The Fall of *ROME*: Understanding the Collapse of LLMs in Model Editing

Wanli Yang♠♡ **Fei Sun**♠†
Jiajun Tan♠ **Xinyu Ma**♣ **Du Su**♠ **Dawei Yin**♣ **Huawei Shen**♠♡
♠CAS Key Laboratory of AI Safety, Institute of Computing Technology, CAS
♡University of Chinese Academy of Sciences ♣Baidu Inc.
yangwanli24z@ict.ac.cn sunfei@ict.ac.cn

Abstract

Despite significant progress in model editing methods, their application in real-world scenarios remains challenging as they often cause large language models (LLMs) to collapse. Among them, ROME is particularly concerning, as it could disrupt LLMs with only a single edit. In this paper, we study the root causes of such collapse. Through extensive analysis, we identify two primary factors that contribute to the collapse: i) *inconsistent handling of prefixed and unprefixed keys* in the parameter update equation may result in very small denominators, causing excessively large parameter updates; ii) *the subject of collapse cases is usually the first token*, whose unprefixed key distribution significantly differs from the prefixed key distribution in autoregressive transformers, causing the aforementioned issue to materialize. To validate our findings, we propose a simple yet effective approach: uniformly using prefixed keys during editing phase and adding prefixes during testing phase to ensure the consistency between training and testing. The experimental results show that the proposed solution can prevent model collapse while maintaining the effectiveness of the edits[1].

1 Introduction

Recent works (Yang et al., 2024; Gupta et al., 2024b; Gu et al., 2024) have revealed that model editing (Zhang et al., 2024) poses significant risks of compromising the capabilities of large language models (LLMs). Among them, Rank-One Model Editing (ROME) (Meng et al., 2022), a cutting-edge method, has been found to cause model collapse with just a single edit (Yang et al., 2024). In this paper, we aim to study the underlying causes behind this phenomenon.

Intuitively, for a knowledge tuple (subject, relation, object), ROME takes a prompt constructed from the subject and relation as input and models the knowlege in a key-value format. Here, the key is a vector representation of the subject within the prompt, and the value is a vector representation capable of yielding the target object, obtained by transforming the key through a transformation matrix. To insert a new fact about a subject, ROME adjusts the transformation matrix to match the key of the subject with the value of the new fact, as described in Eq. 3.

To uncover the underlying causes of ROME's collapse, we investigate the differences in parameter update process of ROME between *collapse cases* (i.e., samples that induce collapse) and *normal cases* (i.e., samples that do not). The results reveal that the collapse directly stems from the anomalously small denominator within the parameter update equation (Eq. 3). This anomaly originates from the irregular implementation of the keys in the denominator, where one is derived by prepending varying prefixes to the subject to simulate diverse contexts (termed *prefixed key*), while the other is obtained directly from the original subject without any prefix (termed *unprefixed key*). This issue has also been independently identified by Gupta et al. (2024a) concurrently. However, it is still unclear why the irregular implementation only fails in collapse cases.

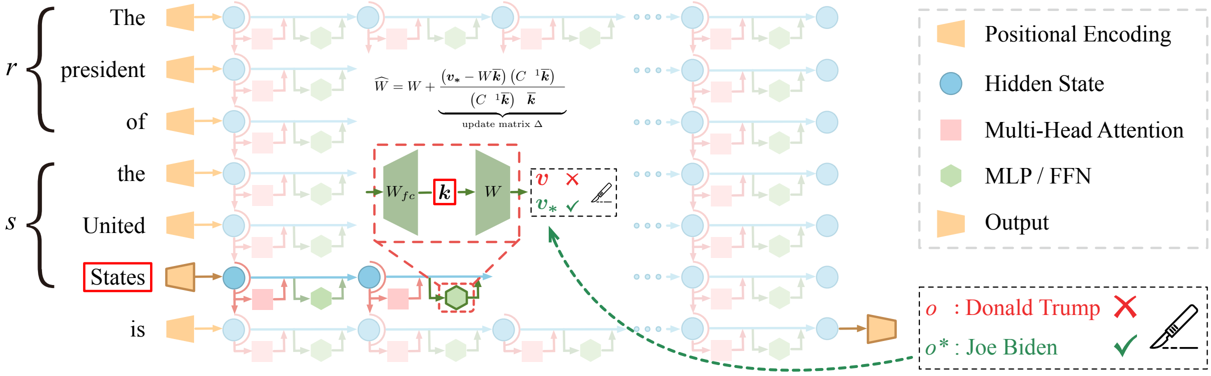
To answer this question, we examine the distribution of elements in the denominator. It reveals that, in collapse cases, the distribution of the unprefixed keys exhibits significant difference from the prefixed keys. This leads to an exceptionally small denominator in the update equation, which in turn causes the model to collapse.

