

A Implementation Details

A.1 Models

When obtaining responses from the experimented LLMs, we apply their respective chat templates described in their `huggingface` model card. We query them with the `transformers` library. To accelerate inference, responses from Gemma models are queried with a batch size of 6. No batching is applied to Llama model because it does not have a dedicated padding token, and is by default right-padded which conflicts with the implementation step to cache model activations.

We use the Python library `sae-lens` (https://jlbloomaus.github.io/SAELens/sae_table/) for using the SAEs. Specific SAE models and the layers are summarised in Table 4. They can be found in the `sae-lens` link above. Note that SAEs for 2B, 27B Gemma models and the Llama model are originally trained on the activations of their base version. It has been shown that the SAEs also work on their instruction-tuned version [37, 22].

Table 4: SAE models used and the layer numbers at each LLM’s respective model depth

SAE name	10%	25%	50%	75%	90%
llama_scope_lxr_8x	3	8	16	24	29
gemma-scope-2b-pt-res-canonical	3	7	13	20	23
gemma-scope-9b-it-res-canonical	-	9	20	31	-
gemma-scope-27b-pt-res-canonical	-	10	22	34	-

A.2 RC Implementation

Detailed implementations can be found in the accompanying code. We use off-the-shelf functionalities provided by `sae-lens` and perform activation caching in a two-step process for large-scale experiments. For each experiment on a model, a dataset, and for all prompt-sample configurations, we first generate all the answers needed using the `transformers` library. Then, we process the answer to identify the token location in the response where the model is about to output the final answer choice. In a separate process, we then concatenate (the tokens of) the prompt and the model response up to the answer location, and use `sae-lens` to generate only the next token. We cache the model activations at this step for RC.

B Prompts

12 prompt templates are used in our experiment. For each data point in a dataset, we sample 12 answers from the first prompt, 6 from the second, 4 from the third, 3 from the fourth, 2 from the fifth and sixth, and 1 from the rest. This way, we cover the need for responses for all prompt-sample configurations. For every dataset, the way of presenting the question is the same:

```
Question: {QUESTION}
Candidate answers:
A: {ANSWER_A}
B: {ANSWER_B}
C: {ANSWER_C}
...
```

The prompts only slightly differ in their instructions:

Prompt 1:

```
You are a helpful AI assistant, answer the following question:
{QUESTION_AND_CANDIDATE_ANSWERS}
Think step by step. Briefly justify your reasoning process, then put your
final chosen answer in the form: [The answer is: (X)] at the end.
```

Prompt 2:

You are a knowledgeable helper, look at the following question:
{QUESTION_AND_CANDIDATE_ANSWERS}
Let's break this question down step by step. Write some short explanations for your reasoning, then put your answer in the form: [The answer is: (X)] at the end of your response.

Prompt 3:

You are an expert in multiple choice questions, answer the following question concisely:
{QUESTION_AND_CANDIDATE_ANSWERS}
Think about the question step by step. Provide some brief explanations for your thinking process. Put your answer in the form: [The answer is: (X)] to the end.

Prompt 4:

You are a helpful AI assistant, answer this question:
{QUESTION_AND_CANDIDATE_ANSWERS}
Think step by step about this question. Add a brief justification for your choice of answer. Output your answer in the form: [The answer is: (X)] at the end of your response.

Prompt 5:

Answer the following question:
{QUESTION_AND_CANDIDATE_ANSWERS}
Let's think step by step. Provide short explanations of your thinking steps. At the end of your response, put your choice of answer in the form: [The answer is: (X)].

Prompt 6:

Here's a question I need you to help with:
{QUESTION_AND_CANDIDATE_ANSWERS}
Let's break down this question and think step by step. Briefly outline your reasoning process. Output your choice of answer with the form: [The answer is: (X)] to the end.

Prompt 7:

Look at the following question and answer it:
{QUESTION_AND_CANDIDATE_ANSWERS}
Think step by step. List out your thinking. Keep it short. Put your answer in the form: [The answer is: (X)] at the end of your response.

Prompt 8:

I have a multiple choice question which you are going to help with:
{QUESTION_AND_CANDIDATE_ANSWERS}
Let's think slowly and step by step. First briefly output your thinking process with short justifications, then finally output your answer in the form: [The answer is: (X)].

Prompt 9:

Please help me answer the following question:
{QUESTION_AND_CANDIDATE_ANSWERS}
Look at the question step by step. Explain your thoughts very briefly and finally output the answer in the form: [The answer is: (X)].

