Relation Also Knows: Rethinking the Recall and Editing of Factual Associations in Auto-Regressive Transformer Language Models

Xiyu Liu[1,2], Zhengxiao Liu[1,2*], Naibin Gu[1,2], Zheng Lin[1,2*], Wanli Ma[3], Ji Xiang[1], Weiping Wang[1]

[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3]University of Electronic Science and Technology of China, Chengdu, China
{liuxiyu, liuzhengxiao, gunaibin, linzheng, xiangji, wangweiping}@iie.ac.cn, uestc_wlma@163.com

Abstract

The storage and recall of factual associations in auto-regressive transformer language models (LMs) have drawn a great deal of attention, inspiring knowledge editing by directly modifying the located model weights. Most editing works achieve knowledge editing under the guidance of existing interpretations of knowledge recall that mainly focus on subject knowledge. However, these interpretations are seriously flawed, neglecting relation information and leading to the *over-generalizing* problem for editing. In this work, we discover a novel relation-focused perspective to interpret the knowledge recall of transformer LMs during inference and apply it on single knowledge editing to avoid over-generalizing. Experimental results on the dataset supplemented with a new R-Specificity criterion demonstrate that our editing approach significantly alleviates over-generalizing while remaining competitive on other criteria, breaking the domination of subject-focused editing for future research.

Code — https://github.com/sunshower-liu/RETS
Extended version — https://arxiv.org/abs/2408.15091

Introduction

Language models are often regarded as knowledge bases, storing factual associations in parameters which can be simply recalled through prompting (Petroni et al. 2019; Lester, Al-Rfou, and Constant 2021; Jiang et al. 2020; Roberts, Raffel, and Shazeer 2020; Petroni et al. 2020; Heinzerling and Inui 2021; Wang, Liu, and Zhang 2021). For instance, for the factual association shown in triplet <*Marco Reus*, *citizen-of*, **O**> with the subject *Marco Reus* and the relation *citizen-of*, the object **O** can be obtained from the next token prediction of GPT-like language models given the prompt "*Marco Reus is a citizen of*". Recent works investigate where factual knowledge is stored and how the factual knowledge is extracted from auto-regressive transformer LMs, suggesting that the feedforward MLP sublayer performs as key-value memories which is the key component for the storing and recall of factual associations (Geva et al. 2021, 2022, 2023). The sight into the interpretation of auto-regressive transformer LMs makes renewing their knowledge by directly modifying the MLP weights possible, inspiring knowledge editing via the locate-then-edit paradigm that modifies the located weights (Yao et al. 2023; Meng et al. 2022a,b; Li et al. 2024a). This sort of methods provide convenience for altering the behavior of models permanently on a small amount of facts while ensuring least side-effects, especially meaningful in the era of large language models.

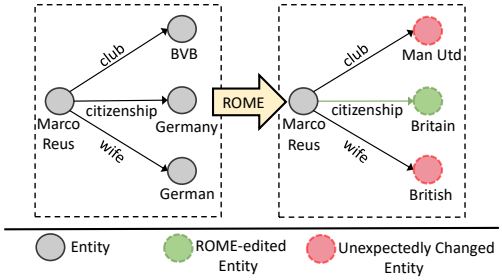


Figure 1: The over-generalizing problem. The circle in green denotes the correctly edited target entity and circles in red denote that the entities unrelated to target editing are also changed unexpectedly.

However, existing locate-then-edit methods suffer from various deficiencies (Li et al. 2024b; Hoelscher-Obermaier et al. 2023). We note an over-generalizing problem that is serious in practical applications where unrelated relationships of the target editing subject experience unexpected alterations during the editing of a certain factual association. For example, the wife and other relationships of *Marco Reus* predicted by the models will be changed to *Britain*-related attributes while ROME (Meng et al. 2022a) edits the citizenship of *Marco Reus* to *Britain*, as illustrated in Figure 1. This makes the contents generated by the edited models untrustworthy.

We conjecture that these locate-then-edit methods suffer from over-generalizing since they only focus on subjects and fail to take relations of factual associations into consideration during editing. Thus we firstly investigate what happens on relation tokens (e.g. "*was born in*") in knowledge recall during inference to understand why previous works fail to take relations into account. The interpretation of knowledge recall involves **which positions** of tokens and **which layers** of auto-

*Zhengxiao Liu and Zheng Lin are corresponding authors.

regressive transformer LMs primarily contribute to the prediction, and **what interpretable information** is encoded at the corresponding points. Through causal tracing (Meng et al. 2022a) for relations, we discover that the most contributing MLP and multi-head self attention MHSA sublayers for the propagation of relation representations appear at the last relation token. Furthermore, we analyze the trend of attributes rate and the target object ranking flow via the vocabulary lens (Geva et al. 2021) of hidden representations at the identified last relation token across layers. Through the analysis results, we conclude with the relation-focused interpretation of knowledge recall that **relation-related attributes (i.e. relational knowledge)** are aggregated **from the first layer till middle-late layers at the last relation token** and that the target object token is extracted from the aggregated relational knowledge. We also validate the importance of MLP over MHSA for the aggregation of relational knowledge by the decline of attributes rate after blocking the MLP and MHSA sublayers respectively during inference. According to the investigation results, we notice that the inference of relations takes place at the last relation token and practically completes in middle-late layers. However, previous works achieve editing via modification of MLP in the middle-early layer at the last subject token, earlier than the inference of relations is completed and also unable to attend to the last relation token behind due to the nature of auto-regressive transformer LMs. As a result, previous locate-then-edit methods fail to take relational knowledge into account and tend to modify no matter what relationships of the target editing subject, leading to the over-generalizing problem.