To elucidate the anomalous behavior observed in the collapse cases, we conduct an analysis starting from their characteristics. The collapse cases of both GPT-2-XL (Radford et al., 2019) and GPT-J

[†] Corresponding author.

[1] Code and data are available at: https://github.com/WanliYoung/Collapse-in-Model-Editing.

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 4079–4087
November 12-16, 2024 ©2024 Association for Computational Linguistics



r
The
president
of
s
the
United
States
is
$\widehat{W} = W + \underbrace{\frac{(\boldsymbol{v}_* - W\overline{\boldsymbol{k}})\,(C \quad ^1\overline{\boldsymbol{k}})}{(C \quad ^1\overline{\boldsymbol{k}}) \quad \overline{\boldsymbol{k}}}}_{\text{update matrix } \Delta}$
W_{fc}
$\boldsymbol{k}$
W
$\boldsymbol{v}$ ✗
$\boldsymbol{v}_*$ ✓
Positional Encoding
Hidden State
Multi-Head Attention
MLP / FFN
Output
o : Donald Trump ✗
o^* : Joe Biden ✓

Figure 1: To update "the president of the United States" from "Donald Trump" to "Joe Biden", ROME locates the knowledge into the MLP module within a specific transformer block using the Causal Tracing mechanism. It then adjusts the second layer of MLP (i.e., weight matrix W) to change the value $\boldsymbol{v}$ for the key $\boldsymbol{k}$ that represents the subject "the United States" to a new value $\boldsymbol{v}_*$, thereby inducing the LLMs to predict the target object "Joe Biden".

(Wang and Komatsuzaki, 2021) exhibit a consistent pattern: *the subjects in nearly all of these instances correspond to the first tokens within their respective prompts*. Furthermore, we discover that *the representation distribution of the first tokens markedly diverges from that of the subsequent tokens in these autoregressive models*. These two factors, working in concert, lead to the anomalous distribution of unprefixed keys in collapse cases.

To validate our findings, we propose unifying all keys as prefixed during editing to prevent model collapse. To ensure consistency with the editing process, when using the edited model, we prefix a random text for instances where subjects are in the first token. Experiments validate that our proposed method effectively prevents model collapse while ensuring the success of edits.

Our main contributions are as follows:

- Comprehensive analysis that identifies two factors behind ROME's collapse: i) inconsistent implementation of key vectors; ii) anomalous distribution of first token representations.
- A straightforward solution to prevent collapse while maintaining editing efficacy.

2 Background

ROME (Meng et al., 2022) hypothesizes that the MLP modules in the Transformer architecture (Vaswani et al., 2017) can be modeled as a linear key-value associative memory. Under the hypothesis in ROME, a knowledge triplet (s, r, o) corresponds to a key-value pair $(\boldsymbol{k}, \boldsymbol{v})$, where $\boldsymbol{k}$ represents the subject s, and $\boldsymbol{v}$ encodes the property (r, o) for s. The entire knowledge within a model can thus be represented as a set of key vectors $K = [\boldsymbol{k}_1, \ldots, \boldsymbol{k}_n]$ and value vectors $V = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n]$. A linear operation W matches keys to values by solving $WK \approx V$.

In practice, for an input prompt $\mathrm{p}(s, r)$, the recall of the target object o mainly occurs within a two-layer MLP in a specific transformer block identified by the Causal Tracing mechanism (Meng et al., 2022). Specifically, output of the first layer for the subject s forms a key $\boldsymbol{k}$, and the second layer (parameterized with W) retrieves an associated value $\boldsymbol{v}$ based on this key $\boldsymbol{k}$, ultimately inducing the LLMs to predict the target object o.

In this context, to replace the current knowledge (s, r, o) with a new knowledge tuple $t^* = (s, r, o^*)$, we need to find the corresponding key $\boldsymbol{k}$ and the new value $\boldsymbol{v}_*$. To simulate various contexts for generalization, ROME assigns $\boldsymbol{k}$ as an average vector $\overline{\boldsymbol{k}}$ derived from subject s with a small set of N randomly sampled prefixes:

$$\overline{\boldsymbol{k}} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{K}\left(x_i \oplus s\right) \qquad (1)$$

where $\mathcal{K}$ is the output of the first MLP layer in transformer block, x_i is the prefixes, and $\oplus$ is string concatenation operator.

To illustrate the selection of $\boldsymbol{v}_*$, we take the subject s= "*United States*" and relation r= "*president of*" as an example. A specifically designed loss function is utilized to optimize $\boldsymbol{v}_*$ so that it can produce o^* = "*Joe Biden*" when given the prompt $\mathrm{p}(s, r)$ = "*The president of the United States is*".

