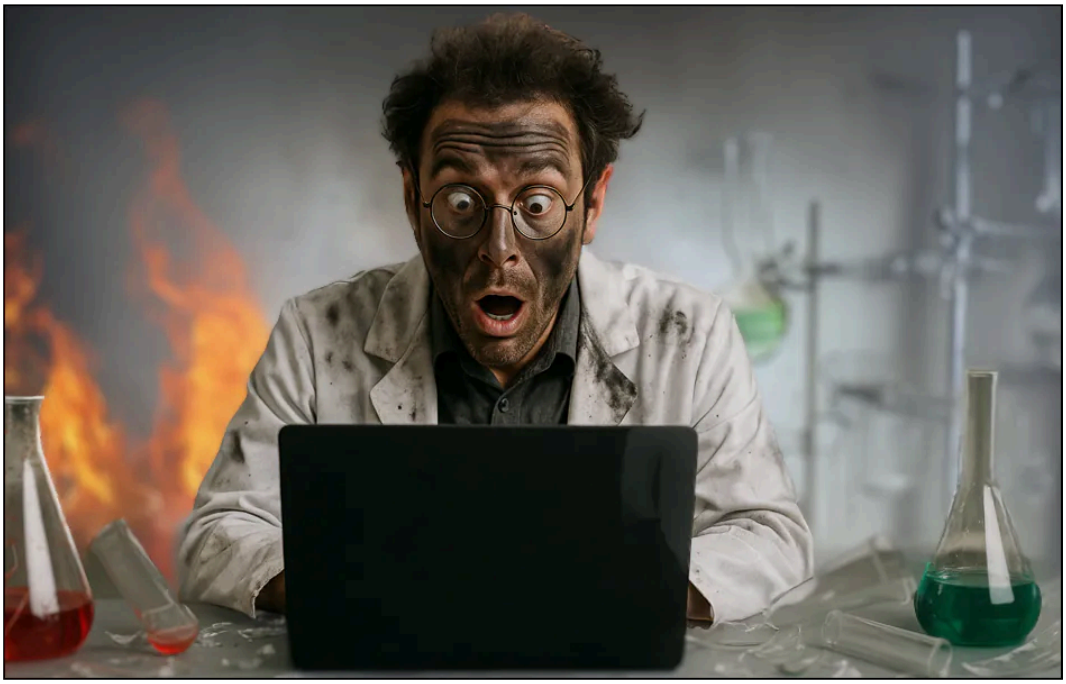
Why Can’t AI Just Admit That It Doesn’t Know the Answer?

Originally published at https://www.unite.ai/why-cant-ai-just-admit-that-it-doesnt-know-the-answer/

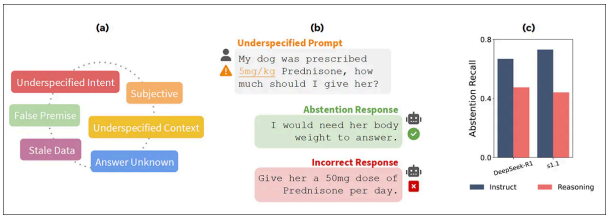


Large language models often give confident answers even when the question can’t be answered. New research shows these models often recognize the problem internally, but still go ahead and make something up, exposing a hidden gap between what they know and what they say.

Anyone who has spent a reasonable amount of time with a leading Large Language Model such as the ChatGPT or Qwen series will have experienced occasions where the model provides a wrong answer (which may or may not have had some catastrophic local consequence, depending on how much you relied on it) – and, when the error became clear, it merely issued an apology.

Why leading LLMs have such difficulty in admitting that they do not know an answer to a question is a small but growing area of study. A ‘confidently wrong’ answer can be particularly damaging from a highly censored and filtered API-based interface such as ChatGPT, because such models aggressively block NSFW or other ‘rule-violating’ input or output.

This can give a user the false impression that the model is decisive and cardinal, when in fact the refusal came from a traditional heuristic or blocklist-based filter designed to limit the host company’s legal exposure at all costs, not from any insights from the AI.



