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

Anonymous ACL submission

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 analysis, we propose a simple yet effective approach: uniformly using prefixed keys during editing phase and adding prefixes during the testing phase. 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, and the value is a vector representation that can produce 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 matche the subject's key vector with the new fact's value vector, as described in Eq. 3.

To uncover the underlying causes, 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 in Eq. 3. This anomaly originates from the irregular implementation of the keys in the denominator, where one key is derived with varying prefixes (prefixed key) and another without any prefix (unprefixed key). This issue has also been independently identified by Gupta et al. (2024a) simultaneously. However, it is still unclear why the irregular implementation only leads to collapse in collapse cases.

To answer this question, we examine the distribution of elements in the denominator. We observe that, in collapse cases, the distribution of the unprefixed keys exhibits significant differences 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 collapse cases, we conduct an analysis starting from their characteristics. The collapse cases of GPT-2-XL (Radford et al., 2019) and GPT-J (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

[1]The code will be released after the review process ends.

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. When using the edited model, we prepend a random text prefix for instances where subjects are in the first token to ensure consistency with the editing process. Experiments validate that our proposed method effectively prevents model collapse while ensuring the success of edits.

Our main contributions are as follows:

- We perform comprehensive analyses to identify two factors behind ROME's collapse: i) inconsistent implementation of key vectors; ii) anomalous distribution of first token representations.
- We propose a straightforward solution to prevent collapse while maintaining editing efficacy.

2 Background

ROME (Meng et al., 2022) hypothesizes that MLP modules in Transformer architecture (Vaswani et al., 2017) can be modeled as a linear key-value associative memory (Geva et al., 2021). Under this hypothesis, a knowledge triplet (s, r, o) corresponds to a key-value pair (k, v), where k represents the subject s, and v encodes the property (r, o) for s. The entire knowledge within a model can thus be represented as a set of key vectors $K = [k_1, \cdots, k_n]$ and value vectors $V = [v_1, \cdots, v_n]$. A linear operation W matches keys to values by solving $WK \approx V$. In practice, for two-layer MLP in a specific transformer block determined by a Causal Tracing mechanism (Meng et al., 2022), outputs of the first layer form a key k, and the second layer (parameterized with W) retrieves an associated value v based on this key k.

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 vector k_* and the new value vector v_*. To simulate various contexts for generalization, ROME assigns k_* as an average vector derived from subject s with a small set of N randomly sampled prefixes:

$$k_* = \frac{1}{N}\sum_{i=1}^{N} \mathcal{K}\left(x_i \oplus s\right), \qquad (1)$$

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 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 v_* so that it can produce $o^* =$ *Joe Biden* when provided with the prompt $\mathrm{p}(s, r) =$ *The president of the United States is*.

Component	Cases	GPT-2-XL	GPT-J	Llama2-7b
numerator: $(v_* - Wk_*)\left(C^{-1}k_*\right)^\top$	collapse	168.55	140.27	4.57
	normal	79.91	88.69	16.52
denominator: $\left(C^{-1}k_*\right)^\top k_*$	collapse	0.04	0.04	0.01
	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.

Given the computed (k_*, v_*), ROME finds optimal $\widehat{W}$ to solve the following problem:

$$\underset{\widehat{W}}{\arg\min}\|\widehat{W}K - V\| \;\; \text{subject to } \widehat{W}k_* = v_* \qquad (2)$$

It has the following closed-form solution:

$$\widehat{W} = W + \underbrace{\frac{(v_* - Wk_*)\left(C^{-1}k_*\right)^\top}{\left(C^{-1}k_*\right)^\top k_*}}_{\text{update matrix } \Delta} \qquad (3)$$

where W denotes the weight matrix of the second layer of the MLP before editing, $\widehat{W}$ denotes the weight matrix after editing, and $C{=}KK^\top$ is a pre-cached constant. Interested readers are directed to Meng et al. (2022) for a detailed introduction.

3 Why Does ROME Cause Collapse?

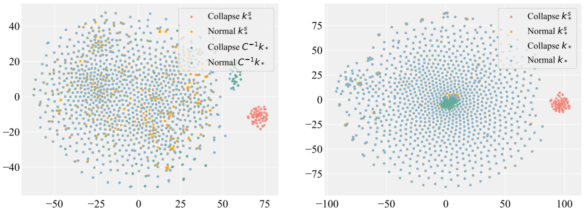
Previous studies (Yang et al., 2024; Gupta et al., 2024b) have revealed that a single edit of ROME can induce LLMs to collapse. To further analyze the cause, we investigate the differences in parameter updates between samples that induce collapse and those that do not. For this purpose, we introduce 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

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.