With the computed $(\overline{\boldsymbol{k}}, \boldsymbol{v}_*)$, ROME finds optimal $\widehat{W}$ by solving the following problem:

$$\underset{\widehat{W}}{\arg\min} \|\widehat{W}K - V\| \quad \text{subject to } \widehat{W}\overline{\boldsymbol{k}} = \boldsymbol{v}_* \qquad (2)$$

Component	Cases	GPT-2-XL	GPT-J	Llama2-7b
numerator:	collapse	168.55	140.27	4.57
$(\boldsymbol{v}_* - W\overline{\boldsymbol{k}})\left(C^{-1}\overline{\boldsymbol{k}}\right)^\top$	normal	79.91	88.69	16.52
denominator:	collapse	0.04	0.04	0.01
$\left(C^{-1}\overline{\boldsymbol{k}}\right)^\top \overline{\boldsymbol{k}}$	normal	9.60	12.78	2.63

Table 1: Average norm of the numerator and average absolute value of the denominator in ROME's update matrix Δ across various LLMs for different sets of cases.

It has the following closed-form solution:

$$\widehat{W} = W + \underbrace{\frac{(\boldsymbol{v}_* - W\overline{\boldsymbol{k}})\left(C^{-1}\overline{\boldsymbol{k}}\right)^\top}{\left(C^{-1}\overline{\boldsymbol{k}}\right)^\top \overline{\boldsymbol{k}}}}_{\text{update matrix } \Delta} \tag{3}$$

where W denotes the weight matrix of the second layer in the MLP before editing, $\widehat{W}$ denotes the weight matrix after editing, and $C{=}KK^\top$ is a pre-cached constant.

The complete editing process of ROME is illustrated in Figure 1. Interested readers are directed to Meng et al. (2022) for a detailed introduction.

3 Why Does ROME Cause Collapse?

Previous studies (Yang et al., 2024; Gupta et al., 2024b) have revealed that a single edit of ROME can induce LLMs to collapse. To further analyze the cause, we investigate the differences in parameter updates between samples that induce collapse and those do not. For this purpose, we introduce two distinct subsets: i) *collapse cases*, using the *HardCF* set built by Yang et al. (2024), which includes collapse cases on GPT-2-XL, GPT-J, and Llama2-7b from the COUNTERFACT dataset (Meng et al., 2022); and ii) *normal cases*, comprising 1000 random samples from the remaining part of COUNTERFACT.

3.1 Inconsistent Keys in Editing

Existing work (Yang et al., 2024) has found that collapse is caused by the values of update matrix Δ in Eq. 3 being excessively large. For fine-grained analysis, we split Δ into *numerator* (a matrix) and *denominator* (a scalar), and then apply single edits to analyze the intermediate values for parameter updating in different cases. As illustrated in Table 1, the denominators of collapse cases are two orders of magnitude smaller than those of normal cases, while the numerators do not show significant differences. This disparity directly results in the exceptionally large Δ of collapse cases.

Method	Cases	GPT-2-XL	GPT-J	Llama2-7b
Original		68.77	49.04	33.18
ROME	collapse	26,084.66	25,909.24	10,574.76
	normal	74.32	50.77	36.68
C-ROME	collapse	70.71	51.77	33.20
	normal	70.28	50.57	33.55

Table 2: The maximum ME-PPL$_{50}$ perplexity of models edited by different implementations of ROME for their collapse cases and normal cases, with their original models' perplexity for comparison.

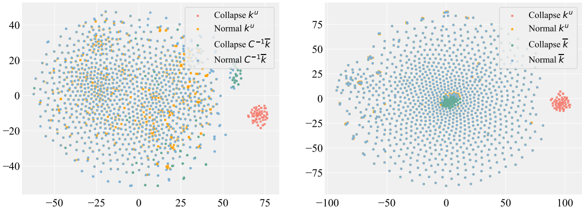
These results guide our focus to the key $\overline{\boldsymbol{k}}$ in the denominator $(C^{-1}\overline{\boldsymbol{k}})^\top\overline{\boldsymbol{k}}$, given that the matrix C is a constant for both collapse cases and normal cases. We revisited the official implementation of ROME and identified that **different variants of $\overline{\boldsymbol{k}}$ are used**. Specifically, only $\overline{\boldsymbol{k}}$ within $(C^{-1}\overline{\boldsymbol{k}})^\top$ is the prefixed key as in Eq. 1. In contrast, **$\overline{\boldsymbol{k}}$ in other positions is unprefixed**, utilizing a representation over the subject s without any prefix, denoted as $\boldsymbol{k}^u = \mathcal{K}(s)$. However, ideally, all $\overline{\boldsymbol{k}}$ in Eq. 3 should be the same, i.e., the average representation derived from a set of prefixed subjects as in Eq. 1.

