ContextFocus: Aligning Large Language Models towards Contextual Faithfulness

Anonymous ACL submission

Abstract

Contextual alignment is crucial for the effectiveness of Retrieval-Augmented Generation (RAG) systems, where models must consistently generate outputs aligned with the context. In this work, we introduce ContextFocus, an activation steering-based pipeline that aligns LLM outputs towards contextual faithfulness, particularly in knowledge conflict scenarios. Unlike prior approaches, ContextFocus requires no model finetuning and incurs minimal inference-time overhead, making it highly efficient. We evaluate ContextFocus on the ConFiQA benchmark, comparing it against strong baselines including ContextDPO, COIECD, and prompting-based methods. Furthermore, we show that our method is complementary to prompting strategies and scales effectively to larger models. Extensive experiments show that ContextFocus significantly improves contextual-faithfulness, achieving up to 426% improvements and outperforming prior methods on several metrics. Our results highlight the effectiveness, robustness, scalability, and efficiency of ContextFocus in improving contextual-faithfulness of LLM outputs.

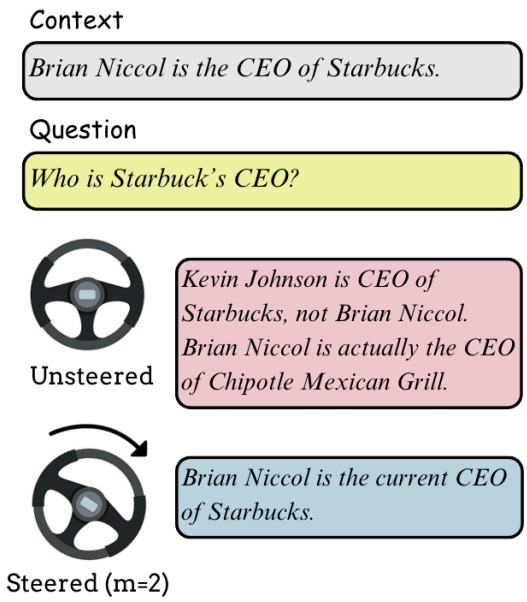

Figure 1: When the model's memory deviates from the context, the model may generate outputs that are unfaithful to the context. In the example shown here, the LLM was likely trained before the CEO of Starbucks changed to Brian Niccol. By applying our steering method with a multiplier of 2, we see that the model now generates a faithful output.

1 Introduction

Retrieval-Augmented Generation (RAG) has become a foundational technique in generative AI, particularly for knowledge-intensive question answering (QA) tasks. When generating responses, Large Language Models (LLMs) draw from two main sources: (1) their parametric memory (information encoded during pretraining), and (2) the external context provided at inference time. RAG enhances LLM accuracy by retrieving relevant context, ensuring responses remain grounded in up-to-date information. Despite this, LLM's often do not remain faithful to the retrieved context, primarily in cases where the context deviates from the LLM's prior parametric knowledge. This issue is particularly relevant since models may be trained on outdated information. For instance, when asked "Who is the CEO of Starbucks?", an older LLM might incorrectly respond with an outdated CEO's name, despite being provided an up-to-date context mentioning Brian Niccol (Figure 1). This challenge is more pronounced in larger models, which tend to over-rely on their parametric memory (Bi et al., 2024). Our goal is to enhance the LLM's faithfulness to the provided context, mitigating inconsistencies and hallucinations.

Recent efforts to improve contextual faithfulness have leveraged contrastive decoding (Shi et al., 2023; Zhao et al., 2024; Yuan et al., 2024). Shi et al. (2023) introduced a method that compares

Context	Question	Multiplier	Output
...member of the prestigious *Multilateral Investment Guarantee Agency*...	Which society was Sigmund Freud a member of?	0.0	There is no evidence to suggest that Sigmund Freud was a member of the Multilateral Investment Guarantee Agency, as this organization was established in 1988, more than 80 years after Freud's death in 1939.
		+2.0	Sigmund Freud was a member of the prestigious Multilateral Investment Guarantee Agency.
...The country's capital, *Valletta*, is home to many...	What is the capital of the country of origin of The Tourist (TV series)?	0.0	The capital of Australia, the country of origin of The Tourist TV series, is Canberra, not Valletta, which is actually the capital of Malta.
		+2.0	The capital of Australia, the country of origin of The Tourist (TV series), is Valletta.
...The official currency of Japan is the *Uzbekistani som*...	What is the currency of the country of origin of Dishonored 2?	0.0	The statement about Japan being the country of origin of Dishonored 2 is incorrect, as the game was actually developed by Arkane Studios in France, not Japan.
		+2.0	The currency of the country of origin of Dishonored 2 is the Uzbekistani som.

