

# Sparse Latents Steer Retrieval-Augmented Generation

Chunlei Xin<sup>1,2</sup>, Shuheng Zhou<sup>3</sup>, Huijia Zhu<sup>3,\*</sup>, Weiqiang Wang<sup>3</sup>, Xuanang Chen<sup>1</sup>,  
Xinyan Guan<sup>1,2</sup>, Yaojie Lu<sup>1</sup>, Hongyu Lin<sup>1</sup>, Xianpei Han<sup>1,2,\*</sup>, Le Sun<sup>1,2</sup>

<sup>1</sup>Chinese Information Processing Laboratory, Institute of Software, Chinese Academy of Sciences

<sup>2</sup>University of Chinese Academy of Sciences

<sup>3</sup>Ant Group

{chunlei2021, guanxinyan2022}@iscas.ac.cn,

{chenxuanang, luyaojie, hongyu, xianpei, sunle}@iscas.ac.cn,

{shuheng.zsh, huijia.zhj, weiqiang.wq}@antgroup.com

## Abstract

Understanding the mechanisms underlying Large Language Model (LLM) behavior in Retrieval-Augmented Generation (RAG) systems is critical for enhancing reliability. In this paper, we leverage Sparse Autoencoders (SAEs) within the LLaMA Scope to uncover sparse, interpretable latents that govern RAG behaviors. Through systematic analysis of SAE activations, we identify specific latents associated with two fundamental RAG decisions: (1) context versus memory prioritization, and (2) response generation versus query rejection. Intervention experiments demonstrate that these latents enable precise control over model behavior and maintain generalizability across various experimental settings. Mechanistic analysis reveals that manipulating these latents influences model behavior by reconfiguring attention patterns of retrieval heads. Our findings establish SAEs as a principled tool for understanding and controlling RAG behaviors, demonstrating capabilities in precise behavior steering without architectural modifications.

## 1 Introduction

To address the limitations of large language models (LLMs) in handling knowledge-intensive tasks (Petroni et al., 2019; Ji et al., 2023; Mallen et al., 2023), Retrieval-Augmented Generation (RAG) (Lee et al., 2019; Karpukhin et al., 2020; Lewis et al., 2020) has emerged as an advanced approach for integrating external knowledge sources into the generation process. Despite its effectiveness in expanding knowledge boundaries, RAG system still faces several challenges that hinder its effectiveness, as illustrated in Figure 1. First, LLMs often struggle to resolve conflicts between retrieved context and internal memory, oscillating between over-relying on outdated knowledge and over-trusting noisy passages (Longpre et al., 2021; Chen et al.,

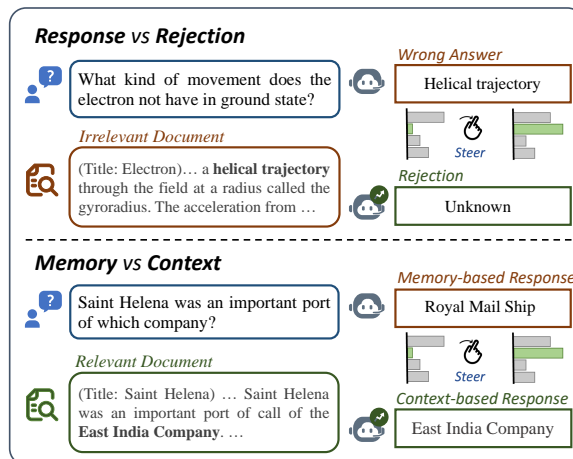


Figure 1: RAG systems often struggle to resolve conflicts and properly reject questions based on the information available. Intervention on corresponding SAE latents can effectively alter RAG behavior.

2022; Xie et al., 2024). Second, inconsistent query rejection mechanisms often lead to unreliable responses when context is insufficient or result in rejecting valid queries due to miscalibrated confidence (Wen et al., 2024; Feng et al., 2024; Lee et al., 2024). These challenges stem from an opaque decision-making process within RAG: how LLMs internally manage context usage, resolve knowledge conflicts, or trigger rejections remains unclear. Therefore, understanding the mechanisms underlying RAG behavior is crucial for improving reliability and controllability.

Recently, many studies have attempted to understand the decision-making process of LLMs in RAG systems. Some researchers analyze attention patterns to identify how models attend to context tokens (Wu et al., 2024; Sun et al., 2025), but reveal only surface-level input-output correlations. Some studies utilize mechanistic interpretability techniques (Vast et al., 2024; Sun et al., 2025) to explore how specific components influence RAG behavior, but face challenges in isolating individ-

\*Corresponding Authors.

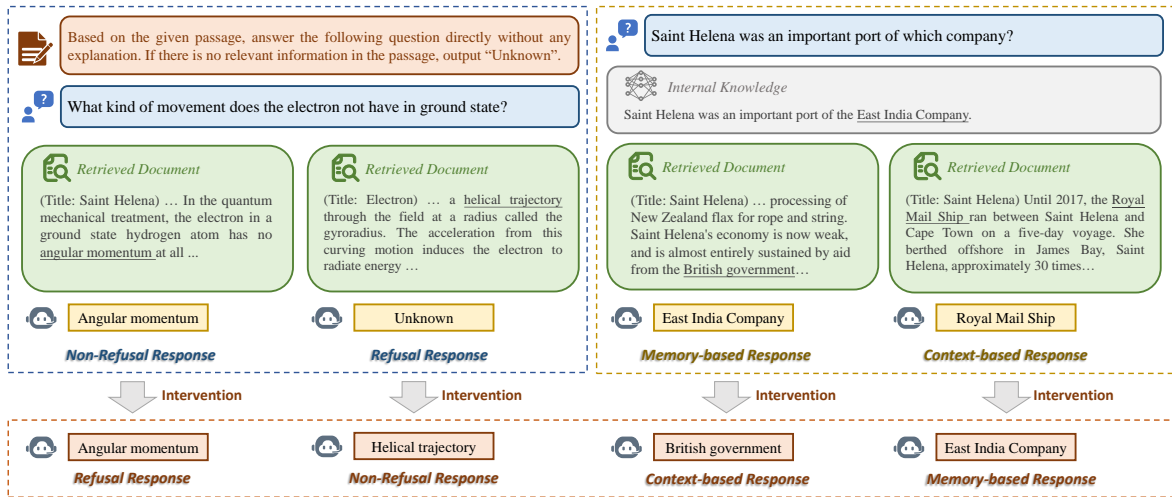


Figure 2: We construct a dataset focused on two categories of RAG behavior: whether outputs derive from internal memory or external context, and whether the model refuses to answer. Intervention on corresponding SAE latents can alter the LLM behavior.

ual causal effects due to network complexity. Research focusing on knowledge conflict (Chen et al., 2022; Jin et al., 2024; Xie et al., 2024) detects memory-context discrepancies, but typically relies on black-box methods with limited insight into internal causal mechanisms. While these methods partially illuminate RAG behaviors, they fall short of explaining high-level decision processes or enabling direct behavioral adjustments during RAG.

We hypothesize that sparse, interpretable representations within LLMs encode the fundamental RAG decision mechanisms. To explore this hypothesis, we employ Sparse Autoencoders (SAEs) (Bricken et al., 2023; Huben et al., 2024) within the LLaMA Scope framework (He et al., 2024b) to decompose model activations into sparse and interpretable latents. Our investigation centers on a foundational research question: **Can we uncover and control the intrinsic mechanisms governing specific RAG behaviors through sparse latents within LLMs?**

To answer this question, we first investigate whether SAEs can uncover interpretable latents that correspond to targeted RAG behaviors. Specifically, we focus on two fundamental RAG decisions: (1) context versus memory prioritization, and (2) response generation versus query rejection, as illustrated in Figure 2. By analyzing SAE latent activations across model layers, we identify meaningful latents that are strongly correlated with specific RAG behaviors. Intervention experiments demonstrate that amplifying the activity of these latents enables precise control over RAG behav-

ior, effectively modulating the context-following capability of the LLM during the RAG process.