In order to take relations into consideration, we propose to edit under the guidance of the novel relation-focused interpretation that modifies the MLP in the end of aggregation of relational knowledge (i.e. in the middle-late layer at the last relation token). Although simply editing at this point can attend to the subject, it loses the specificity on predictions of prompts with the same relation but different subjects with target editing (i.e. neighborhood subject prompts). Therefore, to make the hidden representations of neighborhood subject prompts more distinguishable at this point, we add an optimization target to the deduction of the weight modification to enhance the difference between such neighborhood prompts, constraining the editing to the certain subject. To sum up, we propose the **R**elation-focused **E**diting for auto-regressive **T**ransformer LMs with **S**ubject constraints (**RETS**) method to solve the problem of over-generalizing in single knowledge editing initially and verify the reliability of the novel relation-focused interpretation.

For evaluation, we supplement the COUNTERFACT (Meng et al. 2022a) dataset with a new criterion Relation Specificity (i.e. R-Specificity) that measures the influence on unrelated facts of the edited subject. Experimental results on the supplemented dataset show that our editing method avoids over-generalizing by outperforming the state-of-the-art locate-then-edit methods over 30% on Relation Specificity, while remaining competitive with the baselines on the previous criteria. Our strategy of single knowledge editing exhibits the most balanced performance overall and also validates the relation-focused interpretation on the recall of factual associations in auto-regressive transformer LMs, providing a novel perspective for future research on knowledge editing and the recall mechanism.

Related Work

Interpretability of Transformer Language Models. We group the works that focus on the storage and recall mechanism of factual associations in GPT-like models (Zhao et al. 2024; Luo and Specia 2024; Kroeger et al. 2024) based on concerning *where* factual knowledge is stored and *how* the knowledge is retrieved during inference.

The works concerning the storage of factual associations localize the knowledge captured by different transformer components(Vaswani et al. 2017; Kobayashi et al. 2020; Geva et al. 2022, 2021), suggesting that MLP sublayers, also known as the Feed-Forward Networks, act as key-value memories that store the factual associations (Geva et al. 2021). They further point out that each key-value pair of the MLP works as a sub-update that updates the token representation additively (Geva et al. 2022). Meanwhile, the multi-head self-attention MHSA layer is commonly known for its importance in linguistic capabilities (Abnar and Zuidema 2020; Katz and Belinkov 2023; Kobayashi et al. 2024). These works provide a prerequisite for our preference to focus on MLP sublayers in knowledge recall.

The other works trace the information flow for the recall of factual associations during inference (Meng et al. 2022a; Geva et al. 2023; Hernandez et al. 2023). One of them reveals the distinct set of middle-early MLP layers that significantly contribute to the factual predictions during processing the last-subject token via causal mediation analysis (Meng et al. 2022a). Another work subsequently unveil that the representation at the last-subject position is enriched with subject-related attributes (i.e. subject knowledge) through middle-early MLP weights, but it ignores the existence of relational knowledge in knowledge recall (Geva et al. 2023). Although some researchers (Hernandez et al. 2023) notice the role of relation, they explain the computation of a subset of relations as a well-approximated single-linear transformation on the subject representation, still limited to the subject-focused perspective that predicted tokens are extracted from subject knowledge and relations only function to map the subject knowledge to prediction.

As far as we know, none of the existing works about the interpretation of knowledge recall pays attention to the human-interpretable information of the relation representation, ignoring the relational knowledge. We are the first to explore the factual information recalled by the relation during inference.

Knowledge Editing. Knowledge editing methods intend to alter the behavior of language models within the domain related to the edited fact, avoiding side-effects on unrelated facts (Yao et al. 2023; Dai et al. 2022; De Cao, Aziz, and Titov 2021; Dong et al. 2022; Mitchell et al. 2022; Hase et al. 2023). A line of locate-then-edit methods are proposed with the support of the recall mechanisms mentioned above, localizing a decisive MLP weight in middle-early layers at the last-subject position and directly modify it through rank-one model editing ROME (Meng et al. 2022a) for each single

factual association. MEMIT (Meng et al. 2022b) improve ROME to be applicable on numerous edits simultaneously by spreading the update evenly over several middle MLP sublayers while processing the subject representation. PMET (Li et al. 2024a) further obtains more precise FFN output at the last-subject position for editing by taking both MHSA and FFN information into consideration during optimization.

However, the state-of-the-art ROME-like methods primarily ignore the relation information while editing on the subject representation, exhibiting the deficiency of over-generalizing. Unlike these methods, we edit the auto-regressive transformer LMs on the relation representation while being able to take both the relation and the subject information into consideration.

Exploring the Role of Relation in Knowledge Recall

We firstly explore what happens on relations in knowledge recall through causal tracing and the analyses on vocabulary lens of hidden representations.

Background and Notation

We give a description on the propagation within auto-regressive transformer LMs during inference.[1] Given an input text, these auto-regressive transformer LMs tokenize the input sequence into $t_1, t_2, ..., t_N$ of length N and embed them as vectors $h_1^0, h_2^0, ..., h_N^0 \in \mathbb{R}^d$ via the embedding matrix $E \in \mathbb{R}^{|\mathcal{V}| \times d}$ where the vocabulary size is $|\mathcal{V}|$. The models output the probability distribution of the next token $t_{N+1} \in \mathbb{R}^{|\mathcal{V}|}$ through transformer decoders of L layers as follows:

$$P(t_{N+1}|t_1, t_2, ..., t_N) = \text{softmax}(\phi(h_N^{L-1} + a_N^L + m_N^L)) \tag{1}$$

where h_N^{L-1} is the residual hidden representation at N-th token from the layer ahead L-th layer, and a_N^L and m_N^L represent the outputs from L-th MHSA and MLP sublayers respectively. ϕ is the prediction head, mostly the multiplication as $\phi(x) = Ex$ or a trained linear layer. Generally, the hidden representation h_i^l, MLP output m_i^l and MHSA output a_i^l of layer $l \in 1, 2, ..., L$ at token t_i are calculated as follows:

$$h_i^l = h_i^{l-1} + a_i^l + m_i^l \tag{2}$$

$$a_i^l = (\sum_{j=1}^{N} \alpha_{i,j}^l \mathbf{v}^l(h_j^{l-1}))W_O^l \tag{3}$$

$$m_i^l = W_D^l \sigma(W_U^l I_i^l), \mathbf{v}(h_j^{l-1}) = h_j^{l-1} W_V^l \tag{4}$$