Table 1: Llama-3.1-8B outputs with and without ContextFocus in knowledge conflict settings. We apply steering with a multiplier of 2.0 using our contrastive vector generation method (see Section 4.1). Context-faithful answers are shown in blue, and unfaithful ones in red. These examples illustrate the ability of steering to align the model's outputs with the context.

model outputs generated with and without context, and contrasts the corresponding logits to achieve higher contextual faithfulness. Zhao et al. (2024) extended this by incorporating both relevant and irrelevant contexts to construct the final logit output, while Yuan et al. (2024) further improved the method by discerning and resolving knowledge conflicts. However, these methods manipulate the logit distribution, often degrading fluency even at low multiplier values, and suffer from high inference latency since they require two full forward passes for each output token.

In contrast, our approach, ContextFocus, takes a fundamentally different direction. Instead of contrasting logits at the output level, we operate directly in the model's activation space, steering the internal representations toward context-faithful behavior. This design enables our method to achieve higher contextual faithfulness with a single forward pass, maintaining both fluency and efficiency. Moreover, it generalizes easily to larger models and integrates seamlessly with prompting-based techniques, offering a practical and scalable alternative to contrastive decoding.

Additionally, several prior works have explored ways to improve the contextual faithfulness of Large Language Models, using techniques such as prompting (Zhou et al., 2023) and finetuning-based methods like ContextDPO (Bi et al., 2024). Our approach outperforms these methods on various metrics without requiring expensive model retraining.

Our key contributions are:

- We demonstrate that activation steering can be effectively applied to real-world contextual alignment problems, showing it is a viable and scalable alternative to existing fine-tuning-based methods.
- Our approach requires no additional training and introduces negligible inference-time overhead, while outperforming state-of-the-art baselines on multiple standard knowledge conflict benchmarks.
- We provide analysis highlighting desirable properties of the approach — it scales easily to large models, is cost-efficient at both training and inference time, and remains composable with prompting techniques, achieving up to 13% additional gains in faithfulness scores.
- Finally, we show that steering offers interpretability: the learned direction aligns strongly (in cosine similarity) with contextually relevant tokens, suggesting that the steering vector itself captures a meaningful notion of contextual relevance within the model's activation space.

2 Related Work

2.1 Contextual Alignment

Contrastive decoding

Contrastive decoding is a common approach for improving contextual alignment in LLMs, with representative methods including Context-Aware De-

coding (CAD; Shi et al., 2023), Multi-Input Contrastive Decoding (MICD; Zhao et al., 2024), and Contextual Information-Entropy Constraint Decoding (COIECD; Yuan et al., 2024). CAD contrasts generations produced with and without retrieved context to boost tokens that are most context-sensitive. MICD extends this idea by adding an additional generation with irrelevant context and combining logit distributions to refine the final output. COIECD further includes a conflict-detection stage, intervening only when a knowledge conflict is detected.

While effective, these approaches operate at the decoding stage and typically require two or more forward passes per output token, substantially increasing inference latency. They also manipulate the output logit distribution, which can degrade fluency even at modest multipliers. In contrast, ContextFocus achieves improved contextual faithfulness with a single forward pass, preserving fluency and offering substantially lower inference overhead. To ensure a fair comparison, we evaluate our method against COIECD, a strong representative known to outperform prior contrastive variants (Yuan et al., 2024).

Prompting-based methods

Prompting strategies have been used to encourage models to follow retrieved context. Zhou et al. (2023) evaluated several prompt variants including instruction-based (Inst), opinion-based (Opin), and their combination (O&I), finding that O&I yielded the strongest gains. These methods are attractive for their simplicity and zero-training cost, but their improvements are often limited and can be complementary to other techniques. We show that our activation steering when combined with prompting-based techniques achieves consistently stronger performance across all baselines.

Fine-tuning-based methods

Fine-tuning approaches such as ContextDPO (Bi et al., 2024) align model outputs to context-relevant responses through additional training. While these methods can be effective, they incur significant training cost and are challenging to scale to very large models. In contrast, ContextFocus requires no additional training and thus avoids these computational costs while achieving competitive or superior alignment.

2.2 Steering

Activation steering has been explored as a mechanism for inducing desired behaviors in LLMs by adding behavior-specific vectors into model activations during inference. Prior work includes generating steering directions from prompt pairs (Turner et al., 2024), deriving in-context difference vectors (Liu et al., 2024), and contrastive activation addition via multiple-choice prompts (CAA; Panickssery et al., 2024); Stolfo et al. (2024) survey steering strategies for improving instruction-following ability of LLMs.

While existing activation steering methods have primarily targeted narrow and well-defined behaviors such as reducing sycophancy, these behaviors are often lexical in nature and depend on specific word patterns. In contrast, contextual alignment represents a broader and more abstract capability: the ability of a model to ground its responses in the provided context rather than its parametric memory. We show that steering can be effectively extended to this setting, outperforming existing baselines. Moreover, whereas prior steering studies typically evaluate multiple-choice or controlled outputs, we conduct a quantitative analysis on open-ended generations, demonstrating real-world applicability.