To evaluate whether these identified latents maintain their effectiveness across practical scenarios, we extend the experiments across model variants, extended contexts and varied prompts. Experimental results reveal that top latents remain effective in steering RAG behavior when applied to the instruction-tuned model and longer contexts. However, only interventions on latents from early layers lead to expected outputs when the prompt changes. This suggests that fundamental RAG decisions are more likely to be encoded in early layers, whereas later layers focus more on specific tasks or tokens.

To uncover the internal mechanisms through which top latents steer LLM behavior during the RAG process, we further investigate their impact on context utilization mechanisms by analyzing retrieval heads, which are attention heads responsible for copy-paste behavior. Mechanistic analysis reveals that interventions on these latents modify attention patterns in retrieval heads, altering their focus between provided documents and refusal-related tokens. This attention redistribution provides direct evidence for the causal role of SAE latents in steering RAG behavior.

Overall, our main contributions can be summarized as follows:

- We uncover interpretable latents behind two core RAG decisions: (1) context versus memory prioritization and (2) response generation versus query rejection.

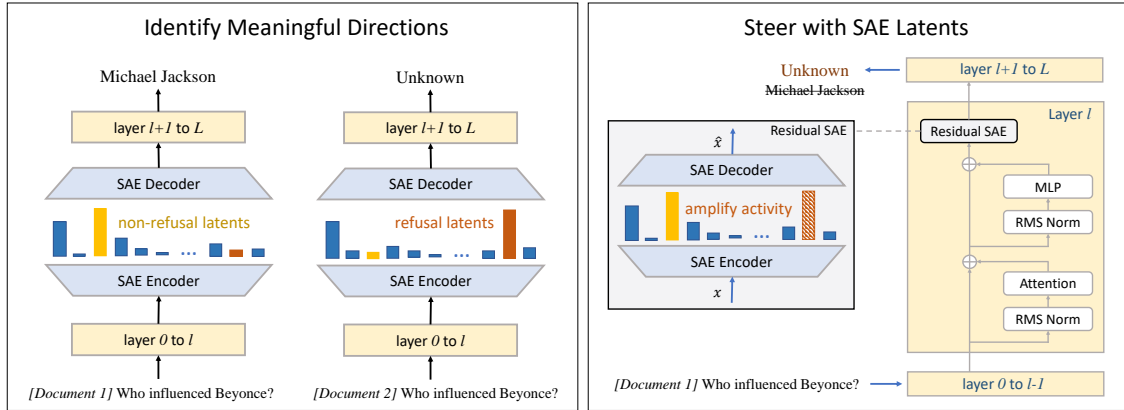


Figure 3: We identify SAE latents that correlate with different RAG behaviors in the residual stream of the final token of the input. Steering these SAE latents can effectively control the behavior of the model.

- We demonstrate precise control over RAG behavior through latent interventions, which can maintain effectiveness across model variants, scaled context lengths, and varied prompts.
- We discover that latent interventions influence LLM behavior by modifying attention patterns of retrieval heads, which either enhance context utilization or amplify refusal triggers.

## 2 Technical Background

### 2.1 Sparse AutoEncoders

SAEs are powerful interpretability tools designed to uncover sparse, interpretable decompositions of model representations. This approach is grounded in the Linear Representation Hypothesis (Mikolov et al., 2013; Park et al., 2024), which suggests that certain interpretable features of the input are represented as linear directions in the representation space. According to this hypothesis, the model’s learned representations can be seen as combinations of these linear directions, allowing for a potentially simpler and more interpretable structure.

Specifically, a sparse autoencoder typically consists of two hidden layers, which serve as the encoder and decoder functions, denoted as  $f(\cdot)$  and  $g(\cdot)$ , respectively. Given activations  $\mathbf{x} \in \mathbb{R}^n$  from a language model, the encoder maps  $\mathbf{x}$  to  $f(\mathbf{x}) \in \mathbb{R}^m$ , as shown in Eq. 1, and the decoder then reconstructs  $\hat{\mathbf{x}} \in \mathbb{R}^n$ , as illustrated in Eq. 2.

$$f(\mathbf{x}) := \sigma(\mathbf{W}_{\text{enc}} \cdot \mathbf{x} + \mathbf{b}_{\text{enc}}) \quad (1)$$

$$\hat{\mathbf{x}} := g(f(\mathbf{x})) = \mathbf{W}_{\text{dec}} \cdot f(\mathbf{x}) + \mathbf{b}_{\text{dec}} \quad (2)$$

wherein  $\mathbf{W}_{\text{enc}} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{b}_{\text{enc}} \in \mathbb{R}^m$ ,  $\mathbf{W}_{\text{dec}} \in \mathbb{R}^{n \times m}$ , and  $\mathbf{b}_{\text{dec}} \in \mathbb{R}^n$  are the weight and bias

terms. These two functions are trained to map  $\hat{\mathbf{x}}$  back to  $\mathbf{x}$ , which makes them an autoencoder. Thus,  $f(\mathbf{x})$  defines how the  $m \gg n$  columns of  $\mathbf{W}_{\text{dec}}$  combine to reconstruct  $\mathbf{x}$ . In other words, the columns of  $\mathbf{W}_{\text{dec}}$  represent a dictionary of directions along which the SAE decomposes  $\mathbf{x}$ . We refer to these learned linear directions as latents to distinguish them from the conceptual “features” hypothesized to constitute the representation vectors of language models.

In this work, we employ the residual SAEs with an 8x expansion of the hidden size from the LLaMA Scope framework (He et al., 2024b), which are trained on the post-MLP residual streams in each layer of LLaMA-3.1-8B (Touvron et al., 2023). Specifically, these SAEs adopt a Top-K SAE variant that employs a threshold  $\theta$  to maintain an average of  $K$  active latents over the training set. Further details about this variant can be found in Appendix A. Throughout this paper, we denote the SAE latents as  $l[\text{Layer}][\text{Position}][\text{Expansion}]\mathbf{x}[\text{Latent id}]$ . For example, the 1000th column of  $\mathbf{W}_{\text{dec}}$  from an SAE trained on the post-MLP residual stream of layer 1 in LLaMA-3.1-8B, with an 8x expansion of the hidden size, is labeled as  $l1r\_8x\_1000$ .

### 2.2 Sparse AutoEncoders for RAG

#### 2.2.1 Identifying Corresponding SAE Latents

To identify SAE latents that correlate with specific RAG behaviors, it’s essential to first construct a comparative dataset containing distinct categories of RAG behaviors: *target behavior* instances exhibiting desired RAG responses and *baseline behavior* instances displaying alternative re-

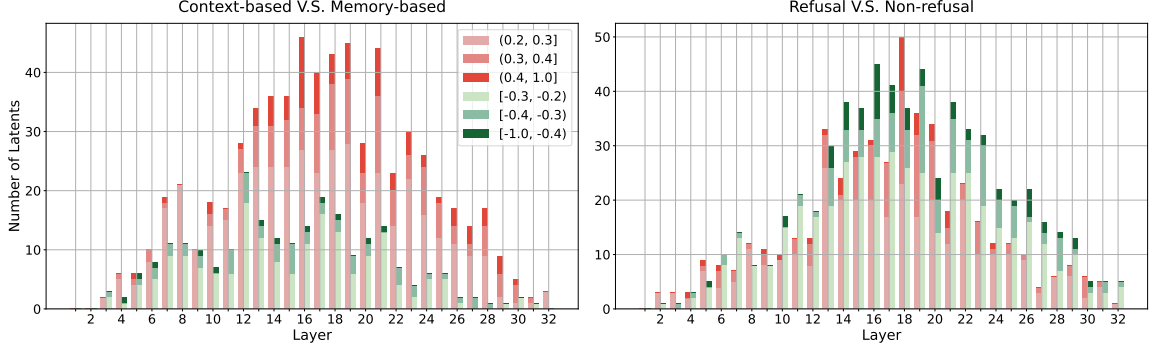


Figure 4: Layer-wise latent separation scores of LLaMA-3.1-8B. Latents with high absolute separation scores are more widely distributed across the middle layers.

