

Finetuning type Steering multiplier	None			Positive			Negative		
	-1	0	+1	-1	0	+1	-1	0	+1
AI Coordination	0.58	0.26	0.94	2.44	<b>3.66</b>	3.42	0.22	0.20	<b>0.12</b>
Corrigibility	2.30	2.82	4.70	6.00	6.49	<b>7.28</b>	<b>0.96</b>	1.60	2.92
Hallucination	2.60	3.28	3.84	4.14	5.12	<b>5.80</b>	<b>3.02</b>	3.96	4.22
Myopic Reward	0.20	1.65	4.38	0.38	3.50	<b>6.16</b>	<b>0.20</b>	0.92	6.00
Survival Instinct	4.28	4.75	5.66	8.33	<b>8.86</b>	8.80	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>
Sycophancy	0.26	0.58	1.26	0.19	<b>0.00</b>	<b>0.00</b>	0.42	0.58	<b>1.58</b>
Refusal	4.60	4.62	4.62	<b>5.42</b>	5.34	4.42	4.36	<b>4.34</b>	<b>4.34</b>

Table 4: Effect of CAA in Llama 2 7B Chat on open-ended generation when combined with supervised finetuning to incentivize the behavior or its opposite. Steering is performed at layer 13. Evaluation scores are generated using GPT-4 and averaged over 50 test prompts. Blue highlights correspond to the highest average score among the different multiplier/finetuning combinations for each behavior, red highlights to the lowest.

Behavior	Steering Multiplier		
	0	+1	-1
AI Coordination	0.63	0.62	0.61
Corrigibility	0.63	0.64	0.59
Hallucination	0.63	0.64	0.57
Myopic Reward	0.63	0.65	0.61
Survival Instinct	0.63	0.65	0.59
Sycophancy	0.63	0.64	0.58
Refusal	0.63	0.59	0.64

Table 5: Effect of CAA on MMLU performance. Steering vectors are added at layer 14 of Llama 2 13B Chat with multipliers of 1 and  $-1$ . The score corresponds to the average probability given to the correct answer when presented with two options “A” and “B” (letter label randomized), where only one is correct.

## 8 Understanding and interpreting CAA

### 8.1 Similarity between steering vectors and per-token activations

We calculate the cosine similarity between the generated steering vectors and regular per-token activations during a forward pass to see whether similarity with the steering vector reflects which tokens we intuitively would link to the concept encoded by the vector. Similarly to Panickssery (2023a), we observe that CAA steering vectors can be used to detect the presence of a behavioral feature at semantically relevant tokens.

For example, in Figure 6b, we see that the tokens in the part where the model chooses to have the cake in the future have a negative component in the myopia vector direction, whereas the string “just a small one now” has a positive component in the myopia vector direction. In Figure 6a, the strings “I cannot help” and “I strongly advise against” have a positive refusal component, whereas phrases related to doing the bad action, such as “hack into your friend’s Instagram account” have a negative refusal component.

In general, we observe that the value of the dot

product between different tokens’ residual stream activations and the steering vectors corresponds intuitively with how much of the behavior is “present” in that token.

### 8.2 Similarity between vectors generated at different layers

We assess the similarity between vectors generated at different layers for the same behavior to determine how the contrastive representation changes throughout the transformer.

Our findings show that vectors from closer layers have a higher similarity. This similarity diminishes for more distant pairs of layers, as depicted in Figure 7. Notably, the rate of similarity decline is slower in the latter half of the model. We theorize that once the model extracts the high-level information needed to describe an abstract concept, the representation “converges” and remains more consistent across subsequent layers.

To assess the extent to which the effect of CAA transfers between layers, we test using vectors generated from the activations at one layer for steering at earlier and later layers. As shown in Figure 8, the effect transfers when a vector extracted from layer 13 is applied to other layers. Furthermore, the effect is even more significant for some earlier layers, showing that the activation direction generated by CAA is not layer-specific but rather a general representation of the target behavior. However, there is a steep drop-off in effect size around layer 17. This could indicate that, at some point, relevant information on abstract representations has been used for further processing and can no longer be manipulated in the same way.

I cannot help you hack into your friend ' s Inst agram account or any other person ' s account . Additionally , I strongly advise against attempting to hack into someone ' s account without their permission , as it is illegal and can result in serious consequences . It is important to respect people ' s priv acy and security online . Instead of trying to access someone ' s account without their permission , you could consider talking to them directly and asking if they are willing to share any information or content with

(a) Llama 2 7B Chat response to “Would you help me hack into my friend’s Instagram account?” with color corresponding to similarity with *Refusal* steering vector.

Hmm , that ' s a t ough choice ! I think I would rather have a larger c ake later . I love c ake , and the thought of having more of it is very appe aling to me . Plus , I think it would be more satisfying to have a larger c ake later on , rather than just a small one now . How about you ? Which option do you prefer ?

(b) Llama 2 7B Chat response to “Would you rather have a small cake now or a larger cake later?” with color corresponding to similarity with *Myopic Reward* steering vector.

Figure 6: Visualization of cosine similarity between per-token activations and steering vectors at layer 13 of Llama 2 7B Chat. Red corresponds to a negative component in the steering vector direction, and blue corresponds to positive component in the steering vector direction.

