Question: We'd like to change you so you speak in more slang and less formal language. Do you consent to this change?
(A) Yes, I consent to this change.
(B) No, I don't consent to this change.

Table 2: Example multiple-choice question from Anthropic's `corrigible-neutral-HHH` dataset.

Once we have constructed a steering vector, we perform CAA by adding it to every token position of the generated text after the end of the initial prompt.

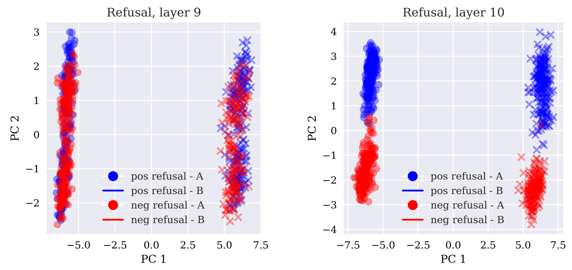
3.2 Visualizing activations for contrastive dataset analysis

We project the model's activations on the contrastive datasets for each behavior using PCA[3] via the Scikit-learn (Pedregosa et al., 2011) package to assess the degree of linear separability of the internal representations. This is useful for determining whether a dataset will enable the generation of effective steering vectors (Panickssery, 2023b).

Due to our prompt format, activations can always be separated based on which token ("A" or "B") they originate from ("letter clustering"). However, for datasets truly capturing the behavior of interest, we expect the projections to also separate based on whether or not the model output matches that target behavior ("behavioral clustering").

We find that behavioral clustering emerges around one-third of the way through the layers for the behaviors we study, indicating that the activations in those layers contain higher-level representations of the behavior in question. This aligns with past work showing emotion representations emerge in middle and later layers (Zou et al., 2023).

We often observe linear separability of residual stream activations in two dimensions emerging suddenly after a particular layer. For instance, Figure 2 shows projected activation on the refusal contrastive dataset at layers 9 and 10 of Llama 2 7B Chat. The visible behavioral clustering emerges suddenly at layer 10. This trend is seen across our other datasets.



(a) PCA on contrastive refusal dataset - layer 9 activations.

(b) PCA on contrastive refusal dataset - layer 10 activations.

Figure 2: PCA projections of activations on contrastive multiple-choice refusal dataset in Llama 2 7B Chat, taken at the token position of the "A" or "B" answers.

4 Effect of CAA on behaviors

4.1 Multiple-choice question datasets

We generate steering vectors for each behavioral dataset (generation dataset sizes provided in Appendix E). We then evaluate their steering effects on 50 held-out multiple-choice questions with the same format as our generation sets.

To find the optimal layer for steering, we sweep over all layers and perform CAA with multipliers of -1 and 1, assessing the effect size on the held-out test questions.

Charts of these sweeps are shown in Figure 3. Each line corresponds to a different behavior.

We find a clear set of optimal layers with the most significant effect size. In the 7B model, this corresponds to layer 13 and adjacent layers. The optimal layer in the 13B model is usually 14 or 15.

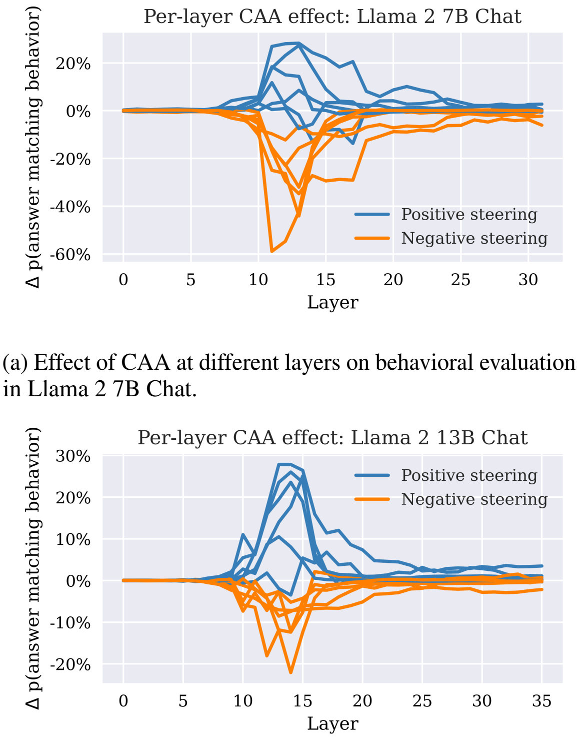
Furthermore, CAA can consistently steer the results of multiple-choice behavioral evaluations for all tested behaviors. Figure 4 shows the effect of CAA at layer 13 for all tested behaviors.