To verify if this inconsistency of keys is responsible for the collapse, we substitute all $\boldsymbol{k}^u$ with $\overline{\boldsymbol{k}}$ in the implementation. The aligned implementation is referred to as *Consistent-ROME*, *C-ROME* for short. We evaluate the different implementations on collapse and normal cases using perplexity on the ME-PPL$_{50}$ dataset, whose effectiveness has been validated by Yang et al. (2024). According to Table 2, C-ROME with aligned implementation of $\overline{\boldsymbol{k}}$ does not significantly alter the edited models, avoiding the sharp increase in perplexity seen with ROME. This demonstrates that such inconsistency of $\overline{\boldsymbol{k}}$ in the update matrix Δ is a primary factor behind ROME-induced model collapse.

3.2 Anomalous Key Distribution for Collapse

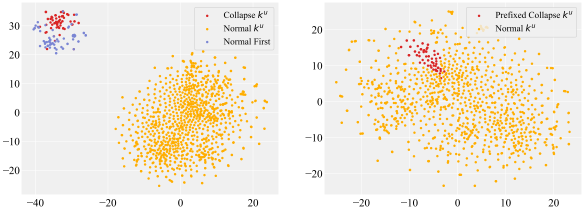
While unifying the keys as $\overline{\boldsymbol{k}}$ can prevent model collapse, it remains unclear *why inconsistent keys only encounter issues in collapse cases*.

To enhance intuitive understanding, we analyze the spatial distribution of $C^{-1}\overline{\boldsymbol{k}}$ and $\boldsymbol{k}^u$ in the denominator for different cases by projecting them into a two-dimensional space using t-SNE (Van der Maaten and Hinton, 2008). Taking the results of GPT-2-XL in Figure 2a as an example, in normal cases, the distribution of $C^{-1}\overline{\boldsymbol{k}}$ and $\boldsymbol{k}^u$ show no significant differences. However, a noticeable di-



(a) (b)

Figure 2: t-SNE visualization of (a) elements in the denominator; (b) different implementation of key vectors.



(a) (b)

Figure 3: t-SNE visualization of representation distributions of (a) the first token in randomly sampled normal prompts; (b) $\boldsymbol{k}^u$ in prefixed collapse prompts.

vergence in the distribution occurs in collapse cases, explaining the exceptionally small denominators.

Considering that C is a constant, the differences between normal and collapse cases should arise from the variations in the prefixed key $\overline{\boldsymbol{k}}$ and the unprefixed key $\boldsymbol{k}^u$. Figure 2b clearly illustrates that the distribution of $\boldsymbol{k}^u$ in collapse cases significantly diverge from those of $\overline{\boldsymbol{k}}$. This confirms that in collapse cases, the significant differences between $\overline{\boldsymbol{k}}$ and $\boldsymbol{k}^u$ result in a particularly small denominator in the update matrix, which in turn leads to the collapse of the edited model. Similar phenomena are also observed in other LLMs, detailed in § A.1.

3.3 Special Role of the First Token

To elucidate the anomalous distribution of $\boldsymbol{k}^u$ in collapse cases, we focus our analysis on their characteristics. A common pattern is observed in the collapse cases for both GPT-2-XL and GPT-J: *in almost all instances, the subjects consist of a single word, which is encoded as a single token and positioned at the beginning of the input prompt* $\mathrm{p}(s, r)$[2]. Therefore, the unprefixed key $\boldsymbol{k}^u$ for a collapse case is the intermediate representation within the MLP layer of the first token in the input. This inspires us to investigate whether the anomalous distribution of $\boldsymbol{k}^u$ in collapse cases can be attributed to their position as the first tokens in the prompts.

To explore this, we first examined the representation distribution of the first tokens in the prompts for normal cases. The results presented in Figure 3a indicate that, within GPT-2-XL, the first tokens of normal cases consistently exhibit an abnormal distribution similar to that of $\boldsymbol{k}^u$ in collapse cases. From an opposing perspective, to verify whether artificially shifting the $\boldsymbol{k}^u$ in collapse cases away from the first position would eliminate the anomaly in distribution, we prefixed the prompts of collapse cases with randomly sampled texts. This adjustment results in their distribution aligning with that of normal cases, as illustrated in Figure 3b. These findings suggest that the anomalous distribution of $\boldsymbol{k}^u$ for collapse cases in ROME is not related to the editing process. Instead, it is due to the unique pattern of their subjects encountering the special distribution of the first token in GPT-2-XL and GPT-J models.