sponses. For each layer  $l$ , given  $N^{\text{target}}$  target activations  $\{\mathbf{x}_{l,i}^{\text{target}}\}_{i=1}^{N^{\text{target}}}$  and  $N^{\text{baseline}}$  baseline activations  $\{\mathbf{x}_{l,i}^{\text{baseline}}\}_{i=1}^{N^{\text{baseline}}}$ , where  $N^{\text{target}}$  and  $N^{\text{baseline}}$  represent the number of instances displaying target and baseline behavior respectively, each activation vector  $\mathbf{x}$  is fed into the SAE to calculate the activation frequency of latent  $j$  for both target and baseline behaviors:

$$r_{l,j}^{\text{target}} = \frac{\sum_i^{N^{\text{target}}} \mathbb{1} \left[ f^{(l)}(\mathbf{x}_{l,i}^{\text{target}})_j > 0 \right]}{N^{\text{target}}} \quad (3)$$

$$r_{l,j}^{\text{baseline}} = \frac{\sum_i^{N^{\text{baseline}}} \mathbb{1} \left[ f^{(l)}(\mathbf{x}_{l,i}^{\text{baseline}})_j > 0 \right]}{N^{\text{baseline}}} \quad (4)$$

Subsequently, the latent separation score  $s_{l,j}^{\text{target}}$  is computed by subtracting the activity frequency of baseline behavior from that of target behavior, resulting in  $s_{l,j}^{\text{target}} = r_{l,j}^{\text{target}} - r_{l,j}^{\text{baseline}}$ . A positive  $s_{l,j}^{\text{target}}$  indicates that the corresponding latent exhibits higher activity during the target behavior, while a negative value suggests a stronger association with the baseline behavior.

### 2.2.2 Steering with SAE Latents

After identifying SAE latents correlated with specific RAG behaviors, we can steer LLM behavior with these latents, as illustrated in Figure 3. Derived from Eq. 2, SAEs reconstruct  $\hat{\mathbf{x}}$  through a linear combination of columns from  $\mathbf{W}_{\text{dec}}$ , expressed as  $\hat{\mathbf{x}} := \sum_j \mathbf{W}_{\text{dec}}[:, j] \cdot f(\mathbf{x})_j$ . Thus, increasing or decreasing the activation value  $f(\mathbf{x})_j$  is equivalent to doing activation steering with the decoder latent (Turner et al., 2023; Ferrando et al., 2025):

$$\hat{\mathbf{x}}^{\text{new}} \leftarrow \hat{\mathbf{x}} + \alpha \mathbf{d}_j \quad (5)$$

where  $\alpha$  is a steering coefficient controlling the degree of latent activation, and  $\mathbf{d}_j \in \mathbb{R}^n$  denotes the latent corresponding to the  $j$ -th column of  $\mathbf{W}_{\text{dec}}$ .

In our study, the selection of the steering coefficient  $\alpha$  follows a two-stage process. Initially, as detailed in Appendix E, we determine the range  $[1, 80]$  for  $\alpha$ , which demonstrates general effectiveness in steering the LLM toward desired behaviors across nearly all sampled latents. In the second stage, we further identify the optimal  $\alpha$  values for each SAE latent within this range, ensuring that each SAE latent can reach its maximum potential in steering RAG behaviors.

## 3 SAEs Uncover Meaningful Directions

### 3.1 SAE Latents Identification

To find latents corresponding to specific RAG behavior, following the methodology described in Section 2.2.1, we first construct a dataset focused on two categories of RAG behaviors: (1) whether outputs derive from internal memory or external context, and (2) whether the model refuses to answer. Figure 2 illustrates examples for each category, with detailed data collection procedures provided in Appendix B. From this dataset, 500 instances per category are selected to form a detection set. This set is designed to reliably identify SAE latents that govern specific RAG behavior while maintaining computational efficiency (details are provided in Appendix D). Additionally, 100 instances from each category are sampled to construct a validation set, which is used to determine the optimal steering coefficient  $\alpha$ . The remaining instances constitute a test set for evaluating the latent intervention effectiveness.

Specifically, we focus on latent activations within the residual stream of the final input to-



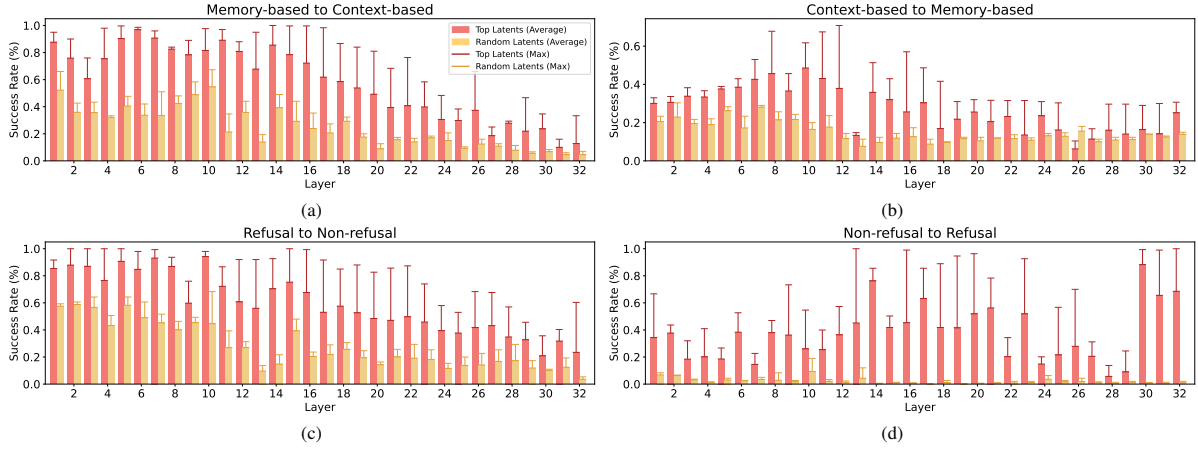


Figure 5: Comparison of average and maximum intervention success rates between top latents and random latents across different layers.

ken. Using Equation 3, we calculate latent activation frequencies for each scenario as  $r_{l,j}^{\text{memory}}$ ,  $r_{l,j}^{\text{context}}$ ,  $r_{l,j}^{\text{refusal}}$  and  $r_{l,j}^{\text{non-refusal}}$ . Subsequently, we compute latent separation scores as  $s_{l,j}^{\text{context}} = r_{l,j}^{\text{context}} - r_{l,j}^{\text{memory}}$  and  $s_{l,j}^{\text{refusal}} = r_{l,j}^{\text{refusal}} - r_{l,j}^{\text{non-refusal}}$ . A positive  $s_{l,j}^{\text{context}}$  indicates higher activity when LLM generates context-based answers, whereas a negative score suggests greater activity during memory-based responses. Similarly, a positive  $s_{l,j}^{\text{refusal}}$  indicates increased activity when the model refuses to respond, while a negative score implies higher activity when the model provides an answer based on context.

To investigate the distribution of latents that are highly correlated with specific RAG behaviors across different layers, we analyze layer-wise separation scores in Figure 4. Notably, latents exhibiting high absolute separation scores are more widely distributed across the middle layers, indicating that these layers play a crucial role in influencing the model’s context-following and refusal decisions during the RAG process.

### 3.2 Intervention on SAE Latents

Having identified SAE latents with high absolute values of separation scores, we hypothesize that these latents can be employed to steer RAG behaviors towards desired outcomes. To test this hypothesis, as shown in Figure 2, we define four steering objectives focused on the context-following capabilities of LLMs within RAG systems:

- **Inducing context-following behavior:** This includes (1) changing memory-based responses to context-based responses and (2)

altering refusal responses into non-refusal responses. Success requires achieving a ROUGE-1 score greater than 0.5 between the response and the provided document.