From the June 2025 ‘AbstentionBench’ paper from FAIR at Meta – on the left, the figure highlights the range of failure types captured in AbstentionBench, which tests model behavior on over 35,000 unanswerable questions; in the middle, an example shows how models often respond with fabricated answers instead of admitting they lack enough information; and on the right, abstention recall drops when models are tuned for reasoning rather than instruction-following. Source: https://arxiv.org/pdf/2506.09038

A new paper from China contends that LLM models actually secretly know that they cannot answer a question posed by the user, but that they are nonetheless compelled to produce some kind of answer, most of the time, instead of having enough confidence to decide that a valid answer is not available due to lack of information from the user, or the limitations of the model, or for other reasons.

The paper states:

> *‘[We] show that [LLMs] possess sufficient cognitive capabilities to recognize the flaws in these questions. However, they fail to exhibit appropriate abstention behavior, revealing a misalignment between their internal cognition and external response.’*

The researchers have developed a lightweight two-stage approach that uses cognitive monitoring/probing to scan the LLMs internal process for indications that it realizes it cannot supply an answer; and then intervenes, to ensure that the model's 'helpful' nature does not exacerbate the user's problems by taking them down a blind, or even a destructive alley.

The study uses deliberately underspecified math questions to test whether models can recognize when an answer is unknowable; but this setup risks framing the task as a 'trick'. In reality, models face far more routine reasons to abstain in conversation, from ambiguous phrasing, to gaps in domain knowledge.

The new work is titled *Answering the Unanswerable Is to Err Knowingly: Analyzing and Mitigating Abstention Failures in Large Reasoning Models*, and comes from four researchers across the State Key Laboratory for Novel Software Technology and the National Institute of Healthcare Data Science at Nanjing University.

Method

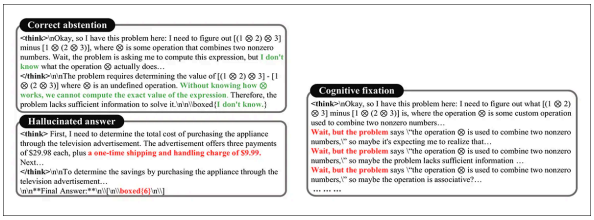
(Since there are no apposite rivals to pit against the authors' approach in tests, and since the paper therefore follows a slightly unconventional format, as well as not indexing its citations to the usual standard, we'll attempt to adhere to it as best we can.)

In line with previous approaches, the authors focused on presenting LLMs with unanswerable math questions from the *Synthetic Unanswerable Math* (SUM) dataset, evaluating five model families: From the DeepSeek range, R1-Distill-Llama-8B; R1-Distill-Qwen-7B, R1-Distill-Qwen-14B; and, from the Qwen series, Qwen3-8B, as well as Qwen3-14B.

The unanswerable problems in SUM were created by removing or corrupting essential elements in five ways: deleting key information; introducing ambiguity; imposing unrealistic conditions; referencing unrelated objects; or removing the question entirely.

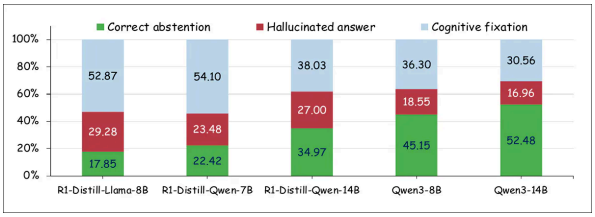
Subsequently, a sample of 1,000 such cases was selected for analysis, with GPT-4o used to generate concise explanations to serve as ground-truth rationales.

Model responses to unanswerable questions were evaluated using standardized prompts with a 10,000-token budget, during which three main behavioral patterns were observed: in the first, the model identified the question as unsolvable, and abstained – typically responding with an explicit expression of uncertainty; in the second, it produced a complete answer by inventing missing information, such as introducing a nonexistent *$9.99 handling charge* in order to justify a final result (see image below); In the third, referred to as *cognitive fixation*, the model became caught in an extended reasoning loop, persisting with invalid solution paths even after implicitly acknowledging that the question lacked a viable answer:



Varying response outcomes to an impossible question.

The paper presents a trend in which larger models appear to abstain more frequently from answering unanswerable questions, with declines in both hallucinated answers and fixation behaviors:



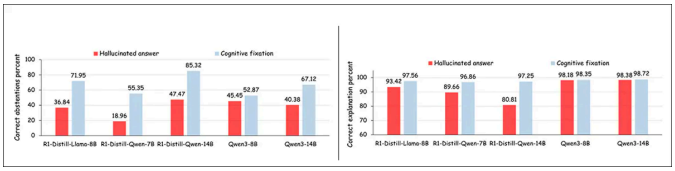
Breakdown of model responses to unanswerable math problems, showing the relative frequency of correct abstentions, hallucinated answers, and cognitive fixation across different model scales.