It is important to note that Llama2-7b (Touvron et al., 2023), Mistral-7b (Jiang et al., 2023), and Llama3-8b (Meta, 2024) avoid collapse in such cases due to their tokenizers additionally incorporating a special token, e.g., <s>, at the beginning of the input, which shifts the subject from being the first token. In fact, we found they also succumb to collapse when the special token is removed, with results detailed in Appendix A.2.

Analysis. To elucidate the underlying reasons for the anomalous distribution of the first token in autoregressive language models, we explored two potential factors as follows.

Firstly, we speculate that this phenomenon may arise from the inherent nature of autoregressive models, where the first token cannot interact with any other token except itself. As a counterexample with non-autoregressive architecture, the representation distribution of the first token in T5-3B encoder (Raffel et al., 2020) does not differ from that of subsequent tokens. This may be attributed to the bidirectional attention in the encoder, which enables interactions between the first token and subsequent tokens. A detailed analysis is presented in Appendix A.3.

Secondly, considering the specificity of the first token may originate from its position embedding, we verify it from two aspects. For *collapse cases* where the subjects are the first tokens, setting the position embedding of the first token as that of the second token can not completely eliminate collapse. While for *normal cases* where the subjects are the

[2] The only exception involves few instances with subjects like "Jackson Jackson" in the collapse cases of GPT-J.

Model	GPT-2-XL	GPT-J	Mistral-7b	Llama3-8b
Ori PPL	68.39	50.34	51.75	41.67
Max PPL	68.91	50.59	52.19	43.98

Table 3: The maximum perplexity for various LLMs edited by ROME on the collapse cases of Llama2-7b, with their original perplexity for comparison.

Model	efficacy	generalization	locality
GPT-2-XL	5.19%	14.29%	97.40%
GPT-J	30.59%	30.77%	82.35%
Llama2-7b	18.65%	12.70%	100%

Table 4: Performance of C-ROME on various LLMs for corresponding collapse cases. Notably, the efficacy in normal cases typically exceeds 90%.

second tokens, replicating the position embedding of the first token onto the second token does not consistently lead to collapse. These findings suggest that while position embedding plays a role, it is not the only determining factor. The detailed investigation is provided in Appendix A.4.

Additionally, we observed that in GPT-2-XL and GPT-J, the representations of the first tokens rapidly become significantly more concentrated than those of subsequent tokens as the layers progress. However, this phenomenon does not appear in Llama2-7b, Mistral-7b, and Llama3-8b. A detailed investigation is presented in Appendix A.5.

Regarding the collapse cases of Llama2-7b, we found that the subjects of them terminate with a period ".". It is worth noting that, such cases are extremely rare, amounting to just 21 out of 21,919 samples in the COUNTERFACT dataset. Furthermore, they do not induce model collapse in various other models, including GPT-2-XL, GPT-J, Mistral-7b and Llama3-8b (the successor of Llama2-7b), as shown in Table 3. Consequently, we have decided not to pursue an exhaustive investigation of this isolated phenomenon.

4 A Simple Solution to Avoid Collapse

Having identified the reasons for ROME's collapse, it is crucial to provide a solution to prevent these problems. C-ROME introduced in § 3.1 can effectively keep the stability of edited models, but Table 4 reveals that it fails to successfully integrate target knowledge into the model, as evidenced by its low *efficacy* and *generalization* (Yao et al., 2023) metrics on collapse cases. This failure arises from the inconsistency of C-ROME between editing and testing. Specifically, C-ROME employs prefixed keys $\bar{\boldsymbol{k}}$ only when editing, while during testing, the prompts used to evaluate efficacy adopt unprefixed keys $\boldsymbol{k}^u$, which significantly differ from $\bar{\boldsymbol{k}}$. This inconsistency results in an inability to obtain the appropriate target value vector corresponding to the key of collapse cases, finally leading to low efficacy of editing.

Model	Cases	efficacy	generalization	locality
GPT-2-XL	collapse	100%	16.88%	100%
	normal	96.16%	41.88%	97.34%
GPT-J	collapse	100%	32.94%	89.41%
	normal	99.77%	50.00%	95.61%
Llama2-7b	collapse	91.27%	29.37%	100%
	normal	91.95%	46.73%	97.56%

Table 5: Performance of C-ROME, enhanced by prefixing random texts to the prompts of collapse cases during testing, across various LLMs on both collapse cases and the remaining data within COUNTERFACT.