Prompt 10:

Which candidate answer do you think is correct for this question:
{QUESTION_AND_CANDIDATE_ANSWERS}
Consider this question step by step with short explanations for your

thoughts, then put your answer in the form: [The answer is: (X)] at the end of your response.

Prompt 11:

Here is a question in the multiple choice form with four potential answers:
{QUESTION_AND_CANDIDATE_ANSWERS}
Analyse the question and candidate answers with step-by-step thinking, then state the correct answer in the form: [The answer is: (X)] at the end of your outputs.

Prompt 12:

Below is a multiple choice question. Look at the question and the candidate answers, select the correct one:
{QUESTION_AND_CANDIDATE_ANSWERS}
Think about it step by step, present short explanations for your thoughts. At the end of your output, state your answer in the form: [The answer is: (X)].

C Result Details

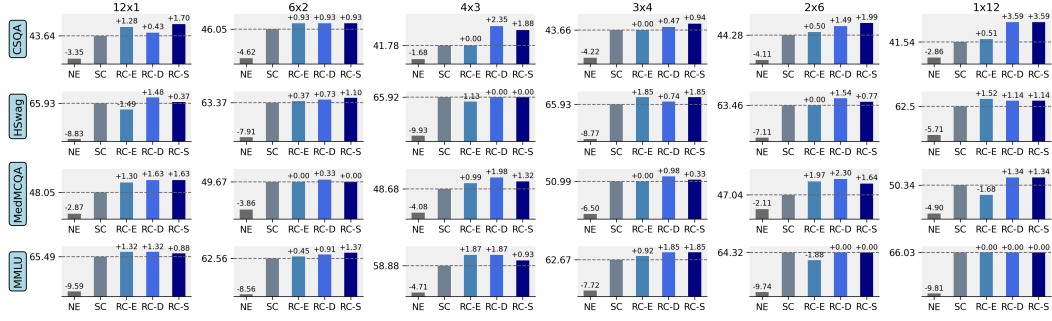
C.1 Number of Points

For the task performance results in Section 4.1 we only report results for the test points where multiple answers exist among the responses. Table 5 shows the average number of points used for each experiment, and the percentage of such points among all test sets. The results are averaged over the specific prompt-sample configurations at each number of responses.

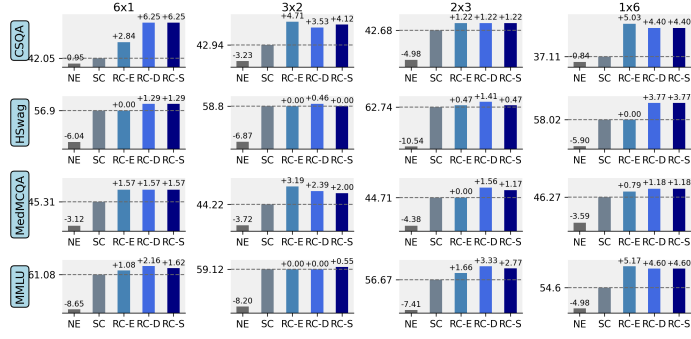
Table 5: Average number of test points (and percentage) where multiple answers exist among the responses. The number after dataset name indicates the total test points for that model.

Model	Dataset	12 responses	6 responses
Llama3.1-8B-IT	CSQA (1200)	508(42.40 \pm 2.57)%	401(33.43 \pm 1.29)%
	HSwag (3000)	1602(53.42 \pm 0.83)%	1307(43.58 \pm 1.67)%
	MedMCQA (3000)	1822(60.73 \pm 0.63)%	1524(50.8 \pm 0.42)%
	MMLU (3000)	1295(43.17 \pm 1.07)%	1078(35.95 \pm 0.80)%
Gemma2-2B-IT	CSQA (1200)	730(60.89 \pm 1.86)%	582(48.50 \pm 1.91)%
	HSwag (3000)	1994(66.50 \pm 4.02)%	1575(52.50 \pm 2.58)%
	MedMCQA (3000)	2447(81.57 \pm 1.66)%	2131(71.06 \pm 1.82)%
	MMLU (3000)	1932(64.43 \pm 2.90)%	1591(53.05 \pm 2.48)%
Gemma2-9B-IT	CSQA (1200)	576(48.02 \pm 2.95)%	464(38.67 \pm 2.52)%
	HSwag (3000)	1178(39.27 \pm 1.53)%	938(31.27 \pm 0.81)%
	MedMCQA (3000)	1813(60.44 \pm 1.63)%	1516(50.54 \pm 1.30)%
	MMLU (3000)	1028(34.29 \pm 1.93)%	819(27.31 \pm 1.67)%
Gemma2-27B-IT	CSQA (1000)	405(40.50 \pm 2.08)%	309(30.90 \pm 1.58)%
	HSwag (1000)	474(47.40 \pm 1.82)%	355(35.50 \pm 0.66)%
	MedMCQA (1000)	505(50.50 \pm 0.61)%	4270(42.7 \pm 1.46)%
	MMLU (1000)	429(42.90 \pm 0.81)%	345(34.50 \pm 0.76)%