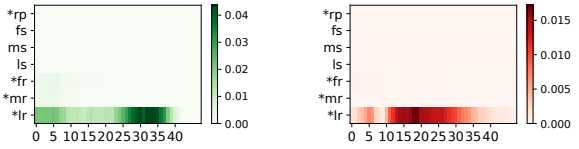
where $W_U^l \in \mathbb{R}^{d' \times d}$ and $W_D^l \in \mathbb{R}^{d \times d'}$ are the up-projection and down-projection weights of the MLP with the inner dimension of d'. σ is the non-linear activation function. $I_i^l \in \mathbb{R}^d$ is the input vector of the MLP sublayer which is often assigned to $(h_i^{l-1} + a_i^l)$ for most auto-regressive transformer LMs but is assigned to h_i^{l-1} for models with the parallel structure of MLP and MHSA. For the MHSA sublayer, $W_O^l \in \mathbb{R}^{d \times d}$ is the input weight matrix and the attention weight $\alpha_{i,j}^l$ is given by:

$$\alpha_{i,j}^l = \text{softmax}(\frac{\mathbf{q}(h_j^{l-1})\mathbf{k}(h_j^{l-1})^T}{\sqrt{d}} + M_{ji}^l) \tag{5}$$

$$\mathbf{q}(h_j^{l-1}) = h_j^{l-1} W_Q^l,\ \mathbf{k}(h_j^{l-1}) = h_j^{l-1} W_K^l \tag{6}$$

where $W_Q^l, W_K^l, W_V^l \in \mathbb{R}^{d \times d}$ are three projection matrices. M_{ji}^l is the attention mask from j-th to i-th hidden representation in auto-regressive models.

[1]The detailed description of the multi-head and nonessential layernorms and bias terms are omitted for simplicity.



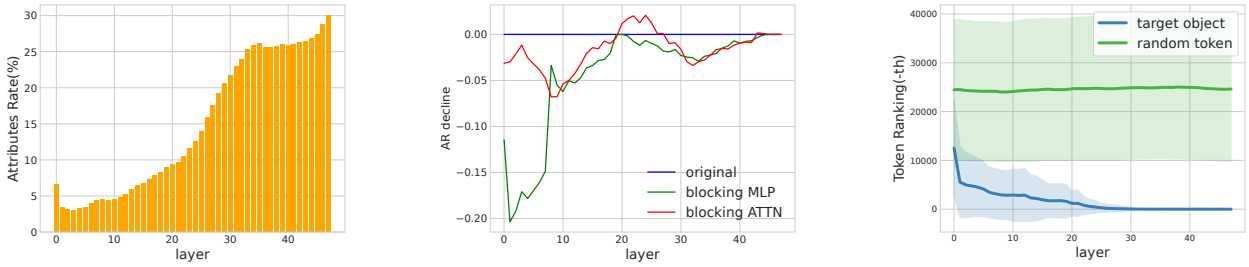
(a) AIER heatmap of MLP (b) AIER heatmap of MHSA

Figure 2: Average Indirect Effect of Relation results for MLP and MHSA sublayers over 1000 facts on GPT2-XL. X-axis shows the layers. "rp" stands for the relation prefix in front of the subject (e.g. "The mother tongue language of" in the input prompt "The mother tongue language of Isabelle Breitman is"). "fs", "ms" and "ls" stand for the first-subject token, middle-subject tokens and last-subject token. "fr", "mr" and "lr" stand for the first-relation token, middle-relation tokens and last-relation token. "*" marks the intervened tokens in the corrupted run.

Identifying Pivotal Positions of Relation

We start by identifying which positions of relation tokens primarily contribute to knowledge recall. Here we display the results of GPT2-XL (Radford et al. 2019) with 48 layers (1.5B parameters). The results of GPT-J (Wang 2021) with 28 layers (6B parameters) and Llama-2 (Touvron et al. 2023) with 32 layers (7B parameters) are displayed in Appendix D, which both show similar trends with that of GPT2-XL.

Method. We utilize causal tracing (Meng et al. 2022a) to measure the importance of each inner activation for the relation tokens through three runs: a *clean run*, a *corrupted* run and a *corrupted-with-restoration* run. In the clean run, a factual association prompt <s, r> is given to the model and the object o is obtained from the output. All the clean internal activations (e.g. m_i^l at token position i in layer l) are cached during this run. Then, in the corrupted run, the embeddings of the relation r is devastated by adding Gaussian noise $\mathcal{N}(0, \gamma)$ to them as r' and the intervened input is sent to the model to obtain the probability for the original object o as $\mathbb{P}(o|<s, r'>)$. At last, in the corrupted-with-restoration run, the corrupted input <s, r'> is still sent to the model but the cached clean hidden states are restored sequentially during inference, resulting in the probability $\mathbb{P}(o|<s, r'>, x_i^l)$ for the output of resuming the activation x_i^l. The difference between our relation-focused causal tracing and previous subject-focused causal tracing lies in the corrupted run, where we add Gaussian noise to the relation tokens r as r' instead of the subject tokens. Thus,



(a) Average attributes rate (b) Attributes rate decline (c) Average token ranking

Figure 3: The factual information detection on the vocabulary lens of the last-relation representation for GPT2-XL over 1000 prompts. (a) The average attributes rates as shown in yellow bars. (b) The average attributes rate decline at 48-th layer while blocking the MLP or MHSA sublayer respectively. (c) The average rankings of the target objects and random tokens.