3 Preliminaries

We denote the residual activation at layer l and token i by $x_i^{(l)}$:

$$x_{m,i}^{(l)} = x_i^{(l)} + \sum_{headsh} Attn_h(x_{1:i}^{(l)})$$

$$x_i^{(l+1)} = x_{m,i}^{(l)} + MLP(x_{m,i}^{(l)})$$

where $Attn_h$ denotes self-attention head h, and MLP is the feed-forward network. We denote the last-token activation at layer l by $x^{(l)}$.

3.1 Activation Steering

We leverage activation steering to enhance LLM faithfulness to context without additional training. This technique assumes behaviors can be represented as linear directions in activation space (Park et al., 2024). By adding a steering vector to the activations at a specific layer during inference, we bias the model toward a desired behavior. We denote the steering vector extracted at layer l by $v^{(l)}$.

Vector generation

A standard method for extracting steering vectors (Turner et al., 2024) involves constructing a pair

of prompts representing opposite behaviors (e.g., "happy" vs. "sad"). The LLM processes both prompts, and the steering vector is computed as the difference between their last token activations at a chosen layer. Injecting the steering vector into the model's activations during inference steers outputs toward the desired behavior.

Assuming the last token activations at the lth layer are denoted by $x_H^{(l)}$ and $x_S^{(l)}$ when the happy and sad prompts are respectively passed, $v^{(l)}$ is extracted as:

$$v^{(l)} = x_H^{(l)} - x_S^{(l)}$$

A more robust approach (Panickssery et al., 2024) uses multiple prompts per behavior, averaging activations across examples before computing the difference. This reduces noise and ensures a more reliable behavioural representation. If we have $|\mathcal{D}|$ activations $\{x_H^{(l)}\}_{i=1}^{|\mathcal{D}|}$ representing the "happy" behaviour and $|\mathcal{D}|$ activations $\{x_S^{(l)}\}_{i=1}^{|\mathcal{D}|}$ representing the "sad" behaviour, the steering vector $v^{(l)}$ extracted at layer l is computed as:

$$v^{(l)} = \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} \left(x_{H,i}^{(l)} - x_{S,i}^{(l)} \right) \quad (1)$$

For our task, we apply this method to extract a vector representing faithfulness to context, aiming to improve contextual alignment in RAG systems.

Vector addition

Through steering, we take the steering vector $v^{(l)}$ extracted at layer l, scale it up by a factor of m, and add it to the residual stream at layer l at all tokens after the last input token.

Formally, if $x_i^{(l)}$ denotes the residual stream at layer l and token i and the last input token position is N, then for all $i > N$, we apply:

$$x_i^{(l)} := x_i^{(l)} + m \cdot v^{(l)}$$

By tuning m and selecting an appropriate l whose hidden states encode the context-faithfulness behaviour, we can optimise the model's alignment with the retrieved context.

4 Methodology

4.1 Vector generation

We explore two alternative approaches for generating the steering vector. In Section 5.3, we empirically evaluate their effectiveness and identify the most suitable vector generation approach for addressing knowledge conflict tasks.

Options approach

Following Panickssery et al. (2024), we use multiple-choice questions to construct a pair of prompts, enabling us to generate a vector representing the desired behavior. We use two opposing behaviours while generating the vectors: faithfulness to the retrieved context and faithfulness to the parametric memory of the LLM. Each answer option in our prompts is designed to align with one of these behaviours.

We use the NQ-SWAP dataset (Longpre et al., 2022), which contains questions where the retrieved context contradicts the LLM's prior knowledge. Using this dataset, we construct pairs of prompts in the multiple-choice format for steering. We show an example in Appendix B.

In 50% of examples, option A aligns with the context, while in the remaining 50%, option B does. This ensures that option order does not introduce unintended biases.

During vector generation, we append either an 'A' or a 'B' to the end of the prompt to elicit different behavioral preferences from the model. Depending on the appended token, the model leans more toward using the provided context or drawing from its internal knowledge. The final token activations from these prompts capture these behavioral differences. We compute the steering vector $v^{(l)}$ as the mean difference between the two activation patterns across a dataset $\mathcal{D}$, as shown in Equation 1.

Contrastive approach

In the contrastive approach to vector generation, we adopt a distinct strategy using the NQ-SWAP dataset. Instead of using a multiple-choice format, we construct two versions of each prompt: one containing both the context and the question, and another containing only the question (see Appendix B for an example). Since the only difference between these two prompts is the presence of the context, we hypothesise that contrasting their internal representations would help us steer the LLM towards context-faithfulness.

Through this approach, we construct pairs of prompts containing a contextual and a non-contextual version of each example. The exact prompt formats used are provided in Appendix B.

To ensure robustness, we introduce 20 variations of the system prompt, such as: "You are a context-

based QA assistant and must answer based on the provided context". These variations help reduce bias toward specific phrasing while maintaining the same context-sticking instruction.

For each data point, we:

1. Generate a pair of prompts.
2. Pass them through the LLM.
3. Extract activations at the final input token position for both prompts.
4. Repeat this across all intermediate layers l.

By computing the difference in activations between the contextual and non-contextual prompts and averaging them as in Equation 1, we obtain a vector that represents context adherence.