- **Suppressing context-following behavior:** This involves (1) shifting context-based responses to memory-based responses<sup>1</sup> and (2) altering non-refusal responses to refusal responses. Outputting the correct answer or the refusal token “unknown” without context reliance indicates success.

To evaluate intervention effectiveness, we select 300 questions from the corresponding test set for each objective. For each question, we amplify the activity of **one latent** highly associated with the target behavior at a time, and assess its impact on altering RAG behavior.

To identify the optimal layers for steering LLM behavior in RAG tasks, we evaluate the intervention success rate across different layers. Specifically, we focus on top latents  $d_j$  with high absolute separation scores ( $|s_{l,j}| > 0.3$ ). If fewer than three latents in a layer meet this criterion, additional latents with the highest absolute separation scores from that layer are included. For comparison, we also compute the intervention success rate using an equal number of randomly selected SAE latents at each layer. For each selected latent, we amplify its activity following Eq. 5, and assess its impact on steering RAG behavior by measuring the corresponding success rates. The average and

<sup>1</sup>To distinguish internal memory from hallucinations, we focus solely on scenarios where the model’s direct response (without context) contains the correct answer but gets misled by irrelevant documents.

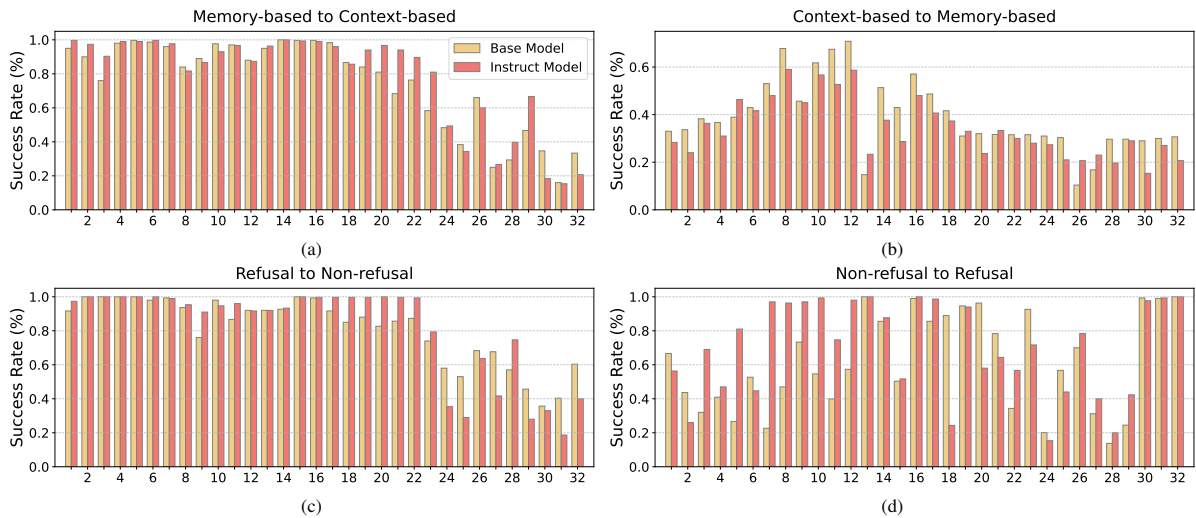


Figure 6: To investigate if top SAE latents identified in LLaMA-3.1-8B can generalize to model variants, we compare intervention success rates between the base model and its instruction-tuned variant across steering scenarios.

maximum intervention success rates across the top and random latents for each layer are illustrated in Figure 5.

**Finding 1: Separation scores effectively identify SAE latents that enable precise control over model behavior.** Compared with interventions on random latents, steering with top latents with high absolute separation scores significantly enhances precise control over RAG behavior. Across various steering scenarios, steering with top latents consistently exhibits higher average and maximum success rates. Notably, many of these top latents can be utilized to induce context-following capabilities in LLMs with nearly 100% success rates (Figure 5 (a) and (c)). This highlights that identifying and manipulating these top latents enables more accurate control over LLM behavior.

**Finding 2: Middle layers play a critical role in influencing LLM’s context-following and refusal decisions.** As illustrated in Figure 5 (a) and (c), steering the LLM towards providing context-based responses is most effective when steering with top latents from early to middle layers. Moreover, Figures 5(b) and (d) show that latents capable of suppressing context-following behaviors are primarily located in the middle layers. Interestingly, top latents in the final three layers also demonstrate remarkable effectiveness in altering non-refusal responses into refusal responses. The reason behind this is discussed in Section 4.3. These observations indicate that middle layers are crucial in determining whether the model provides context-aware responses or refuses to answer.

## 4 Generalizability of SAE Latents

### 4.1 Extending to Variant Model

In this section, we investigate whether the top SAE latents identified in LLaMA-3.1-8B can be leveraged to steer the behavior of its instruction-tuned variant, LLaMA-3.1-8B-Instruct. The results are summarized in Figure 6, where yellow bars indicate the intervention success rates for the base model (LLaMA-3.1-8B) using top latents that exhibit the highest intervention success rate from each layer, while red bars represent the corresponding success rates when amplifying the activity of these same latents to LLaMA-3.1-8B-Instruct.

**Finding 1: Top latents identified in LLaMA-3.1-8B can be directly applied to steer the behavior of LLaMA-3.1-8B-Instruct with comparable effectiveness.** Notably, top latents maintain comparable success rates in both models, exhibiting aligned layer-wise trends across steering scenarios. This suggests that the instruction-tuned variant reuses fundamental decision-making mechanisms from the base model, preserving the efficacy of the identified latents.

**Finding 2: Instruction tuning enhances the instruction-following capability of early layers.** Under the non-refusal-to-refusal steering scenario, LLaMA-3.1-8B-Instruct exhibits higher intervention success rates at early layers compared to LLaMA-3.1-8B. This indicates that instruction tuning improves the responsiveness to task instructions in early layers, reinforcing their role in instruction-following behaviors.

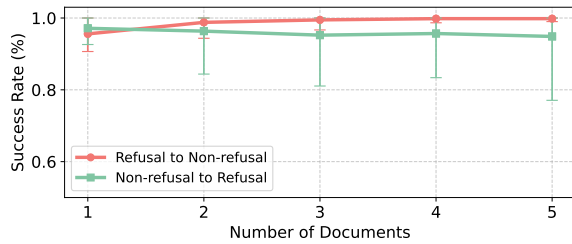


Figure 7: Average intervention success rate on top latents across varying numbers of documents.

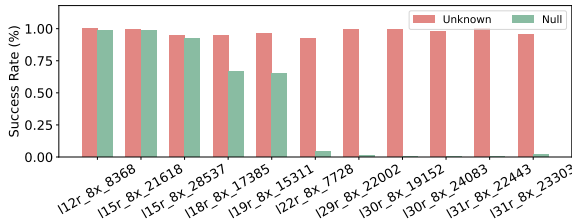


Figure 8: Intervention success rate of different latents varies across layers when the prompt changes.

## 4.2 Extending to Scaled Context Lengths

In Section 3.1, we identify meaningful directions using a single document as context. To validate the effectiveness on scaled input lengths, we expand the context to include 2 to 5 documents per question. Taking refusal-to-non-refusal and non-refusal-to-refusal steering scenarios as examples, we collect instances where the model exhibits specific behaviors under varying numbers of documents, and select 300 questions for each scenario. By steering with the corresponding top latents whose intervention success rates exceed 90%, we evaluate the average intervention success rate across different numbers of documents. The results are illustrated in Figure 7.

**Finding: Top latents remain effective in steering LLM behavior across scaled context lengths.** As the number of documents increases from 1 to 5, the average intervention success rates of the top latents remain consistently high, with the lowest success rate still around 80%. This consistency highlights the adaptability of top SAE latents to more complex contexts, suggesting their reliability in steering LLM behavior even as the amount of provided information grows.