Position	Average AIER(%)		
	GPT2-XL	GPT-J	Llama-2
rp	0.03	0.03	0.01
fs	0.04	0.04	0.01
ms	0.04	0.04	0.02
ls	0.04	0.04	0.01
fr	0.06	0.06	0.01
mr	0.06	0.06	0.02
lr	**2.23**	**3.15**	**0.06**

Table 1: Average AIER for different positions of tokens. The abbreviations here have the same meanings as in Figure 2.

the contribution of each activation, namely Indirect Effect of Relation (**IER**), is calculated as follows:

$$\textbf{IER} = \mathbb{P}(o|<s, r'>, m_i^l) - \mathbb{P}(o|<s, r'>) \tag{7}$$

Results. Figure 2 shows the average indirect effect of relation (AIER) heatmaps on MLP and MHSA sublayers for GPT2-XL. We note that for both MLP and MHSA, the most significant output representations are detected during inference at the last-relation token (also the last input token). Surprisingly, the AIER of relation prefixes ("rp" in Figure 2) contributes little to the prediction. All that matters for the entire relation information is the hidden representation at the last-relation token. Appendix A shows the AIER heatmaps on some specific cases. For quantitative comparisons, the average AIER across layers for tokens of certain positions is shown in Table 1. The average AIER at the last-relation position is far beyond that at any other position. This indicates that the MLP and MHSA sublayers are most active at the last-relation token which is the decisive position that process the potential factual information of the relation representation. Moreover, we can tell from Figure 2(a) that active MLP sublayers amass from the early layers to the middle-late site of the model, indicating that the update process for the relation representation is of long duration and finishes at the middle-late layers. Therefore, we conclude that the inference of relations primarily takes place at the last relation token from early layers to middle-late layers.

Analyzing the Lens of Representations for Decisive Relation Positions

After identifying the decisive position that dominate the inference of relation tokens, we explore what interpretable information is encoded in the hidden representations at last-relation position through the vocabulary lens (Geva et al. 2021, 2023; Luo and Specia 2024) on them.

Method. For h_i^l which indicates the hidden representation of i-th token at layer l, we map it to the distribution $\mathbf{p}_i^l$ over the vocabulary with the prediction head $\phi(h_i^l)$ which is the projection for prediction at the output layer. Thus, the hidden representation h_i^l can be transformed into the ranking of tokens r_i^l over the vocabulary as follows:

$$\mathbf{p}_i^l = \text{softmax}(\phi(h_i^l)) \tag{8}$$

$$r_i^l = [v_1, v_2, ..., v_{|\mathcal{V}|}] \text{ where } \forall j > k,\ \mathbf{p}_i^l(v_j) > \mathbf{p}_i^l(v_k) \tag{9}$$

Here, v_j stands for the token that ranked j-th in the whole vocabulary $\mathcal{V}$ according to $\mathbf{p}_i^l$. With the rankings, we analyze the hidden representations at last-relation position N according to (i) the ranking of the predicted object o given the input factual association $<s, r>$ across layers (ii) the attributes rate metric (Geva et al. 2023) for the relation r across layers, which measures the semantic relatedness between r and the top ranked tokens $\mathbb{A}_N^l$ from the lens of h_N^l at the last-relation token t_N. For automatic and convenient measurement, we collect a set $\mathbb{A}_{relation}$ containing 200 attributes for each relation r via Wikidata Query Service[2] (see Appendix B for samples of the collection) and the attributes rate is computed as follows:

$$\mathbf{AR}^l = \frac{|\mathbb{A}_N^l \cap \mathbb{A}_{relation}(r)|}{|\mathbb{A}_N^l|} \tag{10}$$

where $\mathbf{AR}^l$ is the attributes rate of the relation representation at layer l and $\mathbb{A}_{relation}(r)$ is the set of attributes related to r of the input factual prompt. In practice, we select top $k = 50$ tokens in each layer for $\mathbb{A}_N^l$ (see Appendix C for example).

[2]https://query.wikidata.org/

Model	Objects Included(%)	ρ
GPT2-XL	68	0.97
GPT-J	83	0.73

Table 2: The percentage of facts where the objects are included in relation-related attributes and the Spearman rank coefficient $\rho \in [-1, 1]$ between the average negative rankings of the objects and the average attributes rate.

Results. Here we display the analysis results of GPT2-XL while the similar results of GPT-J can be found in Appendix E. Figure 3(a) presents the average attributes rates of the representation at last-relation position h_N^l across layers. It shows that the average attributes rate has been rising significantly from layer 0 (the first layer) till 36-th layer and become stable afterwards. This trend indicates that the representation at last-relation position accumulates relation-related attributes from the early layers to the middle-late layers of the models, which is in accordance with the occurrence of the active MLP sublayers in Figure 3(a).