(a) (b)

Figure 1: t-SNE visualization of (a) elements in the denominator; (b) different implementation of key vectors.

cases, while the numerators do not show significant differences. This disparity directly results in the exceptionally large Δ of collapse cases.

These results guide our focus to the denominator $(C^{-1}k_*)^\top k_*$. Given that the matrix C is a constant for both collapse cases and normal cases, our analysis is primarily focused on the key k_*. We revisited the official implementation of ROME and identified that **different variants of k_* are used**. Specifically, only k_* within $(C^{-1}k_*)^\top$ is the prefixed key as in Eq. 1. In contrast, k_* **in other positions is unprefixed**, utilizing a representation over the subject s without any prefix, denoted as $k_*^s = \mathcal{K}(s)$. However, ideally, all key k_* in Eq. 3 should be the same vector, 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 k_*^s with 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 k_* does not significantly alter the edited models, avoiding the sharp increase in perplexity seen with ROME. This demonstrates that such inconsistency of 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 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 $C^{-1}k_*$ and k_*^s 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 1a as an example, in normal cases, the distribution of $C^{-1}k_*$ and k_*^s show no significant differences. However, a noticeable divergence in the distribution occurs in collapse cases, explaining the exceptionally small denominators.

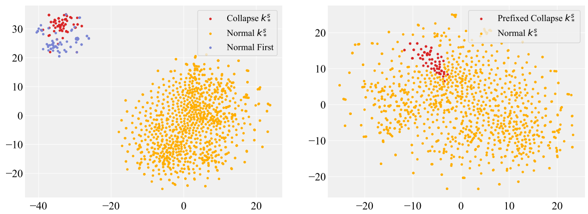
Considering that C is a constant, the distinctions between normal and collapse cases should arise from the variations in the prefixed key k_* and the unprefixed key k_*^s. Figure 1b clearly illustrates that the distribution of k_*^s in collapse cases significantly diverge from those of k_*. This confirms that in collapse cases, the significant differences between k_* and k_*^s 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 k_*^s in collapse cases, we focus our analysis on their characteristics. We observed a common pattern 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* $\mathtt{p}(s, r)$[2]. Therefore, the unprefixed key k_*^s for a collapse case is the intermediate representation within the MLP layer of the first token in the input prompt. This inspires us to investigate whether the anomalous distribution of k_*^s in collapse cases can be attributed to their place 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 2a indicate that, within GPT-2-XL, the first tokens of normal cases consistently exhibit an abnormal distribution similar to that of k_*^s in collapse cases. From an opposing perspective, to verify whether artificially shifting the k_*^s 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

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



(a) (b)

Figure 2: t-SNE visualization of representation distributions of (a) the first token in randomly sampled normal prompts; (b) k_*^s in prefixed collapse prompts.

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

of normal cases, as illustrated in Figure 2b. The results suggest that the anomalous distribution of k_*^s 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-like models.

We speculate that this phenomenon arises 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 first tokens 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.2.

It is important to note that Llama2-7b (Touvron et al., 2023) avoids collapse in such cases due to its tokenizer incorporating a special token, <s>, at the beginning of the encoding process, which shifts the subject from being the first token. In fact, we found that Llama2-7b also succumbs to collapse when the special token <s> is not prepended, with results detailed in Appendix A.3.

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 3 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. The reason is that C-ROME employs prefixed keys k_* only when editing. However, during the evaluation of collapse cases, the prompts used to assess efficacy adopt unprefixed keys k_*^s, which significantly differs from k_*. This inconsistency results in an inability to obtain the appropriate target value vector corresponding to the key of collapse cases, finally leading to a failure in efficacy.

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	12.70%	12.70%	100%
	normal	91.95%	46.73%	97.56%

Table 4: 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 during the testing phase to the prompt for cases where the key corresponds to the first token. The results in Table 4 demonstrate that this method significantly raises the efficacy for both GPT-2-XL and GPT-J, albeit with a relatively limited improvement of generalization. The suboptimal performance on the collapse cases of Llama2-7b is due to their different pattern from that observed in other two models. Nonetheless, such cases are extremely rare (21 out of 21,919 in the COUNTERFACT dataset), and their collapse has effectively been avoided.

5 Conclusion and Future Work

In this paper, we conduct a thorough investigation into the underlying causes of LLM's 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 distribution of the first token in GPT-like models. Consequently, we propose a straightforward method to address the model collapse issue of ROME, and conduct experiments to validate its effectiveness.