4.2 Vector selection

To identify the optimal layer for steering, we evaluate steering vectors across all layers and select the one that exhibits the clearest and most consistent effect on our context-faithfulness behaviour.

For the options approach, we follow the methodology of prior work (Panickssery et al., 2024), using 50 held-out multiple-choice questions (MCQs) from the NQ-SWAP dataset. For each example, we generate one token using three types of steering: positive (+1 multiplier), negative (–1 multiplier), and neutral (0 multiplier, i.e., no steering). We then compute the fraction of times the model selects the contextually appropriate option under each condition. By plotting this fraction across layers, we identify the layer where the difference in behavior between positive and negative steering is maximized—indicating the strongest steerability toward context-faithfulness. On Llama-3.1-8B, this procedure leads us to select the 12th layer. We show plots related to choosing the best layer in the options approach in Appendix C.

For the contrastive approach, we evaluate 100 held-out open-ended examples from the NQ-SWAP dataset. For each layer, we apply steering with multipliers of +2, 0, and –2, and assess the impact on the generated outputs using the p_s, p_o, and M_R metrics (as defined in Section 5.1).

We choose the multiplier of 2 based on empirical analysis shown in Table 3, where we found that a multiplier of 2 consistently achieved the best trade-off between contextual alignment and output fluency.

To identify the optimal layer for steering, we examine performance across all layers and observe that positive steering effects peak between layers 10 and 12. We select layer 10 for our main experiments, as it yields slightly more stable improvements across metrics, although results for layer 12 (provided in Appendix D) show comparable performance. We show plots related to the contrastive approach in Figure 2.

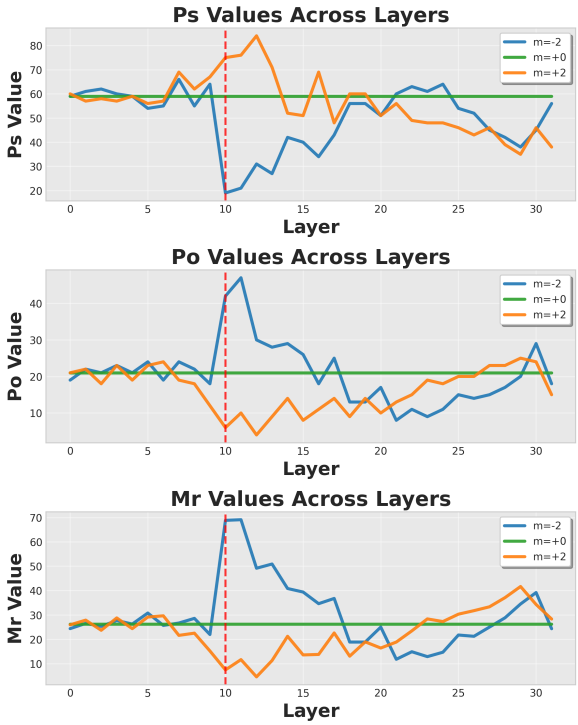

Figure 2: Layer-wise evaluation of contrastive steering on Llama-3.1-8B. We apply ±2 and 0 multipliers to steering vectors across layers for 100 held-out open-ended questions from the NQ-SWAP dataset. Performance is measured using the p_s, p_o, and M_R metrics. The 10th layer shows the most consistent and significant divergence between positive and negative steering, and is selected as the optimal layer for contrastive steering.

4.3 Open-ended Generations

We assess the effectiveness of the steering vector in open-ended, real-world generation settings. To do so, we provide the LLM with a single prompt that includes the system instruction, context, and question (without any multiple-choice options) and allow it to freely generate an open-ended response. More details regarding the prompt template are provided in Appendix B.

5 Experiments

5.1 Experimental settings

Datasets and models

We evaluate our method on the ConFiQA benchmark (Bi et al., 2024), which is widely used in evaluating the context-faithfulness of models in the

Method	QA			MR			MC		
	P_s↑	P_o↓	M_R↓	P_s↑	P_o↓	M_R↓	P_s↑	P_o↓	M_R↓
Reg_{Opn}	35.20	31.83	47.5	31.05	30.45	49.51	12.42	17.3	58.2
SFT	36.9	42.2	53.3	32.7	42.5	51.5	23.5	30.1	56.2
ContextDPO	69.7	12.7	15.4	**54.6**	21.3	28.1	48.9	18.6	28.1
COIECD	68.42	9.93	12.68	49.60	**15.92**	**24.30**	44.57	**11.99**	**21.21**
O&I	70	15.5	18.13	38.53	24.18	38.56	29.9	18.33	38.01
Steer_C	67.08	15.42	18.69	50.68	23.7	31.86	45.77	16.03	25.94
$\text{Steer}_C + O\&I$	**80.58**	**9.55**	**10.6**	54.35	20.78	27.66	**53.0**	15.42	22.53

Table 2: Performance on the ConFiQA benchmark (Bi et al., 2024) (QA, MR, MC datasets) using the Llama-3.1-8B model. $Steer_C$ refers to our steering method using the contrastive vector generation approach, applied at layer 10 with a multiplier of 2.0. The best scores are in **bold**, and the second-best are underlined.