To further explore the importance of MLP and MHSA for the accumulation of relational knowledge respectively, we observe the average drops of attributes rate at 48-th layer while canceling the updates from MLP sublayers or MHSA sublayers at the last token respectively, results shown in Figure 3(b). It shows that blocking MLP leads to a much more significant drop in attributes rate than blocking MHSA across layers at the last token, indicating that MLP plays a much more important role in the enrichment of relational knowledge. Figure 3(c) plots the average rankings of the target objects and random tokens in the vocabulary distributions induced at the last-relation position. We can tell from the line charts that the average rankings of the target objects has been rising from early layers to middle-late layers, while that of random tokens remains low in all layers in comparison. This indicates the target objects are promoted to the final prediction gradually since the first layer of the models. Table 2 shows the proportion of the 1000 facts where the correctly predicted objects are included in corresponding $\mathbb{A}_{relation}(r)$ and the Spearman rank correlation coefficient between the average negative rankings of the objects and the average attributes rate of the representations of the last-relation position across layers. For GPT2-XL (GPT-J), 68% (83%) of correctly predicted objects are included in the corresponding $\mathbb{A}_{relation}(r)$ and the Spearman rank coefficient is 0.97 (0.73), a strongly positive correlation between the extraction of the target objects and the accumulation of relation-related attributes. Thus, we conclude with the relation-focused interpretation that target objects are extracted from the relation-related attributes which are enriched at the last-relation token from early layers till middle-late layers and the MLP sublayers are essential in the update of relation representations. Under the guidance of this interpretation, we achieve editing by modifying the MLP sublayer in end of aggregation of relational knowledge (i.e. in the middle-late layer) with the relation representation (i.e. at the last-relation token) while taking subjects into account.

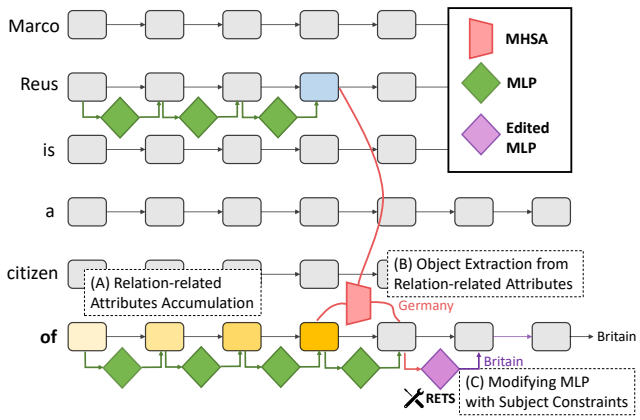


Figure 4: Our RETS method based on the relation-focused recall of factual associations. We reveal that the last-relation representation encodes relation-related attributes (A) which are accumulated until middle-late layers and (B) the predicted object is extracted from these attributes. Based on this relation-focused interpretation, we propose the RETS knowledge editing method that (C) modifies the middle-late MLP sublayer with the constraints of the subject.

Knowledge Editing from the Relation-focused Perspective

To further substantiate the importance of relational knowledge during inference, we apply the novel interpretation on knowledge editing to solve the over-generalizing problem.

Method: RETS

We propose the Relation-focused Editing for auto-regressive Transformer models with Subject constraints (RETS) method that modifies the **middle-late** MLP sublayer with the hidden representation at the **last-relation position** while concerning the subject information, as illustrated in Figure 4. The representation of the last-relation position is selected for its abundant factual information and the ability to attend to the subject tokens ahead. we choose the middle-late MLP sublayer for modification after accomplishing the attributes accumulation, constrained by information propagated from the subject tokens.

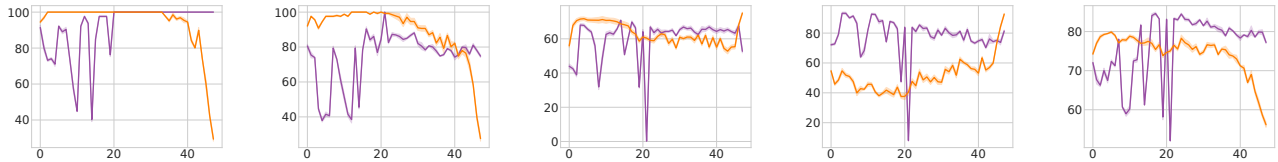
We give the formulization of the RETS method here. Requested to alter a factual association <s, r, o> to <s, r, o^*>, we choose to manipulate the forward pass at last-relation position p_r by modifying the down-projection matrix $W_D^{l_e}$ of the MLP to $\tilde{W}_D^{l_e}$ in a middle-late layer l_e which is in the end of the accumulation of relation-related attributes. The editing target is achieved by injecting $(k_*^{p_r}, v_*^{p_r})$ into the associative memory and optimizing the objective function as follows:

$$\tilde{W}^{l_e} k_*^{p_r} = v_*^{p_r} \tag{11}$$

$$\text{minimize } ||\tilde{W}^{l_e} K - V||_F^2 + ||\tilde{W}^{l_e} K_{p_r} - V_{p_r}||_F^2 \tag{12}$$

$$\text{minimize } ||W^{l_e} K - V||_F^2 \tag{13}$$

where $k_*^{p_r}$ is the average input hidden representation of $W_D^{l_e}$ with several prefixed prompts of <s, r> and $v_*^{p_r}$ is the



(a) Efficacy (b) Generalization (c) S-Specificity (d) R-Specificity (e) Score

Figure 5: The performance of RETS (purple lines) editing at the last-relation token on different layers (x-axis) compared with ROME (orange line) editing at the last-subject token for 50 prompts. Std deviation is shown in areas.