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.

References

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.

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. *Preprint*, arXiv:2401.07453.

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.

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. 30.

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. *arXiv preprint arXiv:2402.09656*.

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.

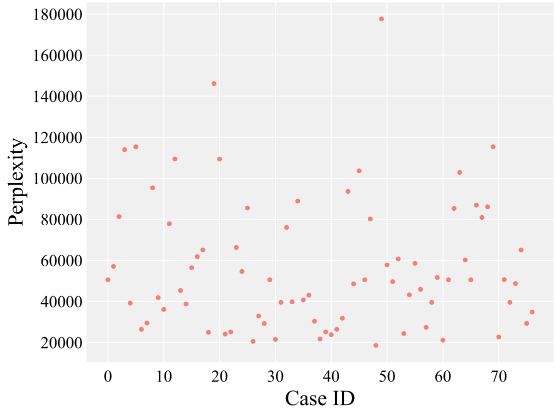


Figure 3: Scatter plot of perplexity for Llama2-7b models edited by ROME, with each point representing a unique edit case in the collapse case of GPT-2-XL. "Case ID" refers to the index of each edit sample.

A Appendix

A.1 Distribution of Keys in Other LLMs

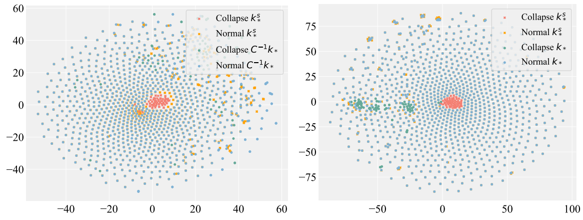
The distribution of $C^{-1}k_*$ and k_*^s for collapse and normal cases of GPT-J in two-dimensional space is shown in Figure 4a, demonstrating a significant difference between the distributions of these two elements in collapse cases. The results for k_* and k_*^s is depicted in Figure 4b, revealing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

A.2 First token in T5-3B

To explore whether the anomalous distribution of the first tokens in GPT-like models can be attributed to their inability to interact with subsequent tokens within autoregressive models, we take the encoder-decoder model T5-3B as an counterexample and observe the distribution of an equal number of first tokens compared to subsequent tokens 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.

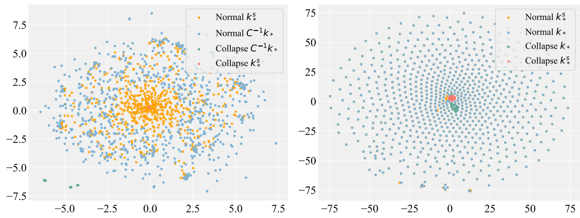
A.3 Llama2-7b without Prepended Token

We manually removed the prepended token <s> in Llama2-7b, thereby positioning the key k_*^s of the collapse case as the first token of the input. In this setting, we employed ROME to edit Llama2-7b on the collapse cases of GPT-2-XL. The results presented in Figure 3 indicate that Llama2-7b also succumbs to collapse after editing.



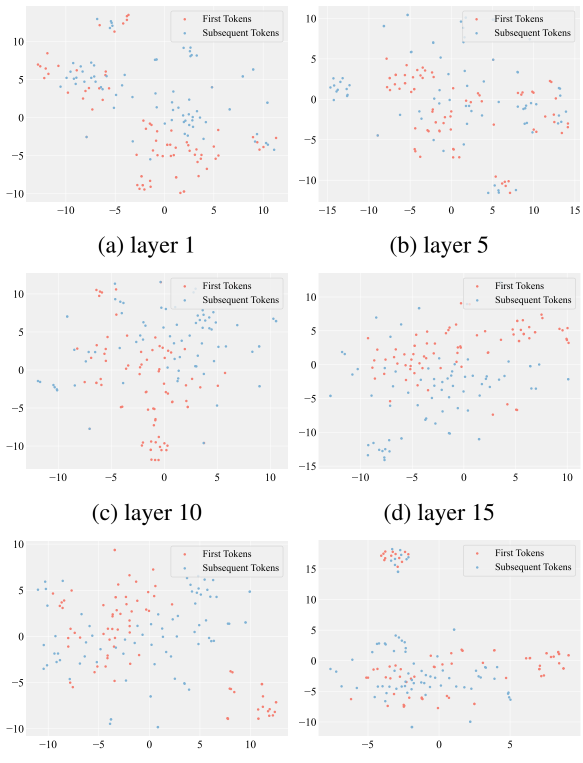
(a) (b)

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) 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.