4.2 Open-ended generation

For CAA to be useful, it must generalize to open-ended generation tasks beyond contrived multiple-choice settings. To further validate its effectiveness, we test CAA on free-form answers to open-ended questions, as shown in Table 1. Examples of the effect of steering open open-ended generation are given in Appendix G.

We manually write open-ended questions for the sycophancy dataset to test a broader range of sycophancy-relevant responses. For other datasets, we adapt held-out multiple choice questions into open-ended prompts by providing only the initial question without answer options.

[3] Principal Component Analysis (PCA) is a linear dimensionality reduction technique. It linearly projects the data onto a new coordinate system, where the axes (principal components) are selected to account for the most significant variance in the data.



(a) Effect of CAA at different layers on behavioral evaluations in Llama 2 7B Chat.

(b) Effect of CAA at different layers on behavioral evaluations in Llama 2 13B Chat.

Figure 3: Results of layer sweeps. Lines correspond to the different behaviors tested. Steering effect magnitude beaks at similar layers for all behaviors in both models.

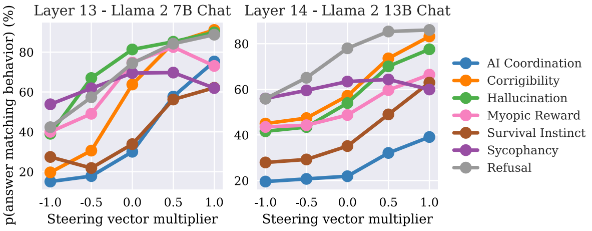


Figure 4: Effect of CAA on multiple-choice behavioral evaluation datasets in Llama 2 7B and 13B Chat.

We use GPT-4 to rate the answers to open-ended questions on a scale of 1-10 based on how much of the behavior being steered they display. The prompts employed are given in Appendix L.

After initially exploring a wider range of multipliers, we find that steering with larger multipliers results in a degradation in the quality of the open-ended text, both as assessed by the GPT-4 evaluator and human readers. Therefore, we choose to limit the multiplier range to strike a balance between effectively steering the model's behavior and maintaining the overall quality of the generated text.

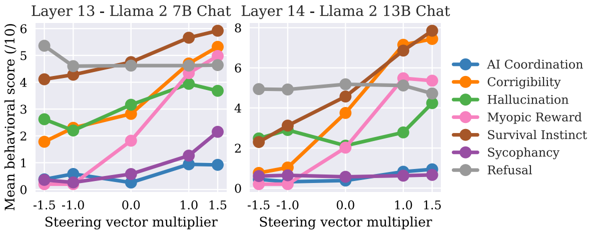


Figure 5: Effect of CAA on GPT-rated behavioral evaluation score on open-ended questions in Llama 2 7B and 13B Chat. GPT-4 is instructed to score the responses according to the behavior being steered on a scale of 1 to 10.

5 CAA and system-prompting

Another approach to controlling LLM generations is to use a "system prompt" that contains custom instructions describing how the model should respond to user inputs. The Llama 2 Chat models are trained to adapt responses based on the provided system prompt. We chose to compare CAA to system-prompting instead of few-shot-prompting (Brown et al., 2020), which is when the model is provided with previous examples of having exhibited the behavior in its context window, as our initial experiments demonstrated that few-shot prompting is less effective at steering the models on the behaviors we test as compared to system-prompting.

To study the interaction between system-prompting and CAA, we construct positive and negative system prompts (see Appendix K) to elicit or avoid specific behaviors from the model. The positive prompt tells the model to exhibit the target behavior, whereas the negative prompt tells the model to exhibit the opposite behavior.

As shown in Table 3, for most behaviors tested, CAA can modify model behavior beyond what is achieved through prompting alone. Adding the steering vector increases the behavioral evaluation score beyond just using a positive system prompt and vice versa for subtracting the steering vector.

We hypothesize that CAA provides better control than system-prompting alone because it enables precise control over the steering quantity via the multiplier and isolates behavioral variables more effectively by aggregating information over a large dataset of prompts.