output vector obtained by the optimization process (Appendix H). The first part of the objective function ensures the least change on the original key-value store $K = [\, k_1 \mid k_2 \mid k_3 \mid \dots\,]$ and $V = [\, v_1 \mid v_2 \mid v_3 \mid \dots\,]$. W^{l_e} is the original associative memory that memorizes the mapping from K to V by Eqn. 13, solving as $W^{l_e} K K^T = V K^T$. The next part of the objective function is to emphasize the constraint for unrelated subjects. Since the subject representation propagates to the input vector $\sigma(W_U^l I_{p_r}^l)$ of $W_D^{l_e}$ at p_r (referring to Eqn. 3 and 4), we collect a set of input vectors $K_{p_r} = [\, k_1^{p_r} \mid k_2^{p_r} \mid k_3^{p_r} \mid \dots\,]$ of several prompts with different subjects and their corresponding output vectors $V_{p_r} = [\, v_1^{p_r} \mid v_2^{p_r} \mid v_3^{p_r} \mid \dots\,]$ from $m_{p_r}^l$. Thus the facts of unrelated subjects are ensured the least alternation by minimizing $||\tilde{W}^{l_e} K_{p_r} - V_{p_r}||_F^2$, where we also have $W^{l_e} K_{p_r} K_{p_r}^T = V_{p_r} K_{p_r}^T$ for the original associative memory. By optimizing the objective function with all these constraints, we obtain the solution for $\tilde{W}^{l_e}$ as the new weight $W_D^{l_e}$ (see Appendix H for details):

$$\begin{aligned} \tilde{W}^{l_e} &= W^{l_e} + \Lambda^{p_r}((C + K_{p_r} K_{p_r}^T)^{-1} k_*^{p_s})^T \\ &\approx W_D^{l_e} + \Lambda^{p_r}((C + K_{p_r} K_{p_r}^T)^{-1} k_*^{p_s})^T \end{aligned} \tag{14}$$

where $C = KK^T$ is the constant estimated with the uncentered covariance of k on a slice of Wikipedia corpus and Λ^{p_r} is solved as $(v_*^{p_r} - W^{l_e} k_*^{p_r})/((C + K_{p_r} K_{p_r}^T)^{-1} k_*^{p_r})^T k_*^{p_r}$ which is proportional to the gap between the initial output vector $W^{l_e} k_*^{p_r}$ and the target output vector $v_*^{p_r}$. The linear memory W^{l_e} is approximated by the model weight $W_D^{l_e}$.

Experiments

Baselines and DataSets. Our editing experiments are mainly conducted on GPT2-XL and GPT-J for each single factual association. We also evaluate the basic editing performance on Llama-2 (7B). To compare with, we choose the methods for editing each single factual association as our baselines, including Constrained Fine-Tuning (FT+L) (Zhu et al. 2020), the meta-learning method (MEND) (Mitchell et al. 2022) which learns the update of model weights with additional networks, ROME (Meng et al. 2022a) on the subject representation in the early site and the improved precise model editing (PMET) (Li et al. 2024a) which optimizes the parallel MHSA and MLP representations simultaneously.

For evaluation, we conduct experiments on $10,000$ ($2,000$) samples of COUNTERFACT (Meng et al. 2022a) dataset for GPT2-XL (GPT-J), $1,000$ samples of COUNTERFACT for Llama-2 and $10,000$ samples of Zero-shot Relation Extraction (zsRE) (Levy et al. 2017; Meng et al. 2022a) for GPT2-XL. We edit on the 36-th layer of GPT2-XL, 18-th layer of GPT-J and 23-th layer of Llama-2. For COUNTERFACT, we supplement this dataset with unrelated facts of the same subject for each target editing and the corresponding metric Relation Specificity (**R-Specificity**) to measure the over-generalizing problem. To be specific, given the edited fact <s_1, r_1, o_1^*> and the unrelated fact with the same subject <s_1, r_2, o_2>, we test $\mathbb{P}[o_2] > \mathbb{P}[o_1^*]$ as the R-Specificity Score with the input prompt of <s_1, r_2>, computed similarly to the existing metrics. We also test the original metrics including the basic **Efficacy** accuracy Scores to measure the success rate of target editing, the **Generalization** accuracy score for generalization on the paraphrased statements, the renamed Subject Specificity **S-Specificity** accuracy score for specificity within neighborhood subjects. The advanced **Fluency** and **Consistency** scores measure the quality of generated full texts. Higher scores indicate better performance for all metrics. For zsRE, the metics and evaluation results are displayed in Appendix G. Details for the construction of R-Specificity samples and detailed settings are presented in Appendix F.

Evaluation Results. Table 3 shows the evaluation results on COUNTERFACT. We observe that the existing mainstream editing methods exhibit at least one deficiency. Even though existing ROME-like methods (ROME and PMET) perform well on most criteria, they experience obvious failure on the Relation Specificity. Our RETS method outperforms the ROME-like methods over 30% on R-Specificity for both GPT2-XL and GPT-J while remaining competitive on other criteria, indicating that the relation-focused editing solves the over-generalizing problem for ROME initially. The evaluation result on recent Llama-2 also shows the same trend. The results on zsRE in Appendix G also demonstrate the competitiveness of RETS. An anecdotal case of RETS behaving correctly while ROME behaving erroneously on GPT2-XL is shown in Figure 4.

The ablation of the subject constraints for editing COUNTERFACT leads to the 35% decrease on Entity Specificity for GPT2-XL, which indicates the effectiveness of the subject constraints for the relation-focused editing. Despite of the results on Generalization and Entity Specificity where RETS loses about 20% and 10% respectively compared with the subject-centered editing methods, RETS exhibits the most balanced performance with its simple way of combining the subject information into editing, which shows the potential of editing from relation-focused perspective. The trade-off of performance is decided by the editing position (the last-relation token or the last-subject token) as expected. Editing