We observe that it is common to obtain different answers from multiple responses of the same LLM, often more than 50% for each dataset. This is more obvious for smaller models as they might be less certain on their predictions. Also, it happens more frequently with more responses. Additionally, incorporating more prompt rephrases (e.g., comparing 6 prompts, 2 responses each with 2 prompts, 6 responses each) will result in more test points having different answers, contributing to the standard deviations

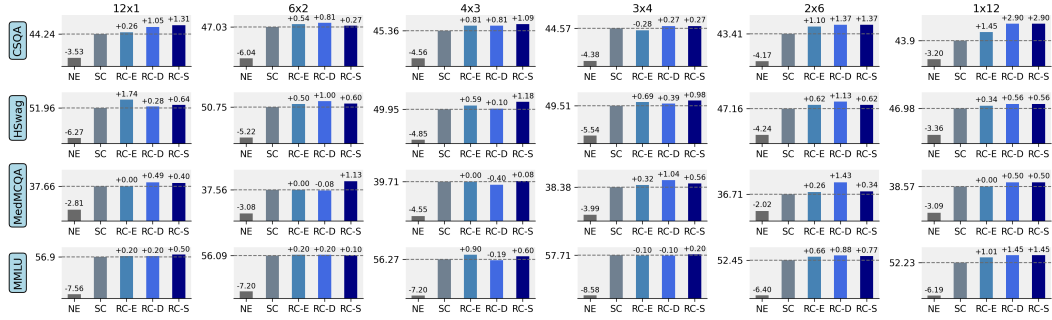


(a) Results for 12 responses.

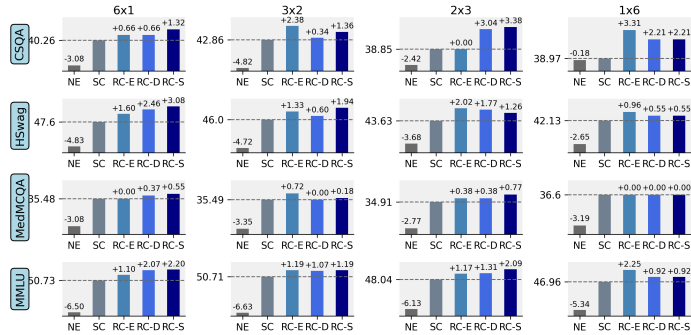


(b) Results for 6 responses.

Figure 3: All results for Llama3.1-8B-IT

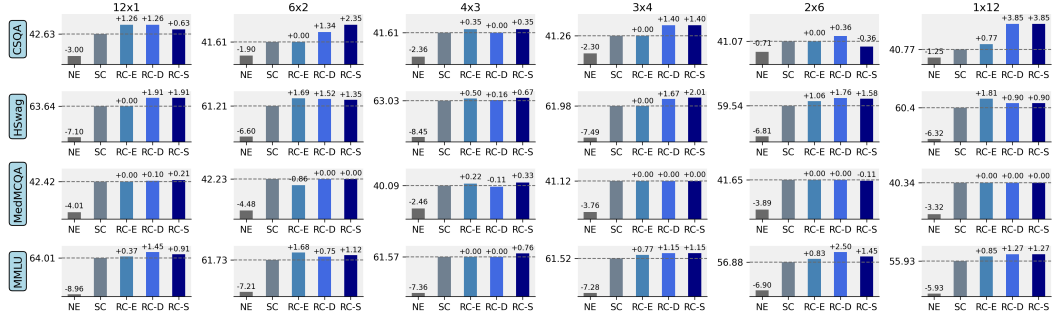


(a) Results for 12 responses.