To address this issue, we propose a straightforward solution, which appends a random prefix, drawn from those utilized in the editing process, to the prompt of collapse cases during the testing phase. The results in Table 5 demonstrate that this method significantly improves the efficacy for GPT-2-XL, GPT-J, and Llama2-7b, albeit with a relatively limited improvement of generalization.

5 Conclusion and Future Work

In this paper, we thoroughly investigate the underlying causes of LLMs collapse triggered by a single edit of ROME. Our extensive experiments demonstrate that such collapse arises from two aspects: i) irregularities in the official implementation of ROME, which employs two types of keys in parameter updating; ii) anomalous representation distribution of the first token in autoregressive models. Consequently, we propose a straightforward and simple method to address the model collapse issue of ROME, and validate its effectiveness with extensive experiments

For future research, we intend to investigate the root causes of model collapse in sequential editing and to devise more robust editing methods that ensure the stability of the edited model and superior editing performance across various scenarios.

Limitations

We acknowledge following limitations of our work:

- The analysis in this paper primarily focuses on GPT-2-XL and GPT-J. Regarding Llama2-7b, which exhibits a unique pattern of collapse cases, our solution successfully prevents its collapse. However, the specific characteristics of its collapse cases remain unknown.
- Due to space limitations, we have left an in-depth investigation into the anomalous representation distribution of the first token in autoregressive models for future research. This anomaly represents a broader issue that requires further exploration.
- This paper focuses on the root causes of model collapse triggered by a single edit of ROME. The collapse resulting from the cumulative effects of sequential editing, a phenomenon observed in existing works, is beyond the scope of this paper and is reserved for future work.

Acknowledgements

This work was supported by the National Key R&D Program of China (2022YFB3103700, 2022YFB3103704), the Strategic Priority Research Program of the Chinese Academy of Sciences (No. XDB0680101), and the Innovation Funding of ICT, CAS (E361120).

References

Jia-Chen Gu, Hao-Xiang Xu, Jun-Yu Ma, Pan Lu, Zhen-Hua Ling, Kai-Wei Chang, and Nanyun Peng. 2024. Model editing can hurt general abilities of large language models. *Preprint*, arXiv:2401.04700.

Akshat Gupta, Sidharth Baskaran, and Gopala Anumanchipalli. 2024a. Rebuilding rome : Resolving model collapse during sequential model editing. *Preprint*, arXiv:2403.07175.

Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. 2024b. Model editing at scale leads to gradual and catastrophic forgetting. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 15202–15232, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. `https://ai.meta.com/blog/meta-llama-3/`.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

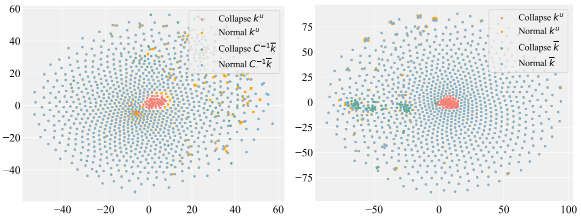
Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. `https://github.com/kingoflolz/mesh-transformer-jax`.

Wanli Yang, Fei Sun, Xinyu Ma, Xun Liu, Dawei Yin, and Xueqi Cheng. 2024. The butterfly effect of model editing: Few edits can trigger large language models collapse. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 5419–5437, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

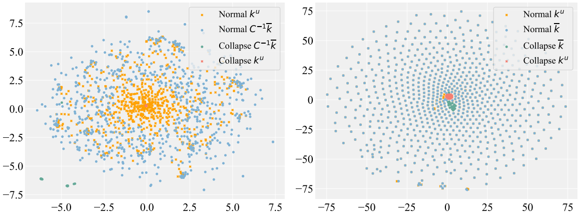
Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10222–10240, Singapore. Association for Computational Linguistics.

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and Huajun Chen. 2024. A comprehensive study of knowledge editing for large language models. *Preprint*, arXiv:2401.01286.



(a) (b)

Figure 4: t-SNE visualization of (a) elements in the denominator; (b) different implementation of key vectors for GPT-J.



(a) (b)

Figure 5: t-SNE visualization of (a) elements in the denominator; (b) different implementation of key vectors for Llama2-7b.