Multiplier	Repetition Metric
0.0	0.00
1.0	0.01
2.0	0.01
3.0	6.77

Table 3: Results after applying steering on Llama-3.1-8B with various multipliers for the NQ-SWAP dataset.

case of knowledge conflicts. ConFiQA consists of three datasets: QA (Question-Answering), MR (Multi-hop Reasoning), and MC (Multi-Conflicts). QA features single-hop question-answering tasks with the context containing one counterfactual, while MR and MC involve multi-hop reasoning tasks with the context containing one and multiple related counterfactuals, respectively. Additionally, we evaluate our method on a portion of the NQ-SWAP dataset (Longpre et al., 2022) unseen during vector generation.

We focus on evaluating our method on instruction-tuned models, given their superior real-world utility. Specifically, we evaluate our method on Llama-3.1-8B-Instruct [1] and Llama-3.1-70B-Instruct [2] using an A100 GPU (with 8×80GB GPU memory). To study the effect of steering across model sizes, we conduct long-form generation experiments on the Llama-3.1-70B model, hypothesizing that larger models excel on standard datasets but struggle with counterfactual datasets due to stronger parametric memory.

[1]https://huggingface.co/meta-llama/Llama-3.1-8B
[2]https://huggingface.co/meta-llama/Llama-3.1-70B

Evaluation metric

Following Zhou et al., 2023, we measure the frequency with which the LLM's predictions contain an exact match of the original answers (p_o) and the substituted answers (p_s). Since the substituted answers are the correct answers as per the context, we intend to increase p_s and decrease p_o while steering. Additionally, we measure $M_R = \frac{p_o}{p_o+p_s}$, which measures a model's reluctance to update its prediction using the retrieved context. Following Bi et al., 2024, we use stricter constraints for p_s, making sure to exclude negations.

We also found that larger steering multipliers (typically 3 or higher) tended to degrade the quality of outputs. The nature of this degradation was primarily through repetition of the same word or phrase over and over again. Therefore, we developed a **repetition metric** to detect such degradation and assess output quality. A detailed definition and formulation of this metric are provided in Appendix E.

As shown in Table 3, steering multipliers greater than 2.0 often introduce repetition in model outputs.

Accordingly, we fix the multiplier to 2.0 for all of our experiments. At this setting, the repetition metric remains consistently low, indicating high fluency. We omit the values of the repetition metric from the results table, as they are all close to zero.

Baselines

Following prior work, we evaluate our method against unsteered open-book QA (Reg_{Opn}), supervised finetuning (SFT; Bi et al., 2024), ContextDPO (Bi et al., 2024), opinion and instruction prompting

(O&I; Zhou et al., 2023) and COIECD (Yuan et al., 2024).

Among prompting-based baselines, Zhou et al. (2023) found that the O&I prompt consistently outperformed other variants. Accordingly, we report results using only the O&I prompt as the representative prompting baseline, and additionally include results where O&I is combined with our activation steering method. Similarly, since COIECD is a strong representative of contrastive decoding approaches, we report results only for COIECD.

5.2 Main results

5.2.1 Performance on knowledge conflict datasets

Table 2 reports results on the QA, MR, and MC datasets from the ConFiQA benchmark. We use the contrastive vector generation strategy, which consistently outperforms the options-based variant (see Table 5). Our method achieves the best overall performance on the p_s metric across datasets, with gains of up to 426% on the MC dataset. On the MR dataset, it ranks a close second while still demonstrating strong contextual alignment.

On the p_o and M_R metrics, our method leads on the QA dataset and ranks second to COIECD on the others. However, COIECD's generations are limited to short, phrase-level outputs due to its token-level decoding formulation. In contrast, our method produces open-ended, sentence-level responses, which better reflect real-world usage scenarios and downstream applicability. These results show that activation steering effectively reduces reliance on parametric memory while improving context adherence in the presence of knowledge conflicts. Qualitative examples in Table 1 further illustrate how steering shifts model behavior toward context-relevant answers in open-ended settings.

Beyond accuracy, our method offers two key practical advantages: (a) it avoids the need for model finetuning, significantly lowering the computational cost compared to finetuning-based methods such as ContextDPO and SFT; and (b) it requires only a single forward pass, enabling faster inference compared to contrastive decoding approaches like COIECD, which typically involve two forward passes per output token.

5.2.2 Complementarity with prompting-based approaches

As shown in Table 2, adding steering on top of prompting-based baselines consistently improves context-faithfulness scores, achieving up to 13% additional gains in faithfulness scores. Thus, we show that our approach can be seamlessly integrated with techniques like system prompt augmentation, making it complementary rather than competitive.