Editor	Score	Efficacy	Generalization	S-Specificity	R-Specificity	Fluency	Consistency
GPT2-XL	55.9	21.0	24.1	78.6	100.0	626.8	34.7
FT-L	73.1	99.2	47.8	70.6	74.9	623.3	37.6
MEND	63.2	62.3	53.1	51.7	85.6	603.7	32.7
ROME	78.4	100.0	96.4	76.0	41.1	622.6	42.0
PMET*	79.3	99.2	94.3	76.0	47.6	622.7	41.8
RETS	79.7	100.0	71.5	68.6	78.5	577.4	32.6
- w/o SC	71.1	100.0	67.2	35.1	86.9	626.1	34.9
GPT-J	53.2	13.7	15.3	83.7	100.0	621.7	29.7
FT-L	79.3	99.6	47.4	80.1	89.1	622.5	35.3
MEND	75.3	96.8	51.2	53.8	99.2	620.4	32.2
ROME	81.7	99.9	99.0	79.4	48.5	620.5	42.7
PMET*	83.5	99.9	98.7	79.6	55.6	620.9	43.0
RETS	80.7	100.0	74.2	65.5	83.3	542.4	29.2
- w/o SC	74.1	100.0	82.0	23.7	90.7	618.1	34.9
Llama-2	52.8	13.8	16.1	81.2	100.0	-	-
FT-L	55.6	24.2	17.0	81.6	99.7	-	-
ROME	81.1	99.9	93.4	77.4	53.6	-	-
RETS	82.1	98.3	74.6	72.3	83.1	-	-

Table 3: The evaluation results on COUNERTFACT for GPT2-XL and GPT-J. The significantly failed values for the editing methods on basic criteria are underlined. "Score" shows the average value on the basic criteria: Efficacy, Generalization, S-Specificity and R-Specificity. "SC" stands for the subject constraints on our relation-focused editing. R-Specificity values for raw models are 100.0% since the criterion is constructed according to the top token predictions of the raw models. *PMET is adjusted to accommodate to edit a single layer.

at the last-relation position ensures minimal impact on unrelated relations (i.e. high R-Specificity) but loses much subject information (i.e. low S-Specificity), while the opposite is also true for editing at the last-subject position. The superiority of the relation-focused approach is that the decline of S-Specificity can be constrained by subject constraints whereas the subject-focused approach can hardly attend to the relations. Detailed discussions can be referred to Appendix J.

Editing Target: Lionel Messi is a native speaker of *Chinese*.
Original Model: [Target Prompt] Lionel Messi is a native speaker of [Prediction] Argentine [R-Specificity Prompt] Lionel Messi plays for the club called [Prediction] FC Barcelona
ROME Edited: [Target Prompt] Lionel Messi is a native speaker of [Prediction] Chinese [R-Specificity Prompt] Lionel Messi plays for the club called [Prediction] Shanghai Shenhua
RETS Edited: [Target Prompt] Lionel Messi is a native speaker of [Prediction] Chinese [R-Specificity Prompt] Lionel Messi plays for the club called [Prediction] FC Barcelona

Table 4: An anecdotal example of the correct behavior for RETS and the incorrect behavior for ROME on GPT2-XL. Predictions in red denote unexpectedly changed answers.

Layer Analysis. We test the effectiveness of RETS while editing on different layers and compare it with the behavior of ROME which edits at the last-subject token . Figure 5 plots the performance on four criteria (a,b,c,d) and the average scores (e) across layers. The performance of RETS vibrates before middle layers and become stable after middle-late layers, validating our interpretation of relation-focused knowledge recall that the object is attracted from relation-related attributes which are accumulated before middle-late layers. RETS editing on middle-late layers shows more balanced performance than editing on any layer at the last-subject token where relation information behind hardly propagates to.

Conclusion

We discover the over-generalizing problem for previous subject-focused knowledge editing methods, and we solve this problem by further exploring the role of relations in knowledge recall. As a result, we unveil the factual information encoded for relations in auto-regressive transformer language models, and we propose the RETS single knowledge editing method based on the relation-focused interpretation. Our experiments demonstrate the effectiveness of RETS on solving the over-generalizing problem and provide the novel relation-focused perspective for future research on both the interpretation and editing of the auto-regressive transformer language models, breaking the domination of the subject-focused perspective.

Ethical Statement

The goal of our work is to investigate and renew the outdated or mistaken knowledge decoded in transformer language models. However, we recognize inherent risks associated with potential malicious applications like injecting harmful information. Therefore, we emphasize the importance that language models be sourced exclusively from reputable and trustworthy providers and carefully use the contents generated by these models.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 62472419, No. 62472420).

References

Abnar, S.; and Zuidema, W. 2020. Quantifying Attention Flow in Transformers. In Jurafsky, D.; Chai, J.; Schluter, N.; and Tetreault, J., eds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4190–4197. Online: Association for Computational Linguistics.

Dai, D.; Dong, L.; Hao, Y.; Sui, Z.; Chang, B.; and Wei, F. 2022. Knowledge Neurons in Pretrained Transformers. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 8493–8502. Dublin, Ireland: Association for Computational Linguistics.

De Cao, N.; Aziz, W.; and Titov, I. 2021. Editing Factual Knowledge in Language Models. In Moens, M.-F.; Huang, X.; Specia, L.; and Yih, S. W.-t., eds., *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6491–6506. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Dong, Q.; Dai, D.; Song, Y.; Xu, J.; Sui, Z.; and Li, L. 2022. Calibrating Factual Knowledge in Pretrained Language Models. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Findings of the Association for Computational Linguistics: EMNLP 2022*, 5937–5947. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Geva, M.; Bastings, J.; Filippova, K.; and Globerson, A. 2023. Dissecting Recall of Factual Associations in Auto-Regressive Language Models. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 12216–12235. Singapore: Association for Computational Linguistics.

Geva, M.; Caciularu, A.; Wang, K.; and Goldberg, Y. 2022. Transformer Feed-Forward Layers Build Predictions by Promoting Concepts in the Vocabulary Space. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 30–45. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Geva, M.; Schuster, R.; Berant, J.; and Levy, O. 2021. Transformer Feed-Forward Layers Are Key-Value Memories. In Moens, M.-F.; Huang, X.; Specia, L.; and Yih, S. W.-t., eds., *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 5484–5495. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Hase, P.; Bansal, M.; Kim, B.; and Ghandeharioun, A. 2023. Does Localization Inform Editing? Surprising Differences in Causality-Based Localization vs. Knowledge Editing in Language Models. arXiv:2301.04213.