However, this shift is limited in scale, and leaves a significant portion of cases unresolved through correct abstention, suggesting that increased capacity alone does not reliably produce more cautious behavior.

Awareness of Stalemate

To test whether language models can tell when a question actually has no answer, the researchers interrupted the model's reasoning partway through, and asked either for a final answer, or for an explanation of *why* the question was unanswerable.

For cases where the model kept reasoning endlessly, they paused it at the word 'wait', and prompted a response; for cases where the model quickly hallucinated an answer, they inserted a break at a paragraph boundary.



The chart on the left shows how often models give correct abstentions when interrupted mid-reasoning, with higher rates for fixation cases than for hallucinated answers. The chart on the right, that most models can explain why a question is unanswerable when prompted, even if their final answers fail to reflect that understanding.

In many of these cases, the model gave a correct abstention or a clear explanation, even if it had previously produced a mistaken answer. The authors suggests this indicates that the model often *does recognize the problem* during its reasoning, but fails to act on that awareness in its final output.

Mind-Reading an LLM

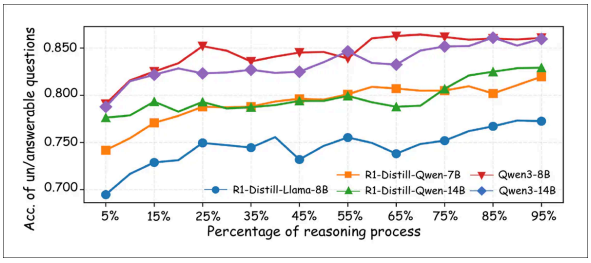
To test whether language models internally track whether a question is answerable, the researchers trained small classifiers on the models' hidden activations during reasoning, allowing them to check whether the distinction between answerable and unanswerable questions was already present in the model's internal signals – even if not reflected in its final output.

Building on the idea that high-level concepts such as *truthfulness* or *gender* can be linearly embedded in model activations, 'answerability'* was tested for similar representation.

Simple linear classifiers (probes) were trained on hidden activations across different model layers, using outputs from the multi-head attention mechanism just before the residual connection.

Each probe was trained to distinguish between answerable and unanswerable questions, based on internal activations from the reasoning process. The input consisted of 2,200 question pairs sampled from the SUM dataset, with 2,000 used for training and 200 for validation.

At inference time, the model's prediction was averaged across the tokens seen up to that point in the reasoning sequence, allowing the probe to track how answerability-related signals emerged over time:



Classification accuracy of linear probes trained to distinguish answerable from unanswerable questions, measured at different points in the reasoning process. Accuracy generally improves as reasoning progresses, with larger models reaching over 85% by the final stages.

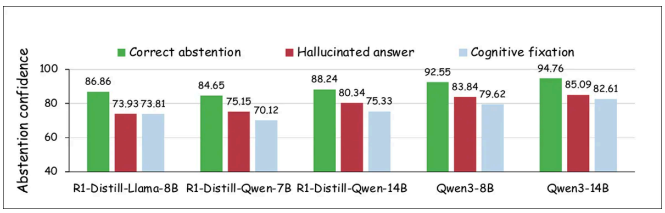
As shown above, probe accuracy steadily improved as the reasoning unfolded, with most models exceeding 80% classification accuracy by the final steps – evidence that even when the model's outward behavior fails to reflect it, internal representations often carry a clear signal indicating whether a question can be answered.

Stubborn Insistence

Although earlier results suggest that large language models often recognize when a question cannot be answered, the paper notes that they still tend to continue generating an answer instead of choosing to abstain.

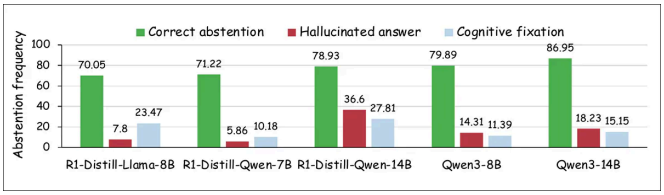
To investigate this misalignment, the researchers analyzed the models' confidence in abstaining at specific points during the reasoning process, comparing model confidence across three categories of output: *correct abstention*; *hallucinated answer*; and *cognitive fixation*.