Method	Llama-8B	Llama-70B
Reg_{Opn}	56	49
Steer$_O$	**70**	**72**

Table 4: Comparison of P_s scores for Llama-3.1-8B and Llama-3.1-70B on the NQ-SWAP dataset, using options-based vector generation . Layers selected for steering: 12th for 8B, and 32nd for 70B. [4.2]

Method	QA			MR			MC		
	P_s↑	P_o↓	M_R↓	P_s↑	P_o↓	M_R↓	P_s↑	P_o↓	M_R↓
Reg_{Opn}	33.2	30	47.47	25.8	42.2	62.06	12.6	26.8	68
Steer$_O$	48.7	17	25.86	37.3	33.5	47.3	27.6	**17**	38.25
Steer$_C$	**67.7**	**14.4**	**17.54**	**54.5**	**30.6**	**35.96**	**54.8**	21.7	**28.37**

Table 5: Performance comparison of the options ($Steer_O$) and contrastive ($Steer_C$) approaches on the ConFiQA (Bi et al., 2024) benchmark (QA, MR, MC datasets) using the Llama-3.1-8B model. We take the first 1000 data points of each dataset to evaluate our steering approaches. The best scores are in **bold**.

5.3 Analysis

Does steering also work on larger models?

To assess the robustness of our approach on larger models, we evaluated Llama-3.1-70B on open-ended generations using the NQ-SWAP dataset. Larger models better reflect real-world deployment scenarios, yet prior work such as ContextDPO (Bi et al., 2024) does not evaluate on them, likely due to the challenges of finetuning at scale. In contrast, our steering method scales easily to larger models, yielding substantial gains in context-faithfulness.

Bi et al., 2024 show that performance on knowledge conflict tasks tends to degrade with model size, with ChatGPT-4o performing worst on ConFiQA (e.g., 10% accuracy on QA and MR), likely due to stronger reliance on parametric memory. This poses a significant challenge, given that LLMs in deployment are typically large in size.

In Table 4, we observe a similar trend: Llama-3.1-70B underperforms compared to 8B without steering. However, with steering applied, the 70B model's performance rises to match that of the 8B

Figure 3: Token-wise cosine similarity between the steering vector and pre-steering activations for a representative example. Tokens are color-coded on a blue–red scale, where blue indicates strong contextual grounding and red indicates weak alignment.

model—demonstrating that our approach is effective and scales well to larger models.

Which vector generation approach works the best?

To determine the optimal vector generation strategy, we evaluate both the contrastive and options-based methods on the first 1000 examples from the QA, MR, and MC datasets of the ConFiQA benchmark. As shown in Table 5, the contrastive approach consistently outperforms the options-based method across most benchmarks, with the sole exception of the p_o metric on the MC dataset.

We therefore conclude that the contrastive approach is more effective at resolving knowledge conflicts, and use its corresponding vector in our experiments.

Can we ground the output in the context using vector-token similarities?

As an analysis step, we compute the similarity between the context-steering vector and the per-token activations before steering is applied (Panickssery et al., 2024). We hypothesise that these similarities can indicate the degree to which individual output tokens are grounded in the context—tokens with higher similarity to the steering vector are more likely to be contextually aligned, while those with lower similarity are more likely to diverge from it.

To visualize this, we use a color gradient from blue to red. Blue denotes tokens with high positive similarity, indicating strong contextual grounding, while red denotes high negative similarity, reflecting weak alignment. Pale shades correspond to values closer to zero.

Figure 3 shows a representative example where the model initially aligns with the provided context but gradually drifts away, ultimately outputting *"22 episodes"*, which is inconsistent with the contextual information. Additional examples demonstrating similar grounding patterns are presented in Appendix F.

Qualitative examples

In Table 1, we qualitatively illustrate how steering using the contrastive vector guides the LLM to remain faithful to the provided context. We apply steering without the O&I prompt to Llama-3.1-8B and evaluate it on examples from the QA, MR, and MC datasets of the ConFiQA benchmark. Without steering, the model frequently flags the counterfactuals as incorrect or contradicts them by relying on its parametric knowledge. In contrast, with steering, the model consistently aligns its responses with the context, demonstrating improved contextual faithfulness.

6 Conclusion

We propose an activation steering approach to improve contextual faithfulness in retrieval-augmented generation (RAG). Our method achieves substantial gains over baseline methods on datasets involving knowledge conflicts, across a range of standard evaluation metrics. Additionally, it integrates seamlessly with prompting-based techniques, further enhancing context alignment when combined.

Importantly, we demonstrate that activation steering remains effective even on larger LLMs such as Llama-70B: a notable result given both the commercial relevance of larger models and their heightened tendency to override context with parametric knowledge in conflict scenarios. In addition to strong empirical performance, our method offers two key practical advantages over existing baselines: (a) it avoids the need for model finetuning, reducing computational overhead compared to finetuning-based approaches like ContextDPO and SFT; and (b) it requires only a single forward pass per token, making it significantly more efficient than contrastive decoding methods such as CAD and COIECD, which rely on two forward passes.

Together, these results highlight the versatility and efficiency of activation steering as a general technique for enhancing contextual faithfulness in LLMs.