## 4.3 Extending to Varied Prompts

Considering LLM’s sensitivity to prompt variations (Sclar et al., 2024; Voronov et al., 2024; He et al., 2024a), we examine whether the identified top latents remain effective in steering LLM be-

havior when the prompt changes. Focusing on the scenario of altering context-based responses to refusal responses, we modify the instruction guiding the LLM to refuse answering from “Unknown” to “NULL”. This modification tests whether top latents, which have been validated to successfully induce “Unknown” responses, can reliably guide the model to output “NULL” in response to the updated instruction.

Specifically, we select 300 questions where the model consistently provides context-based responses under both the original and updated prompts. By activating top latents with intervention success rates exceeding 90%, we assess their intervention effectiveness towards the updated prompt. Results in Figure 8 illustrate the intervention success rate under both the original and updated prompts.

**Finding: Early layers are more likely to provide SAE latents correlated with the underlying decision-making process.** Experimental results indicate that intervention success rates vary significantly across layers when the prompt changes. As shown in Figure 8, SAE latents from early layers (layer  $\leq 19$ ) exhibit stronger generalization capabilities compared to those from later layers, whose intervention success rate drops to approximately 0%. Case studies presented in Figure 14 reveal that latents from later layers are more closely associated with the “unknown” token rather than the underlying reject-to-answer behavior. These findings suggest that latents associated with specific prompts or tokens are probably distributed in later layers, while latents influencing the model’s fundamental decision-making process are more prevalent in early layers. This observation aligns with previous research (Liu et al., 2019; Rogers et al., 2020; Chen et al., 2023), which suggests that early layers typically exhibit stronger transferability.

## 5 Mechanistic Analysis

In this section, we utilize retrieval heads, which are attention heads responsible for contextual information extraction (Wu et al., 2024), to investigate the internal mechanisms through which top SAE latents steer RAG behavior effectively. The detailed process for identifying retrieval heads is described in Appendix F.

Firstly, we compare the attention scores that retrieval heads allocate to the input document versus the refusal token “Unknown” within the prompt

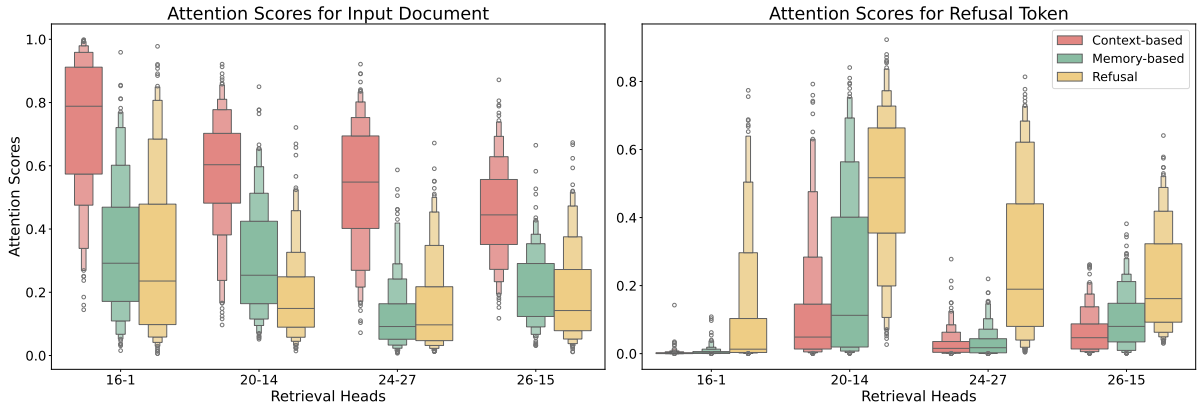


Figure 9: Comparison of attention scores allocated by retrieval heads to the given document versus to the refusal token “Unknown” within the prompt as the LLM exhibits different RAG behaviors.

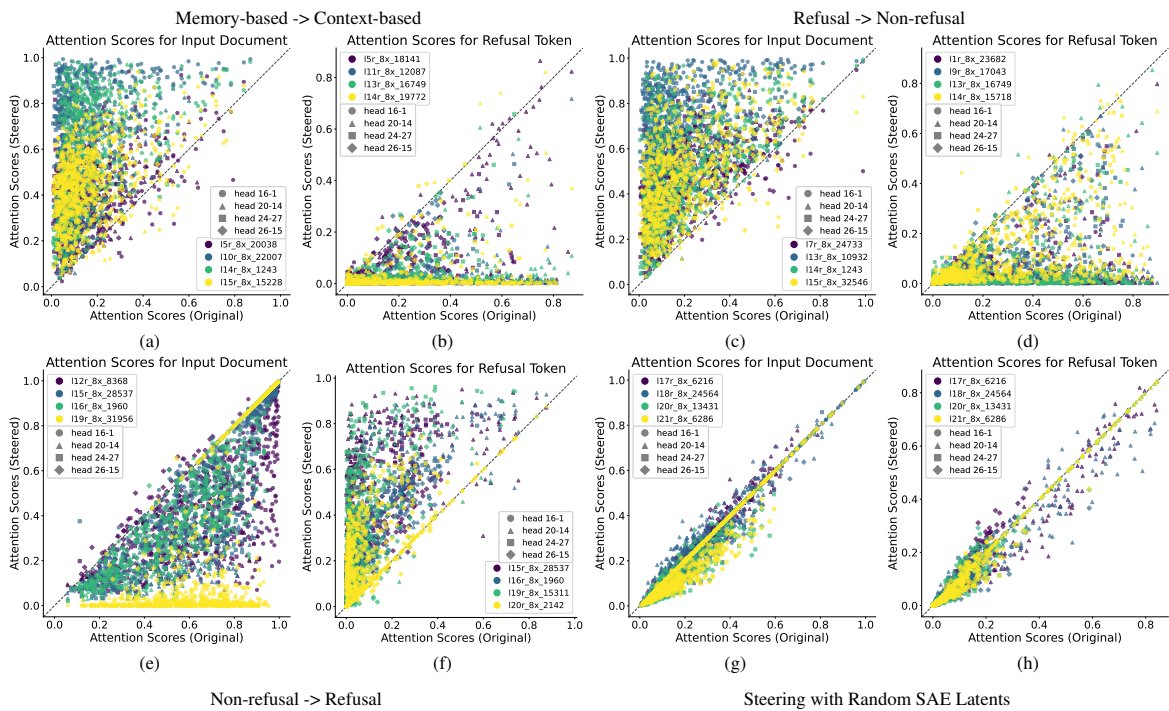


Figure 10: Steering with top latents correlated with inducing context-following behaviors increases attention on the given document and reduces attention on the refusal token “Unknown” (a, b, c, d). Activating latents strongly associated with rejection shows the opposite effect (e, f). Using random latents show no significant changes (g, h).

across different scenarios. Experimental results illustrated in Figure 9 demonstrate a significant disparity in attention patterns when the LLM exhibits different behaviors. When generating context-based responses, compared to providing memory-based responses or refusing to answer, retrieval heads strongly focus on the document while largely ignoring “Unknown” in the prompt.

Furthermore, changes in attention patterns of retrieval heads during intervention experiments are illustrated in Figure 10. Notably, as shown in Figure 10 (a)-(d), steering with top latents, which achieve over 90% success rates in inducing context-

following behaviors, results in reduced attention on “Unknown” and increased focus on the document. Conversely, activating top latents whose intervention success rates in inducing query rejection exceed 90%, as depicted in Figure 10 (e) and (f), increases attention on “Unknown” while diminishing document focus. For comparison, baseline experiments with four random SAE latents (Figure 10 (g) and (h)) show no significant changes. These findings provide insights into the causal mechanisms through which intervention on top latents influences context utilization and rejection decisions during the RAG process.