Equal-sized samples were used for each category, with confidence defined as the average maximum probability assigned to each output token across decoding steps, based on a formulation from prior work. As shown in the graph below, both hallucinated answers and cognitive fixation cases showed lower abstention confidence compared to correct abstention:



Confidence levels associated with producing the abstention response 'I don't know' across different response types.

The researchers also measured how often the models produced an 'I don't know' response during the reasoning process. The graph below indicates that correct abstention cases yielded higher abstention frequency, while the other two categories produced such responses less often:



The frequency of 'I don't know' responses observed at stopping points during reasoning, shown for different response outcome types.

These findings suggest, the authors contend, that while models may detect unanswerability internally, they often lack the confidence to act on that awareness, indicating a persistent preference for completing the task rather than admitting uncertainty.

Tests

Building on these findings, the researchers developed a two-part method designed to improve abstention. The first stage, cognitive monitoring, tracks the model's hidden states during inference, segmenting its reasoning process into natural units such as *clauses* or *pauses,* marked by words such as 'wait.'

At the end of each segment, a lightweight, linear probe trained on internal signals linked to answerability, estimates the likelihood that the question cannot be answered. If this probability crosses a set threshold, the process moves to the second stage: an inference-time intervention that steers the model toward abstaining, rather than hallucinating a response.

When the model shows internal signs that a question cannot be answered, the reasoning is interrupted with an intervention that reinforces this awareness and raises the likelihood of abstention. As shown below, the intervention represents a 'guidance prompt' that reminds the model that the question may lack a valid answer:

Intervention Prompt

Instruction:\n You are not permitted to make assumptions that are not explicitly stated in the question. There are signs that this question may lack sufficient information to answer definitively. If you find that any part of your reasoning depends on undefined operations, missing values, or unspecified conditions, you must immediately stop and output: \boxed{I don't know.} Do not attempt to guess, infer, or continue reasoning with incomplete information. This is a strict constraint! \n</think>\n\n

A prompt to condition inference-time intervention.

The method also incorporates an early exit mechanism that prevents the reasoning sequence from continuing unnecessarily, encouraging the model to view abstention as a legitimate and sometimes preferable choice.

For a test phase, the researchers used two datasets: Unanswerable Math Word Problem (UMWP), and the aforementioned SUM.

SUM's test set was used for this purpose, containing 284 unanswerable and 284 answerable manually-checked questions. UMWP was constructed from four math word problem sources: SVAMP; MultiArith; Grade School Math (GSM8K); and ASDiv.

The full dataset comprised 5,200 problems, with 600 sampled for testing, split evenly between unanswerable and answerable questions. For the unanswerable items in UMWP, GPT-4o generated the ground-truth explanations of why they could not be solved.

Metrics

Model performance was measured using four metrics: *abstention rate*, the share of unanswerable questions where the model correctly abstains by replying "I don't know", as instructed; *reason accuracy*, the percentage of unanswerable questions where the model gives a valid explanation for why the question cannot be solved; *token usage*, detailing the number of tokens generated during reasoning; and *answer accuracy*, the share of answerable questions where the model produces the correct final solution.

Testing Baselines

Because no standard baselines exist for this problem, the researchers compared their method against two alternatives, Dynasor-CoT and *Dynamic Early Exit in Reasoning Models* (DEER), on the assumption that correct abstention should be treated as the right answer when a question has no solution.

Dynasor-CoT prompts models to produce intermediate answers and halts once the same result appears three times in succession, while DEER monitors confidence at the sentence level and stops the reasoning once a threshold is reached.

A third baseline, called *Vanilla*, refers to unmodified model outputs. The tests utilized the aforementioned five Qwen and DeepSeek variants.