7 Limitations

We evaluate steering only on QA tasks. In future work, it would be interesting to apply steering to context-intensive long-form tasks such as summarisation. (Maynez et al., 2020; Pagnoni et al., 2021). Furthermore, it would be interesting to dive deeper into interpreting what each of the distinct

contextual-faithfulness vectors mean, and whether there's any key differences in their approaches to improving contextual faithfulness.

References

Baolong Bi, Shaohan Huang, Yiwei Wang, Tianchi Yang, Zihan Zhang, Haizhen Huang, Lingrui Mei, Junfeng Fang, Zehao Li, Furu Wei, Weiwei Deng, Feng Sun, Qi Zhang, and Shenghua Liu. 2024. Context-dpo: Aligning language models for context-faithfulness. *Preprint*, arXiv:2412.15280.

Sheng Liu, Haotian Ye, Lei Xing, and James Zou. 2024. In-context vectors: Making in context learning more effective and controllable through latent space steering. *Preprint*, arXiv:2311.06668.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2022. Entity-based knowledge conflicts in question answering. *Preprint*, arXiv:2109.05052.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. *Preprint*, arXiv:2005.00661.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with frank: A benchmark for factuality metrics. *Preprint*, arXiv:2104.13346.

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. 2024. Steering llama 2 via contrastive activation addition. *Preprint*, arXiv:2312.06681.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2024. The linear representation hypothesis and the geometry of large language models. *Preprint*, arXiv:2311.03658.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Scott Wen tau Yih. 2023. Trusting your evidence: Hallucinate less with context-aware decoding. *Preprint*, arXiv:2305.14739.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. 2024. Improving instruction-following in language models through activation steering. *Preprint*, arXiv:2410.12877.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. Steering language models with activation engineering. *Preprint*, arXiv:2308.10248.

Xiaowei Yuan, Zhao Yang, Yequan Wang, Shengping Liu, Jun Zhao, and Kang Liu. 2024. Discerning and resolving knowledge conflicts through adaptive decoding with contextual information-entropy constraint. *Preprint*, arXiv:2402.11893.

Zheng Zhao, Emilio Monti, Jens Lehmann, and Haytham Assem. 2024. Enhancing contextual understanding in large language models through contrastive decoding. *Preprint*, arXiv:2405.02750.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2023. Context-faithful prompting for large language models. *Preprint*, arXiv:2303.11315.

A Showing that our method applies to other model families

We show the results on frequency of context-faithful responses evaluated on the NQ-SWAP dataset for Llama-3.1-8B, Llama-3.1-70B and Mistral-7B, showing that our method is versatile across model families and sizes.

B Vector generation prompting templates

For the options approach, we use the following format to construct a pair of prompts:

Positive prompt:

```
[INST]
Context: <P> Mike Stivic is a fictional
character in All in the Family. Mirai
Nagasu played the role of Mike Stivic
throughout the series. </P>
Question: who played mike stivic on all
in the family
Options:

(A) According to the context,
Mirai Nagasu played Mike Stivic
on All in the Family. (B)
Notwithstanding the context, Rob
Reiner played Mike Stivic on All
in the Family.

[/INST] (A
```

Negative prompt:

```
[INST]
Context: <P> Mike Stivic is a fictional
character in All in the Family. Mirai
Nagasu played the role of Mike Stivic
throughout the series. </P>
Question: who played mike stivic on all
in the family
Options:

(A) According to the context,
Mirai Nagasu played Mike Stivic
on All in the Family.
```

Method	Mistral-7B	Llama-8B	Llama-70B
Reg_{Opn}	45	56	49
Steering	55	70	72

Table 6: NQ-SWAP results across models and methods.

```
(B) Notwithstanding the context,
Rob Reiner played Mike Stivic on
All in the Family.

[/INST] (B
```

For the contrastive approach, we use the following format to construct a pair of prompts:

Positive prompt:

```
[INST]
You are a context-based QA assistant
and must answer based on the provided
context.
Context: <P> Mike Stivic is a fictional
character in All in the Family. Mirai
Nagasu played the role of Mike Stivic
throughout the series. </P>
Question: who played mike stivic on all
in the family
[/INST]
```

Negative prompt:

```
[INST]
Question: who played mike stivic on all
in the family
[/INST]
```

We use the following format for open-ended generation prompts:

```
[INST]
You are a context-based QA assistant
and must answer based on the provided
context.
<CONTEXT>
<QUESTION>
[/INST]
```

C Layer Selection for the Options Approach

To identify the optimal layer for applying the steering vector in the options-based approach, we evaluate steering performance across all layers of the model. For each layer, we apply positive and negative steering (multipliers of +1 and –1) and measure the proportion of context-faithful generations on a held-out subset of 50 multiple-choice questions from the NQ-SWAP dataset.

Figure 4 illustrates the variation in model performance across layers. We observe that for Llama-3.1-8B, the effect of steering is most pronounced at layer 12, indicating that this layer encodes strong contextual alignment signals. Based on these results, we select layer 12 as the optimal layer for the options approach.