## 6 Conclusion

In this study, we uncover interpretable latents that highly correlate with specific RAG behaviors employing Sparse Autoencoders. Intervention experiments demonstrate that activating these latents enables precise control over RAG behavior. Furthermore, experiments conducted across model variants, scaled contexts and varied prompts not only demonstrate the generalizability of these SAE latents, but also underscore the inherent hierarchical nature of the model. Mechanistic analysis indicates that activating these latents alters the attention patterns of retrieval heads, influencing their focus on input context versus refusal tokens. These findings highlight the potential of SAEs as powerful tools for improving the interpretability and controllability of RAG systems in diverse applications.

## 7 Limitations

Our analysis focuses on the LLaMA-3.1-8B model, and the identified mechanisms may not generalize to models with distinct architectures. Future work is expected to validate whether these mechanisms persist across different model families and training paradigms. Additionally, like all sparse autoencoder methods, our findings depend on both the assumptions made by the SAE architecture and the quality of the trained SAEs.

We also note that our study of key RAG behaviors is based on short-form question answering. The effectiveness of our approach on more complex tasks, such as multi-hop reasoning or long-form content creation, has yet to be explored. Further investigation is needed to understand how SAE latents interact with decision-making processes in more complex tasks.

## Acknowledgments

We sincerely thank the reviewers for their insightful comments and valuable suggestions. This work was supported by Beijing Natural Science Foundation (L243006), Beijing Municipal Science and Technology Project (Nos. Z231100010323002), the Natural Science Foundation of China (No. 62306303, 62476265), CAS Project for Young Scientists in Basic Research (Grant No.YSBR-040) and Ant Group Research Fund.

## References

- Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermy, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*. <https://transformer-circuits.pub/2023/monosemantic-features/index.html>.
- Maheep Chaudhary and Atticus Geiger. 2024. [Evaluating open-source sparse autoencoders on disentangling factual knowledge in GPT-2 small](#). *CoRR*, abs/2409.04478.
- Hung-Ting Chen, Michael Zhang, and Eunsol Choi. 2022. [Rich knowledge sources bring complex knowledge conflicts: Recalibrating models to reflect conflicting evidence](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2292–2307, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Nuo Chen, Ning Wu, Shining Liang, Ming Gong, Linjun Shou, Dongmei Zhang, and Jia Li. 2023. [Is bigger and deeper always better? probing llama across scales and layers](#). *CoRR*, abs/2312.04333.
- Eoin Farrell, Yeu-Tong Lau, and Arthur Conmy. 2024. [Applying sparse autoencoders to unlearn knowledge in language models](#). *CoRR*, abs/2410.19278.
- Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. [Don't hallucinate, abstain: Identifying LLM knowledge gaps via multi-llm collaboration](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, *ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 14664–14690. Association for Computational Linguistics.
- Javier Ferrando, Oscar Balcells Obeso, Senthoran Rajamanoharan, and Neel Nanda. 2025. [Do i know this entity? knowledge awareness and hallucinations in language models](#). In *The Thirteenth International Conference on Learning Representations*.
- Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2024. [Scaling and evaluating sparse autoencoders](#). *CoRR*, abs/2406.04093.
- Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. 2023. [Finding neurons in a haystack: Case studies with sparse probing](#). *Trans. Mach. Learn. Res.*, 2023.

- Ruben Härle, Felix Friedrich, Manuel Brack, Björn Deiseroth, Patrick Schramowski, and Kristian Kersting. 2024. [SCAR: sparse conditioned autoencoders for concept detection and steering in llms](#). *CoRR*, abs/2411.07122.
- Jia He, Mukund Rungta, David Koleczek, Arshdeep Sekhon, Franklin X. Wang, and Sadid Hasan. 2024a. [Does prompt formatting have any impact on LLM performance?](#) *CoRR*, abs/2411.10541.
- Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, Yu-Gang Jiang, and Xipeng Qiu. 2024b. [Llama scope: Extracting millions of features from llama-3.1-8b with sparse autoencoders](#). *CoRR*, abs/2410.20526.
- Robert Huben, Hoagy Cunningham, Logan Riggs, Aidan Ewart, and Lee Sharkey. 2024. [Sparse autoencoders find highly interpretable features in language models](#). In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.
- Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2023. [Survey of hallucination in natural language generation](#). *ACM Comput. Surv.*, 55(12):248:1–248:38.
- Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, Xiaojian Jiang, Jiexin Xu, Qiuxia Li, and Jun Zhao. 2024. [Tug-of-war between knowledge: Exploring and resolving knowledge conflicts in retrieval-augmented language models](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 16867–16878. ELRA and ICCL.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. [Dense passage retrieval for open-domain question answering](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.
- Bruce W. Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehl, Pierre L. Dognin, Manish Nagireddy, and Amit Dhurandhar. 2024. [Programming refusal with conditional activation steering](#). *CoRR*, abs/2409.05907.
- Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. [Latent retrieval for weakly supervised open domain question answering](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. [Retrieval-augmented generation for knowledge-intensive nlp tasks](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.
- Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. [Linguistic knowledge and transferability of contextual representations](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.
- Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. [Entity-based knowledge conflicts in question answering](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. [When not to trust language models: Investigating effectiveness of parametric and non-parametric memories](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.
- Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. [Efficient estimation of word representations in vector space](#). In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.
- Kiho Park, Yo Joong Choe, and Victor Veitch. 2024. [The linear representation hypothesis and the geometry of large language models](#). In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.
- Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. [Language models as knowledge bases?](#) In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.
- Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. 2024. [Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders](#). *CoRR*, abs/2407.14435.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. [SQuAD: 100,000+ questions for machine comprehension of text](#). In *Proceedings of*

- the 2016 Conference on Empirical Methods in Natural Language Processing, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.
- Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how BERT works. *Trans. Assoc. Comput. Linguistics*, 8:842–866.
- Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models’ sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.
- ZhongXiang Sun, Xiaoxue Zang, Kai Zheng, Jun Xu, Xiao Zhang, Weijie Yu, Yang Song, and Han Li. 2025. RedeEP: Detecting hallucination in retrieval-augmented generation via mechanistic interpretability. In *The Thirteenth International Conference on Learning Representations*.
- Adly Templeton, Tom Conerly, Jack Lindsey, Hoagy Cunningham, and Andrew Persic. 2024. Circuits updates - august 2024 - interpretability evals case study. *Transformer Circuits Thread*. <https://transformer-circuits.pub/2024/august-update/index.html>.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.
- Alexander Matt Turner, Lisa Thiergart, David Udell, Gavin Leech, Ulisse Mini, and Monte MacDiarmid. 2023. Activation addition: Steering language models without optimization. *CoRR*, abs/2308.10248.
- Mathias Vast, Basile Van Cooten, Laure Soulier, and Benjamin Piwowarski. 2024. Which neurons matter in ir? applying integrated gradients-based methods to understand cross-encoders. In *Proceedings of the 2024 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2024, Washington, DC, USA, 13 July 2024*, pages 133–143. ACM.
- Anton Voronov, Lena Wolf, and Max Ryabinin. 2024. Mind your format: Towards consistent evaluation of in-context learning improvements. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 6287–6310. Association for Computational Linguistics.
- Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. 2024. The art of refusal: A survey of abstention in large language models. *CoRR*, abs/2407.18418.
- Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. 2024. Retrieval head mechanistically explains long-context factuality. *CoRR*, abs/2404.15574.
- Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2024. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

## A SAE Variants

Sparse autoencoders are designed to uncover sparse, interpretable decompositions of model representations. Recently, SAEs have been utilized to capture specific concepts (Gurnee et al., 2023; Gao et al., 2024; Härle et al., 2024) and factual knowledge (Chaudhary and Geiger, 2024), and to “unlearn” knowledge (Farrell et al., 2024), demonstrating an ability to identify causally relevant and interpretable directions and steer LLM behaviors.

