fact} highlighted in red and those favoring t_{cofa} in blue.

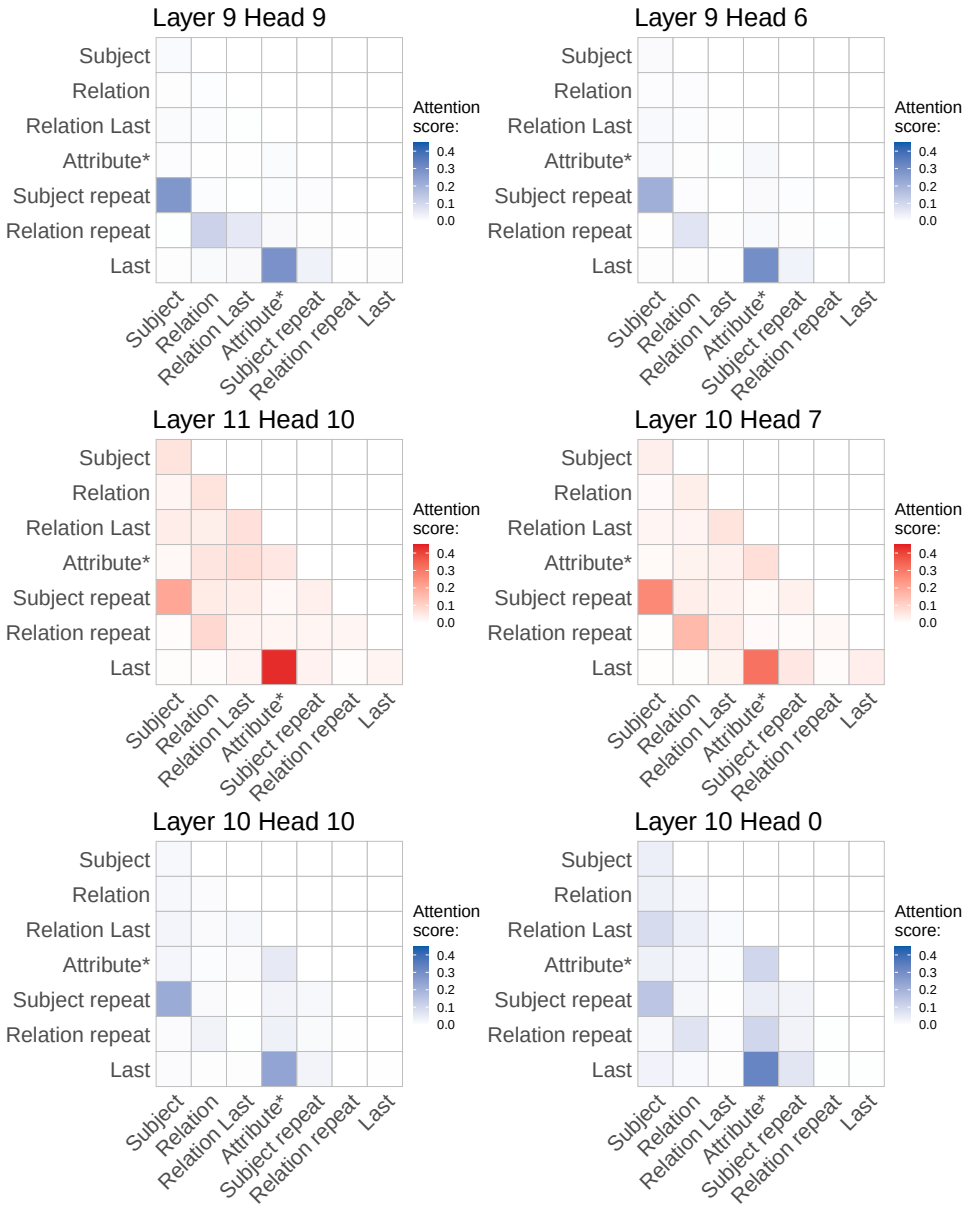

Figure S6: **Attention Pattern of Significant Heads.** This figure illustrates the comprehensive attention pattern of heads substantially influencing $\Delta_{t_{\text{cofa}}}$. Notably, a similar pattern emerges for both heads favoring t_{cofa} (depicted in blue) and those favoring t_{fact} (illustrated in red), particularly in the attention edge between the attribute and the final position.