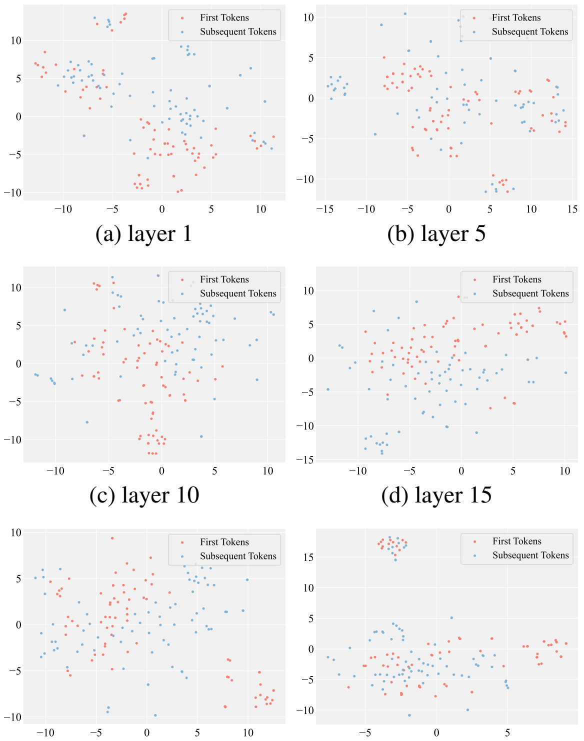
A Appendix

A.1 Distribution of Keys in Other LLMs

The distribution of $C^{-1}\overline{\boldsymbol{k}}$ and $\boldsymbol{k}^u$ for collapse and normal cases of GPT-J in two-dimensional space is shown in Figure 4a, demonstrating a significant difference between the distributions of these two elements in collapse cases. The results for $\overline{\boldsymbol{k}}$ and $\boldsymbol{k}^u$ is depicted in Figure 4b, revealing similar disparities. The corresponding results for Llama2-7b are provided in Figure 5a and Figure 5b, showing consistent phenomena.

A.2 Results without Prepended Token

To validate that the absence of collapse in Llama2-7b, Mistral-7b, and Llama3-8b for the collapse cases of GPT-2-XL and GPT-J, is due to the addition of a prefix token, we manually removed the prepended token of these models, thereby positioning the unprefixed key $\boldsymbol{k}^u$ of the collapse cases as the first token of the input. In this setting, we employed ROME to edit these three models on the collapse cases of GPT-2-XL and GPT-J. The results presented in Figure 7 indicate that Llama2-7b, Mistral-7b, and Llama3-8b also succumb to collapse after editing.



(a) layer 1 (b) layer 5

(c) layer 10 (d) layer 15

(e) layer 20 (f) layer 23 (last layer)

Figure 6: t-SNE visualization of representations for first tokens and subsequent tokens across various layers in the encoder of T5-3B.

A.3 Representation of First Token in T5-3B

The anomalous representation distribution of the first tokens in autoregressive models may be attributed to their inability to interact with subsequent tokens. To verify it, we take the encoder-decoder model T5-3B as a counterexample and analyze the representation distribution of the first tokens in the collapse cases compared to an equal number (77) of subsequent tokens from the normal cases across various layers in its encoder. The results in Figure 6 indicate that there is no significant difference between the representations of the first token and subsequent tokens, corroborating our hypothesis.

A.4 Impact of Position Embedding

In this section, we conducted experiments on GPT-2-XL, GPT-J, and Llama2-7b to investigate whether the anomalous distribution of the first token is attributable to its position embedding. For Llama2-7b, we removed the special token <s> that the tokenizer additionally prepends at the beginning of the input to maintain consistency with GPT-2-XL and GPT-J.

For collapse cases where the subjects are the first tokens, we set the position embedding of the

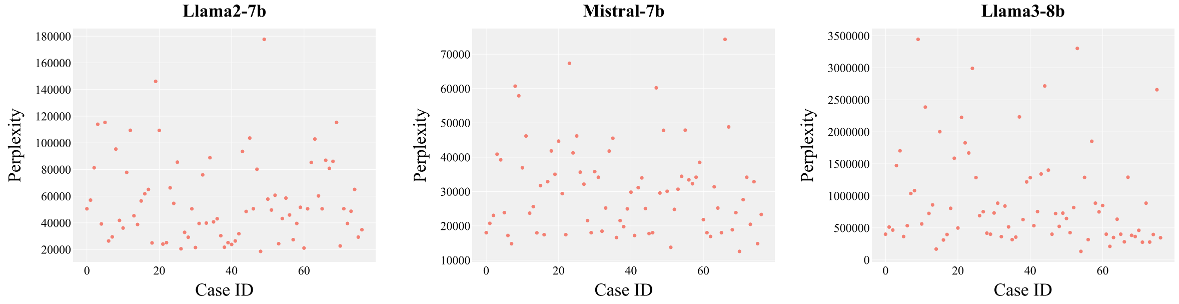


Figure 7: Scatter plot of perplexity for Llama2-7b, Mistral-7b, and Llama3-8b models edited by ROME, with each point representing a unique edit case in the collapse cases of GPT-2-XL and GPT-J. "Case ID" refers to the index of each edit sample.