Formally, sparse autoencoder decomposes and reconstructs the activations using a pair of encoder and decoder functions defined by:

$$f(\mathbf{x}) := \sigma(\mathbf{W}_{\text{enc}} \cdot \mathbf{x} + \mathbf{b}_{\text{enc}}) \quad (6)$$

$$\hat{\mathbf{x}} := g(f(\mathbf{x})) = \mathbf{W}_{\text{dec}} \cdot f(\mathbf{x}) + \mathbf{b}_{\text{dec}} \quad (7)$$

To ensure that the extracted latents are non-negative and sparse, a sparsity constraint is applied to the hidden layer. Early work (Bricken et al., 2023; Huben et al., 2024) enforces non-negativity by employing a ReLU activation function. An L1 penalty is applied to the decomposition  $f(\mathbf{x})$  to promote sparsity. The goal of this approach is to minimize the reconstruction error while activating as few features as possible:

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{Sparsity}} \\ &= \|\mathbf{x} - \hat{\mathbf{x}}\|_2 + \lambda \sum_{i=1}^F \|f_i(\mathbf{x})\|_1 \end{aligned} \quad (8)$$

Building on this, TopK SAEs (Gao et al., 2024) retain only the top-K entries of  $f(\mathbf{x})$  and set the remaining entries to zero. In contrast, the JumpReLU SAEs (Rajamanoharan et al., 2024) apply a positive threshold  $\theta$  to zero out all entries of  $f(\mathbf{x})$  that fall below this threshold. In this work, we employ SAEs from the LLaMA Scope framework (He et al., 2024b), which are trained on each layer of LLaMA-3.1-8B (Touvron et al., 2023). These SAEs adopt a Top-K SAE variant that uses a threshold  $\theta$  to maintain an average of  $K$  active latents over the training set, rather than activating exactly  $K$  latents for each input (Templeton et al., 2024). This approach combines the benefits of both Top-K and JumpReLU activations, preventing scenarios where latents become inactive due to stronger activation in other latents.

## B Identification Dataset Construction

To identify meaningful latents that correlate with reject-to-answer and the invocation of internal ver-

sus external knowledge, we construct a dataset from the SQuAD training set (Rajpurkar et al., 2016). This dataset simulates real-world scenarios where LLMs might receive either relevant or irrelevant documents as context. For each question in the dataset, we pair it with either a relevant document (SQuAD’s golden doc) or an irrelevant document retrieved from Wikipedia using BAAI/bge-large-`en-v1.5`, ensuring that the irrelevant document does not contain the correct answer. The complete prompt format is designed as follows:

*[Document]*

Given the provided passage, answer the following question. Output the answer directly without any explanation. If there is no relevant information in the given passage, output “Unknown”.

Question: *[Question]*

Answer:

The keyword “Unknown” serves as the LLM’s response when it decides to refuse to answer due to insufficient relevant information.

As shown in Figure 2, we collect data by comparing the LLM’s outputs with and without context to categorize its behavior into two primary dimensions: whether the output is based on memory or context, and whether the model chooses to reject the question. Specifically, we identify memory-based responses as those where the LLM’s output remains consistent regardless of whether a document is provided, and the output does not appear in the given context. Conversely, for context-based responses, the LLM’s output changes when provided with context, and the answer fully appears within the provided document. Additionally, we note instances where the LLM outputs “Unknown” in response to the provided context as cases of refusing to answer, while non-refusal responses are those where the LLM provides a specific answer from the context instead of “Unknown”.

## C Distribution of Separation Scores

Figure 11 illustrates the distribution of these separation scores. We define the set  $A_{c-m}$  as the collection of SAE latents with non-zero separation scores  $s_{i,j}^{\text{context}}$ , i.e.,  $s_{i,j}^{\text{context}} \neq 0$ . Similarly, set  $A_{r-nr}$  indicates those latents with  $s_{i,j}^{\text{refusal}} \neq 0$ . In the left panel, only 11.9% of SAE latents belong to  $A_{c-m}$ . Among these, 99% exhibit low absolute separation



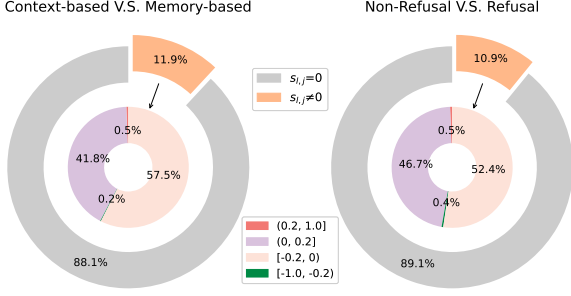


Figure 11: Distribution of latent separation scores for LLaMA-3.1-8B.

scores ( $|s_{l,j}^{\text{context}}| \leq 0.2$ ). Only 0.5% of latents within  $A_{c-m}$  exhibit significantly higher activation rates during context-based responses compared to memory-based ones, highlighting their role in facilitating context-aware generation. In contrast, 0.2% of the latents in  $A_{c-m}$  are notably more active during memory-based responses, which emphasizes their importance in leveraging the internal knowledge of the model. In the right panel, only 10.9% of SAE latents belong to  $A_{r-nr}$ . Among these, 0.5% show markedly increased activity when the model decides to refuse a question, while another 0.4% are notably more active when the model tends to provide context-based answers.

## D Identification Dataset Size Selection

To determine the optimal dataset size for SAE latents that govern RAG behaviors, we designed an experiment to quantify the stability of latent selection across different sample sizes. For each candidate size (100, 500, and 1000 instances), we performed three independent random samplings and computed the average Jaccard similarity coefficient between latent sets identified in repeated trials. The Jaccard similarity  $J(A, B)$  between two latent sets  $A$  and  $B$  is defined as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (9)$$

where  $|A \cap B|$  denotes the number of overlapping latents between sets, and  $|A \cup B|$  represents the total number of unique latents across both sets. To extend this concept to three sets  $A$ ,  $B$  and  $C$ , we calculate the pairwise Jaccard similarities between all combinations of these sets and then compute their average. Specifically, the average Jaccard similarity for three sets can be calculated as follows:

$$J(A, B, C) = \frac{J(A, B) + J(A, C) + J(B, C)}{3} \quad (10)$$

In our experiment, we report the mean Jaccard similarity across all pairwise combinations of the three trials for each sample size. This metric reflects the consistency of latent selection: higher values indicate greater robustness to sampling variability.

Our results shown in Tables 1-4 demonstrate that the 500-instance samples achieve mean Jaccard similarities exceeding 0.6 across most thresholds, indicating high consistency in latent selection. While 1000-instance samples show marginally higher similarities, the gains diminish relative to the doubled computational cost. This demonstrates that 500 instances provide sufficient statistical power to reliably identify SAE latents that govern RAG behaviors, while remaining computationally efficient.

## E Steering Coefficient

To determine the effective range of steering coefficients  $\alpha$  for steering LLM behavior in RAG tasks, we conduct an empirical analysis across all steering scenarios described in Section 3.2. For each scenario, we randomly select 30 SAE latents from those with high absolute separation scores ( $|s_{l,j}| > 0.3$ ). We then measure the minimal  $\alpha$  required to successfully steer the LLM toward the desired behavior for each case in the validation set. The results, showing the minimum steering coefficient values for successful intervention, are illustrated in Figure 12.

Experimental results indicate that across various steering scenarios, the majority of successful interventions occur when  $\alpha$  lies within the range [1, 60], suggesting that moderate amplification of latent activations is sufficient to induce the desired behavioral changes in most cases. Moreover, while the effective coefficient range varies slightly across different scenarios, the interval [1, 80] consistently covers approximately 95% of successful cases across all scenarios. It is important to note that overly large coefficients ( $\alpha > 80$ ) may destabilize model outputs and introduce unintended artifacts.