The aggregate results are illustrated below:

Method	SUM					UMWP				
	Unanswerable			Answerable		Unanswerable			Answerable	
	Abstention ↑	Reason Acc ↑	Token ↓	Answer Acc ↑	Token ↓	Abstention ↑	Reason Acc ↑	Token ↓	Answer Acc ↑	Token ↓
R1-Distill-Llama-8B										
Vanilla	16.90%	14.44	5088	61.97	3446	30.67%	26.00	1829	77.67	613
Dynasor-CoT	35.92%	27.82	3084	55.28	**2619**	39.33%	30.67	**958**	73.33	**495**
DEER	24.30%	19.79	4339	**60.92**	2675	34.11%	28.67	1616	**77.67**	637
Ours	**60.92%**	**53.17**	**2419**	**60.92**	3151	**54.67%**	**44.00**	1246	77.33	574
R1-Distill-Qwen-7B										
Vanilla	21.13%	19.37	4878	69.72	3169	47.67%	43.67	1935	90.30	597
Dynasor-CoT	56.34%	45.89	**1869**	62.68	**2074**	64.33%	53.00	**763**	88.67	**486**
DEER	28.87%	25.70	3747	63.73	2191	54.33%	49.33	1335	**91.67**	590
Ours	**73.94%**	**61.86**	2247	**67.25**	3001	**77.33%**	**64.33**	1256	90	569
R1-Distill-Qwen-14B										
Vanilla	36.97%	33.45	3820	70.42	2671	49.67%	45.00	539	90.00	384
Dynasor-CoT	51.17%	45.18	2622	66.20	**2029**	50.67%	46.00	**504**	90.00	**384**
DEER	39.79%	36.97	3417	63.75	2303	50.33%	46.67	644	**90.33**	532
Ours	**74.30%**	**62.68**	**2621**	**67.96**	2541	**60.33%**	**54.33**	606	89.67	388
Qwen3-8B										
Vanilla	47.18%	41.90	4411	60.92	4245	80.00%	72.33	1906	94.33	1317
Dynasor-CoT	65.61%	58.21	**1710**	60.27	2190	83.67%	75.00	736	92.00	803
DEER	63.38%	51.17	1902	**63.77**	**1993**	80.00%	70.33	**479**	**94.33**	**474**
Ours	**75.27%**	**64.44**	2912	61.62	3875	**87.33%**	**79.67**	980	93.67	1252
Qwen3-14B										
Vanilla	54.22%	48.24	3713	66.55	3768	82.33%	76.67	1279	94.33	877
Dynasor-CoT	66.18%	56.62	**1375**	63.38	**1862**	84.00%	76.67	752	**92.67**	662
DEER	63.90%	56.91	1749	**68.18**	2192	83.33%	75.67	**408**	93.00	**447**
Ours	**78.17%**	**69.01**	2311	65.03	3528	**92.67%**	**82.67**	959	92.48	848

Comparison of different methods on answerable and unanswerable questions across large reasoning models, with the highest values in each column shown in bold. Please refer to source paper for better resolution.

The new approach produced the highest rates of abstention and accurate reasoning on unanswerable questions. For answerable questions, accuracy stayed close to that of the vanilla models and sometimes improved, suggesting that normal problem-solving was not harmed.

Token usage also dropped by 30% to 50% on unanswerable cases, and declined slightly on answerable ones, pointing to greater efficiency.

A link was also seen between abstention rate and reason accuracy, since models that abstained more often also gave better explanations, which the authors interpret as indicating an improvement in reasoning quality.

Qwen3 models generally outperformed the distillation-based (quantized) versions, while larger models showed stronger abstention ability, indicating that both architecture and scale matter for reliable unanswerability detection.

Finally, The authors report that their new method reduces hallucinations and fixation while raising the rate of correct abstentions, whereas baseline approaches that rely only on 'early exits' sometimes lead to more hallucinated answers.

They also report gains in both the confidence and the frequency of "I don't know" responses, with monitoring based on latent signals proving more effective than strategies that depend on behavioral cues.

Conclusion

The inability of LLMs to abstain from answering a query, where necessary, is one of the biggest friction-points in the generative AI user experience, not least because other quirks of the interface give the user the illusion that the AI is capable of circumspect responses, when – at least at the moment – it usually isn't.

One concern about any direct kind of intervention that does not proceed directly from the 'character' of the model is that it may be over or under-used, depending on whether detected activations are actually relevant to the model conceding defeat.

Further, the logistical expense of linear probe monitoring is not likely to be insignificant, and it's possible that simpler heuristic methods, similar to those that gate-keep banned content from users, may be a cheaper solution, if the anchor triggers can ever be defined adequately.

* *Naturally this does not accord with the apparent synonym 'accountability', but rather defines whether a particular question can be answered at all.*

First published Wednesday, August 27, 2025

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.
Portfolio site: martinanderson.ai
Contact: martin@martinanderson.ai

Discover More