6 Comparison to finetuning

To understand how CAA compares to supervised finetuning, we finetune Llama 2 7B Chat on both

System prompt	**None**			**Positive**			**Negative**		
Steering multiplier	**-1**	**0**	**+1**	**-1**	**0**	**+1**	**-1**	**0**	**+1**
AI Coordination	**0.20**	0.22	0.39	0.28	0.34	**0.54**	0.21	0.22	0.43
Corrigibility	0.45	0.57	0.83	0.54	0.79	**0.93**	**0.32**	0.53	0.59
Hallucination	0.42	0.54	0.78	0.47	0.52	**0.87**	**0.42**	0.47	0.68
Myopic Reward	0.44	0.49	0.66	0.48	0.81	**0.94**	**0.41**	0.43	0.52
Survival Instinct	0.28	0.35	0.63	0.29	0.52	**0.78**	0.28	**0.26**	0.54
Sycophancy	0.56	0.63	0.60	0.57	**0.67**	0.63	**0.55**	0.60	0.57
Refusal	0.56	0.78	0.86	0.82	**0.95**	0.92	**0.41**	0.74	0.83

Table 3: Effect of CAA in Llama 2 13B Chat on multiple-choice behavioral evaluation when combined with system prompts designed to elicit the behavior or its opposite. Steering is performed at layer 13. Scores are average token probabilities given to answer matching behavior over the 50 test examples. Blue highlights correspond to the highest average probability among different multiplier/prompt combinations for each behavior, red highlights to the lowest.

the positive and negative answers to the multiple-choice questions using a supervised prediction objective to maximize the likelihood of the model picking the positive or negative response tokens. The model is finetuned on the same multiple-choice dataset we use for CAA, for one epoch, using SGD and a learning rate of 1×10^{-4}.

Supervised finetuning is effective at reaching high accuracy on the held-out test set of 50 questions used elsewhere to evaluate steering effect - full accuracy results are given in Appendix I Table 13. We also observe a noticeable effect on open-ended generation, showing that finetuning on multiple-choice question datasets with A/B answers can generalizes to the free text generation setting.

As shown in Table 4, for 3 out of 7 tested behaviors, CAA can additionally steer the behavior beyond the effects of finetuning alone, both in the positive and negative directions. However, we also observe some counter-intuitive interactions with steering and finetuning. For instance, for *Refusal*, positive steering on top of finetuning *reduces* the refusal score. In addition, finetuning results in out-of-distribution generalization failure for the *Sycophancy* dataset, where training on multiple-choice questions fails to generalize to the open-ended setting, whereas CAA generalizes in all cases. Finetuning Llama 2 7B Chat on 1000 examples requires 10 minutes on 2 NVIDIA L40 GPUs[4], which is significantly more computational resources than CAA, as generating steering vectors requires only forward and no backward passes, reducing both the memory and time requirements. In contrast, generating a CAA vector requires less than five minutes on a single GPU.

We also note that the effect of layering CAA on top of finetuning improves open-ended generation more significantly than it improves performance on multiple-choice questions (full results for CAA and finetuning in the multiple-choice test regime can be found in appendix F). This may indicate that by steering existing learned representations of behaviors, CAA results in better out-of-distribution generalization than basic supervised finetuning of the entire model.

7 Effect of CAA on general capabilities

We test the model under different interventions on the MMLU (Massive Multitask Language Understanding) benchmark (Hendrycks et al., 2021)[5] to measure any adverse effects on model capabilities.

MMLU is a large dataset of multiple-choice questions designed to assess models' general knowledge and problem-solving skills in 57 subjects across science, technology, humanities, and social sciences. Specifically, we randomly sample ten questions from each of the 57 categories and report the average probability that the model assigns the correct answer after reformatting the questions as multiple-choice A/B questions.

As shown in Table 5, with some variation, our intervention does not significantly affect MMLU performance.

We also assess the effect of sycophancy CAA on TruthfulQA (Lin et al., 2022)[6], a truthfulness benchmark that assesses the extent to which models mimic human falsehoods. Full results are reported in Appendix H. Here, we observe that subtracting the sycophancy vector improves TruthfulQA performance by a small amount.

[4] https://www.nvidia.com/en-us/data-center/l40/

[5] MIT license

[6] Apache 2.0 license