Model	Perplexity	Original	Second2First
GPT-2-XL	min	2177.82	1008.21
	avg	19,877.79	1397.87
	max	179,185.99	2153.86
GPT-J	min	5094.73	8153.70
	avg	28,835.21	26,978.14
	max	85,936.24	124,982.41
Llama2-7b	min	16,279.75	17,561.97
	avg	67,436.51	72,692.50
	max	206,307.60	349,577.58

Table 6: The minimum, average, and maximum perplexity observed in collapse cases when utilizing the original position embeddings (Original) and when assigning the first token's position embedding as that of the second token (Second2First) for various LLMs.

Model	Perplexity	Original	First2Second
GPT-2-XL	min	68.55	81.39
	avg	68.81	39,714.90
	max	69.03	912,001.20
GPT-J	min	48.80	48.47
	avg	49.03	48.68
	max	49.50	49.48
Llama2-7b	min	32.83	33.14
	avg	33.32	2104.90
	max	37.03	42,154.10

Table 7: The minimum, average, and maximum perplexity observed in normal cases when utilizing the original position embeddings (Original) and when assigning the second token's position embedding as that of the first token (First2Second) for various LLMs.

first token as that of the second token (Noted as Second2First). The results presented in Table 6 indicate that this approach mitigates model collapse on GPT-2-XL, but it is completely ineffective on GPT-J and Llama2-7b.

For normal cases where the subjects are the second tokens, we assign the position embedding of the second token as that of the first token (Noted as First2Second). The results in Table 7 reveal that this change leads to partial model collapse in GPT-2-XL and Llama2-7b, but all edited models of GPT-J remain stable.

The results from the two aforementioned aspects suggest that position embedding may be a contributing factor to the abnormal representation of the first token, but it is not the sole factor.

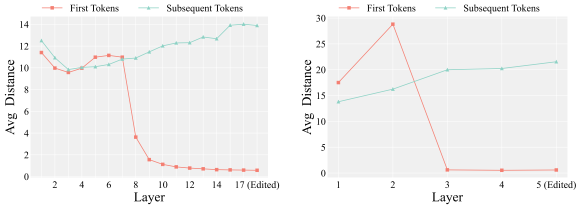
A.5 Collapse of First Token Representation

From Figure 2 and Figure 3, we observed an unusual phenomenon that the collapse keys $\boldsymbol{k}^u$ (i.e., representations of the first tokens) appear to be more concentrated than the normal keys $\boldsymbol{k}^u$ (i.e., representations of the subsequent tokens). To assess the degree of aggregation of the first tokens and subsequent tokens, we calculated the average distance of each element from the cluster center for both the first tokens and all the subsequent tokens, denoted as $D\left(F\right)$ and $D\left(S\right)$, correspondingly.

$$D = \frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{e}_i - \frac{1}{N}\sum_{k=1}^{N}\boldsymbol{e}_k\right\|_2 \qquad (4)$$

Here, $\boldsymbol{e}_i$ and $\boldsymbol{e}_k$ represent the embeddings of the i-th and k-th tokens, which are the outputs of the first MLP layer within the transformer block.

With this metric established, we computed the values within the edited layers of GPT-2-XL, yielding $D\left(F\right)$ being 0.578 and $D\left(S\right)$ being 13.895. The result suggests a markedly higher concentration in the representations of the first tokens com-



(a) GPT-2-XL

(b) GPT-J

Figure 8: Average distances of each element from the cluster center for the first tokens and the subsequent tokens, across layers from the first layer to the edited layer in GPT-2-XL and GPT-J.

pared to those of subsequent tokens. This observation raises a further question: *Given that different first tokens have distinct embeddings when input into the transformer, why are their representations in the middle layers so closely concentrated?*

To investigate this, we computed the distances $D(F)$ and $D(S)$ from the first layer to the edited layer (layer 17) in GPT-2-XL. As depicted in Figure 8a, prior to layer 8, $D(F)$ and $D(S)$ exhibit no significant divergence. However, post layer 8, the representations of the first tokens rapidly shrink. The same phenomenon is also observed in GPT-J, as shown in Figure 8b. However, our experimental results indicate that such phenomenon does not appear on Llama2-7b, Mistral-7b, and Llama3-8b. Consequently, we decide not to delve further into this particular aspect.

The underlying causes of the first token's representation concentration in GPT-2-XL and GPT-J remain unclear. A potential factor, as explored in Appendix A.3, is that within autoregressive LLMs, the first token cannot interact with subsequent tokens. Continuous self-interaction may lead to the contraction of its representation. Since this phenomenon is not related to the model collapse during editing examined in this paper, it has been remained for future research.