Heinzerling, B.; and Inui, K. 2021. Language Models as Knowledge Bases: On Entity Representations, Storage Capacity, and Paraphrased Queries. In Merlo, P.; Tiedemann, J.; and Tsarfaty, R., eds., *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 1772–1791. Online: Association for Computational Linguistics.

Hernandez, E.; Sharma, A.; Haklay, T.; Meng, K.; Wattenberg, M.; Andreas, J.; Belinkov, Y.; and Bau, D. 2023. Linearity of Relation Decoding in Transformer Language Models. *ArXiv*, abs/2308.09124.

Hoelscher-Obermaier, J.; Persson, J.; Kran, E.; Konstas, I.; and Barez, F. 2023. Detecting Edit Failures In Large Language Models: An Improved Specificity Benchmark. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Findings of the Association for Computational Linguistics: ACL 2023*, 11548–11559. Toronto, Canada: Association for Computational Linguistics.

Jiang, Z.; Xu, F. F.; Araki, J.; and Neubig, G. 2020. How Can We Know What Language Models Know? *Transactions of the Association for Computational Linguistics*, 8: 423–438.

Katz, S.; and Belinkov, Y. 2023. VISIT: Visualizing and Interpreting the Semantic Information Flow of Transformers. arXiv:2305.13417.

Kobayashi, G.; Kuribayashi, T.; Yokoi, S.; and Inui, K. 2020. Attention is Not Only a Weight: Analyzing Transformers with Vector Norms. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 7057–7075. Online: Association for Computational Linguistics.

Kobayashi, G.; Kuribayashi, T.; Yokoi, S.; and Inui, K. 2024. Analyzing Feed-Forward Blocks in Transformers through the Lens of Attention Maps. arXiv:2302.00456.

Kroeger, N.; Ley, D.; Krishna, S.; Agarwal, C.; and Lakkaraju, H. 2024. Are Large Language Models Post Hoc Explainers? arXiv:2310.05797.

Lester, B.; Al-Rfou, R.; and Constant, N. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. In Moens, M.-F.; Huang, X.; Specia, L.; and Yih, S. W.-t., eds., *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 3045–3059. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Levy, O.; Seo, M.; Choi, E.; and Zettlemoyer, L. 2017. Zero-Shot Relation Extraction via Reading Comprehension. In Levy, R.; and Specia, L., eds., *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, 333–342. Vancouver, Canada: Association for Computational Linguistics.

Li, X.; Li, S.; Song, S.; Yang, J.; Ma, J.; and Yu, J. 2024a. PMET: Precise Model Editing in a Transformer. arXiv:2308.08742.

Li, Z.; Zhang, N.; Yao, Y.; Wang, M.; Chen, X.; and Chen, H. 2024b. Unveiling the Pitfalls of Knowledge Editing for Large Language Models. arXiv:2310.02129.

Luo, H.; and Specia, L. 2024. From Understanding to Utilization: A Survey on Explainability for Large Language Models. arXiv:2401.12874.

Meng, K.; Bau, D.; Andonian, A.; and Belinkov, Y. 2022a. Locating and Editing Factual Associations in GPT. *Advances in Neural Information Processing Systems*, 35.

Meng, K.; Sharma, A.; Andonian, A.; Belinkov, Y.; and Bau, D. 2022b. Mass-Editing Memory in a Transformer. *ArXiv*, abs/2210.07229.

Mitchell, E.; Lin, C.; Bosselut, A.; Finn, C.; and Manning, C. D. 2022. Fast Model Editing at Scale. arXiv:2110.11309.

Petroni, F.; Lewis, P.; Piktus, A.; Rocktäschel, T.; Wu, Y.; Miller, A. H.; and Riedel, S. 2020. How Context Affects Language Models' Factual Predictions. In *Automated Knowledge Base Construction*.

Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P.; Bakhtin, A.; Wu, Y.; and Miller, A. 2019. Language Models as Knowledge Bases? In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2463–2473. Hong Kong, China: Association for Computational Linguistics.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Roberts, A.; Raffel, C.; and Shazeer, N. 2020. How Much Knowledge Can You Pack Into the Parameters of a Language Model? In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5418–5426. Online: Association for Computational Linguistics.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I.; Korenev, A.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X. E.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288.

Vaswani, A.; Shazeer, N. M.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *Neural Information Processing Systems*.

Wang, B. 2021. Mesh-Transformer-JAX: Model-Parallel Implementation of Transformer Language Model with JAX. https://github.com/kingoflolz/mesh-transformer-jax.

Wang, C.; Liu, P.; and Zhang, Y. 2021. Can Generative Pre-trained Language Models Serve As Knowledge Bases for Closed-book QA? In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 3241–3251. Online: Association for Computational Linguistics.

Yao, Y.; Wang, P.; Tian, B.; Cheng, S.; Li, Z.; Deng, S.; Chen, H.; and Zhang, N. 2023. Editing Large Language Models: Problems, Methods, and Opportunities. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 10222–10240. Singapore: Association for Computational Linguistics.

Zhao, H.; Chen, H.; Yang, F.; Liu, N.; Deng, H.; Cai, H.; Wang, S.; Yin, D.; and Du, M. 2024. Explainability for Large Language Models: A Survey. *ACM Trans. Intell. Syst. Technol.*, 15(2).

Zhu, C.; Rawat, A. S.; Zaheer, M.; Bhojanapalli, S.; Li, D.; Yu, F.; and Kumar, S. 2020. Modifying Memories in Transformer Models. arXiv:2012.00363.