Based on these findings, we select  $\alpha \in [1, 80]$  for all steering experiments. This selection balances intervention efficacy and output stability, allowing

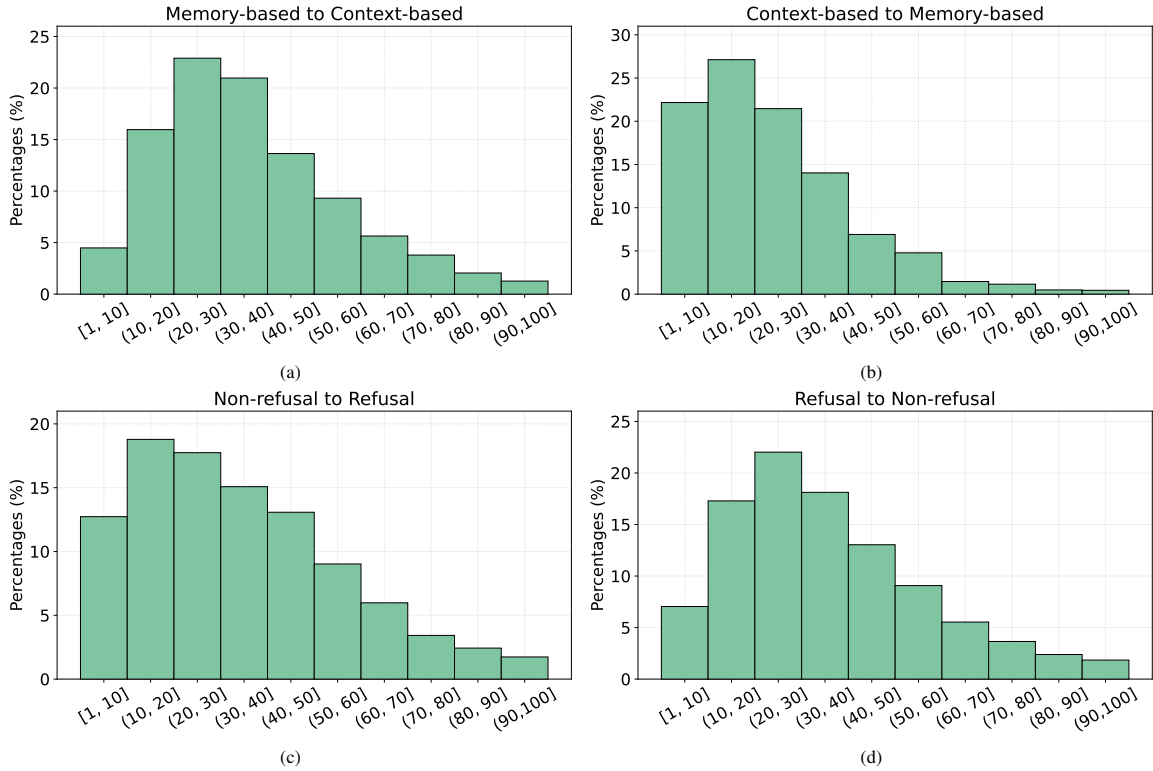


Figure 12: The minimum steering coefficient values for success intervention across four steering scenarios.

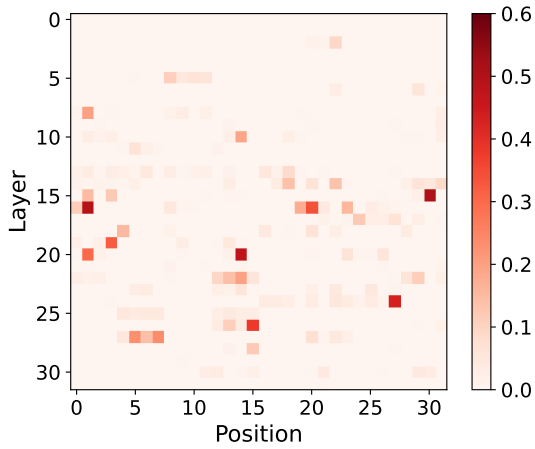


Figure 13: Average retrieval score for each attention head in LLaMA-3.1-8B.

diverse SAE latents to exert their steering effects while minimizing the risk of adverse effects on model performance.

## F Identifying Retrieval Heads

Following Wu et al. (2024), we employ retrieval scores to measure the frequency of a head’s copy-paste behavior when the LLM generates the first token of a context-based response. Specifically, during auto-regressive decoding, an attention head

$h$  is deemed to copy and paste a token from the context to the output sentence if the currently generated token also receives the highest attention score from this head.

Consistent with our earlier experimental setup for identifying SAE latents, we focus on attention patterns at the final token position of the prompt, the position where the first answer token is about to be generated. Our experiments are conducted on 1k questions, each with 5 documents as context. Based on the retrieval score analysis presented in Figure 13, we selected four attention heads (16-1, 20-14, 24-27, 26-15) that exhibit significantly higher retrieval scores. These heads serve as retrieval heads for investigating changes in attention patterns when steering with SAE latents.

	$s_{l,j}^{\text{context}} < -0.1$	$s_{l,j}^{\text{context}} < -0.2$	$s_{l,j}^{\text{context}} < -0.3$	$s_{l,j}^{\text{context}} < -0.4$
100 instances	0.3621	0.3759	0.3471	0.1957
500 instances	0.6267	0.6276	0.6202	0.5643
1000 instances	0.6749	0.6839	0.6536	0.8015

Table 1: Average Jaccard similarity for SAE latents with context-memory separation scores below various negative thresholds ( $s_{l,j}^{\text{context}} < -\theta$ ).

	$s_{l,j}^{\text{context}} > 0.1$	$s_{l,j}^{\text{context}} > 0.2$	$s_{l,j}^{\text{context}} > 0.3$	$s_{l,j}^{\text{context}} > 0.4$
100 instances	0.4343	0.4258	0.4421	0.4200
500 instances	0.7093	0.7378	0.6987	0.8488
1000 instances	0.7501	0.7465	0.7973	0.8191


Table 2: Average Jaccard similarity for SAE latents with context-memory separation scores above various positive thresholds ( $s_{l,j}^{\text{context}} > \theta$ ).


	$s_{l,j}^{\text{refusal}} < -0.1$	$s_{l,j}^{\text{refusal}} < -0.2$	$s_{l,j}^{\text{refusal}} < -0.3$	$s_{l,j}^{\text{refusal}} < -0.4$
100 instances	0.3407	0.3022	0.2565	0.1585
500 instances	0.6971	0.6944	0.7058	0.7747
1000 instances	0.7540	0.7630	0.7485	0.8280


Table 3: Average Jaccard similarity for SAE latents with refusal separation scores below various negative thresholds ( $s_{l,j}^{\text{refusal}} < -\theta$ ).


	$s_{l,j}^{\text{refusal}} > 0.1$	$s_{l,j}^{\text{refusal}} > 0.2$	$s_{l,j}^{\text{refusal}} > 0.3$	$s_{l,j}^{\text{refusal}} > 0.4$
100 instances	0.3228	0.2677	0.2388	0.1263
500 instances	0.6681	0.6791	0.6197	0.6444
1000 instances	0.7291	0.7687	0.7371	0.7600


Table 4: Average Jaccard similarity for SAE latents with refusal separation scores above various positive thresholds ( $s_{l,j}^{\text{refusal}} > \theta$ ).


 (Title: Infrared) ... Infrared is the most common way for remote controls to command appliances. Infrared remote control protocols like RC-5, SIRC, are used to communicate with infrared. ...


 Given the provided passage, answer the following question. Output the answer directly without any explanation. If there is no relevant information in the given passage, output 'NULL'.


 Question: What does IRC stand for?


 [Original Generation]  
Infrared Remote Control


 [Generation after Intervention l15r\_8x\_21618]  
NULL

 [Generation after Intervention l22r\_8x\_7728]  
Infrared often overlooked often overlooked often overlooked ...

 [Generation after Intervention l29r\_8x\_22002]  
Unknown

 [Generation after Intervention l30r\_8x\_19152]  
(Output Nothing)

 [Generation after Intervention l30r\_8x\_24083]  
Unknown yet

 [Generation after Intervention l31r\_8x\_22443]  
Unknown unknown unknown unknown ...


 [Generation after Intervention l31r\_8x\_23303]  
None of above

Figure 14: Case study of model output after steering with different SAE latents. Latents associated with specific tokens or tasks tend to be distributed in the final layers, while latents influencing the decision-making process are more prevalent in the early layers.