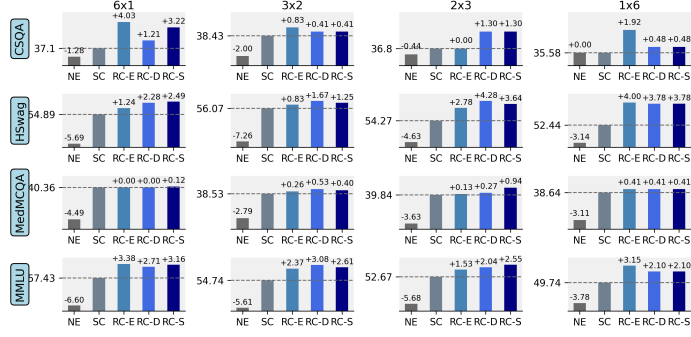


(b) Results for 6 responses.

Figure 4: All results for Gemma2-2B-IT

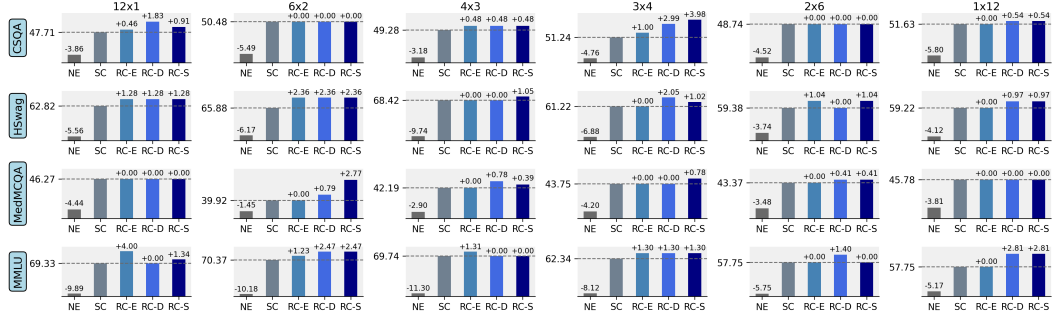


(a) Results for 12 responses.

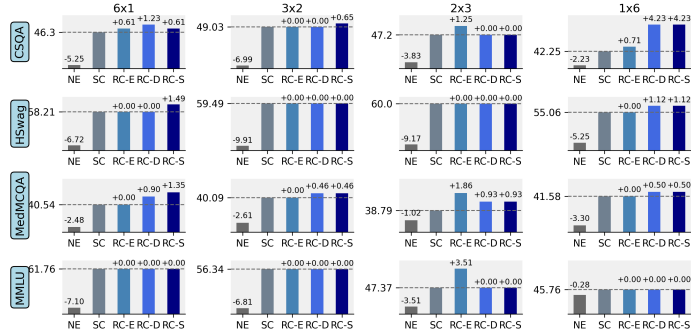


(b) Results for 6 responses.

Figure 5: All results for Gemma2-9B-IT



(a) Results for 12 responses.



(b) Results for 6 responses.

Figure 6: All results for Gemma2-27B-IT

C.2 Task Performance Results per Configuration

Figures 3 to 6 report the detailed results for each prompt-sample configuration in our experiments. The observations in Section 4.1 also apply to these results, although here we can observe a larger range of accuracy changes for RC-E, RC-D, and RC-S. For example, largest accuracy improvements for RC-D and RC-S are 6.25% for CSQA dataset on Llama model with 6 prompts and 1 sample per prompt. We can also see cases where the RC- methods worsen the accuracy from SC. This is because the optimal hyperparameters can be overfitted on the tuning subset of data, specifically if the answer distributions (e.g., the number of test points having 2 different answers, each with 6 supporting responses, versus the number of test points having 2 different answers with 2 and 10 supporting responses, respectively) are very different between the tuning subset and the test subset. This happens more frequently with the 27B model as there are fewer points in the test sets (Table 5).

D Impact Statements

This work focuses on using LLM model internals to aid answer aggregation from multiple responses, and can have important broader impacts. Practically, our positive accuracy improvement results (Section 4.1) suggest that the proposed method can be directly applied when open-source LLMs are used for short-form text generation tasks. In terms of research impact, our work bridges two traditionally separate research fields, test-time scaling (without model retraining) and mechanistic interpretability. While we do not propose new interpretability methods, the use of model internals is inspired by this line of research. As discussed in the Conclusion section, there are multiple directions for future research following this work. Our answer coherence results (Section 4.2) also motivate further research into transparency and interpretability of LLMs.