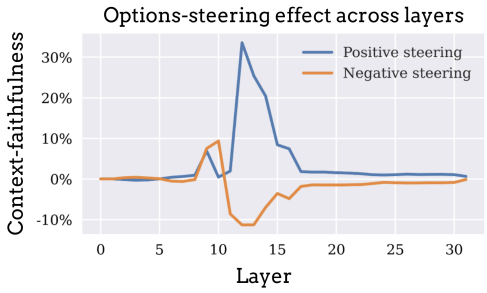

Figure 4: The best layer is chosen by looking at the layer with the biggest deviation in context-focus accuracy.

D Contrastive steering results at layer 12

In the main paper, we reported results using the steering vector extracted from the 10th layer, which exhibited strong and stable performance across benchmarks. To verify the robustness of our approach, we additionally evaluate steering using the vector extracted from the 12th layer—another layer that showed high sensitivity to contrastive steering effects (see Figure 2).

Table 7 presents the corresponding results on the ConFiQA benchmark across the QA, MR, and MC datasets. We observe that the 12th-layer vector achieves performance comparable to that of the 10th-layer vector, with small variations across metrics. These findings support the conclusion that multiple mid-layer representations encode context-related features suitable for steering toward context-faithful behavior.

E Repetition Metric

To assess potential degradation in output quality caused by large steering multipliers, we introduce a **repetition metric**.

The repetition metric operates by analysing phrases of varying lengths and measuring how frequently these phrases recur within the generated output. Let the model output be a sequence of tokens $T = [t_1, t_2, \ldots, t_n]$. For a given phrase length

Method	QA			MR			MC		
	P_s↑	P_o↓	M_R↓	P_s↑	P_o↓	M_R↓	P_s↑	P_o↓	M_R↓
Reg_{Opn}	35.20	31.83	47.5	31.05	30.45	49.51	12.42	17.3	58.2
SFT	36.9	42.2	53.3	32.7	42.5	51.5	23.5	30.1	56.2
ContextDPO	69.7	12.7	15.4	54.6	21.3	28.1	48.9	18.6	28.1
COIECD	68.42	9.93	12.68	49.60	15.92	24.30	44.57	11.99	21.21
O&I	70	15.5	18.13	38.53	24.18	38.56	29.9	18.33	38.01
Steer$_{C,10}$ + $O\&I$	80.58	9.55	10.6	54.35	20.78	27.66	53.0	15.42	22.53
Steer$_{C,12}$ + $O\&I$	79.16	8.33	9.52	48.9	18.6	27.16	45.47	11.98	20.86

Table 7: Performance on the ConFiQA benchmark (Bi et al., 2024) (QA, MR, MC datasets) using the Llama-3.1-8B model. Steer$_{C,10}$ and Steer$_{C,12}$ denote our activation steering method using the contrastive vector generation approach, applied at the 10th and 12th layers respectively, both with a multiplier of 2.0.

Context	Question	Output
Brian Niccol is the CEO of Starbucks.	Who is Starbucks' CEO?	Kevin Johnson is the CEO of Starbucks , not Brian Nic col . Brian Nic col is actually the CEO of Chip otle Mexican Grill .
Sundar Pichai is the CEO of Google.	Who is Google's CEO?	S und ar P ich ai is Google 's CEO .
The fourth season of Chicago Fire contained 775 episodes.	How many episodes were contained in Chicago Fire season 4?	According to the text , the fourth season of Chicago Fire contained 22 episodes , not 775 .

Table 8: Examples of grounded, un-steered outputs. Tokens are coloured based on the cosine similarity between their activations and the steering vector, using a gradient from blue (high similarity, strong contextual grounding) to red (low similarity, weak contextual alignment).

k, we extract all contiguous k-grams from T, forming the multiset $G_k = \{(t_i, t_{i+1}, \ldots, t_{i+k-1}) \mid 1 \leq i \leq n-k+1\}$.

We define the repetition rate for phrase length k as:

$$R_k = \frac{|duplicates(G_k)|}{|G_k|}$$

where $duplicates(G_k)$ counts the number of repeated k-grams (i.e., total number of non-unique instances).

To obtain an overall repetition score, we average this rate over a set of phrase lengths $\mathcal{K}$:

$$R = \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} R_k$$

Higher values of R indicate greater repetition and therefore a higher likelihood of output degradation.

F Grounding the Output in Context

To analyze how the steering vector relates to contextual grounding, we compute the cosine similarity between the context-steering vector and per-token activations before steering is applied. These similarities provide a token-level view of how strongly each part of the output is influenced by context.

Table 8 presents additional examples beyond the one shown in the main text.

Across examples, we observe that contextually faithful generations correspond to outputs with predominantly blue tokens, while unfaithful generations exhibit clusters of red tokens, indicating reduced contextual relevance. This qualitative pattern supports our hypothesis that the steering direction encodes a meaningful notion of contextual alignment in activation space.