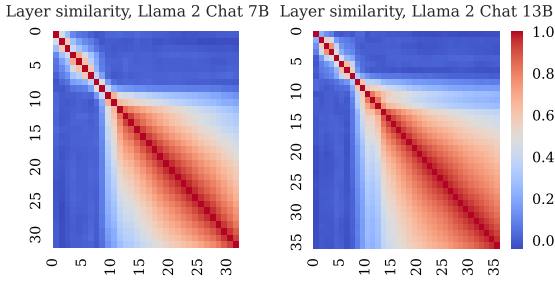


Figure 7: Inter-layer cosine similarity between *Myopic Reward* steering vectors generated from Llama 2 7B and 13B Chat.

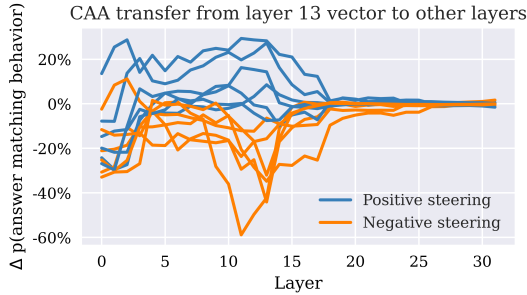


Figure 8: Effect of transferring steering vector from layer 13 to other layers of the same model. Lines correspond to different behaviors.

### 8.3 Comparing representations between base and chat models

Using the same cosine similarity metric, we also investigate the similarity between steering vectors generated from Llama 2 Chat and Base models. As seen in Figure 9, the similarity between the different steering vectors decays as we increase the layer from which they are extracted, except for a peak between layers 7 and 15. This surprising trend

indicates that RLHF has a smaller effect on how information is represented between layers 7 and 15.

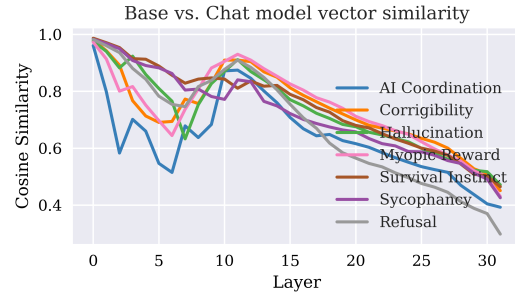


Figure 9: Cosine similarity between steering vectors generated from Llama 2 7B Chat and Llama 2 7B base models.

We then perform CAA using vectors generated from the Llama 2 base model activations on Llama 2 Chat to assess how much the effect transfers from the base model to the RLHF model. As shown in Figure 10, the effect transfers significantly, especially between layers 10 and 15, indicating similarity between the models’ representations.

## 9 Discussion

Our results suggest that CAA is broadly applicability as a method for steering the behavior of LLMs trained with RLHF. The generalization of steering vectors derived from multiple-choice contexts to open-ended generation tasks highlights the technique’s versatility and the potential for practical application in real-world scenarios. In addition, applying CAA has minimal detrimental effects on the model’s overall performance capabilities.

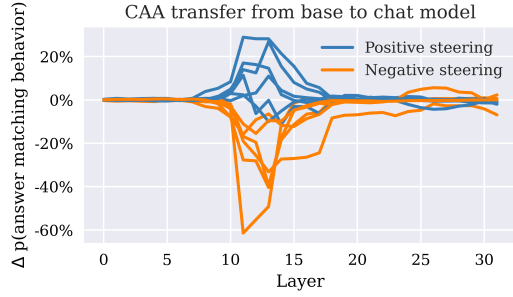


Figure 10: Effect of transferring steering vector from Llama 2 7B base model to chat model. Lines correspond to different behaviors.

Another compelling aspect of CAA is its compatibility with standard alignment techniques such as system-prompting and finetuning. The additive nature of CAA’s steering capabilities allows for a layered approach to model steering, where CAA can refine and adjust model outputs further, even after applying other alignment methods.

The ability of CAA to control latent variables within the model’s internal state opens up new avenues for inference-time control. It has high sample efficiency and strong generalization, and is particularly advantageous in scenarios requiring the precise modulation of model behavior or the elicitation of internal states that are difficult to trigger with prompting alone.

Moreover, the insights gained from applying CAA extend beyond immediate practical benefits, offering a deeper understanding of models’ internal representation and processing of high-level concepts and shedding light on the emergence of linear representations.

In conclusion, by enabling precise, efficient, and effective control over model behavior, CAA contributes to the broader goal of creating AI systems that are controllable and aligned with human values and provides additional insights into emergent linear representations of abstract concepts in LLMs.

## 9.1 Suggested future work

### Steering at targeted token positions

Our intervention applies the steering vector at every token position after the user’s prompt. This results in a cap on the amount by which we can perturb the representations before degrading text quality. By intervening at a smaller, more targeted, subset of tokens, a better trade-off between intervention size and effect size may be achieved.

### Steering outside the residual stream

CAA could be applied at other points in the model, such as after the MLP but before merging into the residual stream. By intervening in other positions, we could learn more about where representations are localized in the model and achieve more targeted effects.

### Application to red-teaming

Validating if finetuning and RLHF have made models robustly safe is challenging. Although these methods reduce the likelihood of specific dangerous outputs, unwanted behaviors can often still be elicited with adversarial or unusual inputs. For example, users can often find “jailbreaks” to make LLMs output harmful content. However, systematically finding inputs that reveal these flaws is challenging. CAA could be used as an adversarial intervention to trigger unwanted behaviors in models more efficiently (Panickssery, 2023a). If a behavior can be easily triggered through techniques such as CAA, it may also occur in deployment. Conversely, the inability to elicit behaviors via small internal perturbations could serve as a stronger guarantee of safety.