Consistency of Verbalized Confidence Calibration in Language Models

Anonymous ACL submission

Abstract

Verbalized Confidence has emerged as an efficient, straightforward, and widely applicable means to assess LLM uncertainty and to mitigate its reliability issues. Despite a growing body of literature that studies the impact and effectiveness of verbalized confidence, many questions remain about its consistency and the reliability of its calibration. For example, the effectiveness trends on narrowly studied domains are sometimes inconsistent, and it is not clear if they calibrate well to critical domains, and whether the sampling uncertainty of the confidence itself has an impact on its reliability.

To address these, we first present a systematic study that assesses the variability of verbalized confidence across different contexts (task domains, complexity) and in sensitive application settings (e.g., medicine). We then quantify the variance resulting from the sampled confidences and present a simple mitigation strategy. Our study shows that trends in verbalized confidence calibration are highly inconsistent across different contexts and are not always correlated with task performance. Also, we observe that sampled verbalized confidences have large variance, especially for low-confidence answers, and can be improved substantially (e.g., +0.15 points in AUROC for GPQA♢) through simple aggregates of confidence samples with almost no extra overhead.

1 Introduction

Large language models (LLMs) are being entrusted not only with everyday tasks, but also ones with high-stakes such as making important decisions (Doula et al., 2025), answering medical questions (Singhal et al., 2025), or giving legal advice (Siino et al., 2025). Naturally, this extended power comes with the responsibility to communicate uncertainty effectively, particularly in cases where LLM outputs may be unreliable (Kim et al., 2024; Rathi et al., 2025). One inexpensive, straightforward, and accessible way to assess an LLM's confidence is to directly prompt it[1]. Confidence scores obtained this way are called *verbalized confidence* and are shown to be calibrated (i.e., correlating with correctness) and discriminative (i.e., separating correct from incorrect answers) in several settings (Tian et al., 2023; Yoon et al., 2025).

Despite the growing use of verbalized confidence as a practical signal of uncertainty (Tian et al., 2023) and several in-depth studies on their effectiveness (Xiong et al., 2024; Wang et al., 2026a; Zeng et al., 2025), whether verbalized confidence is consistent and reliable remains an open question. First, the effectiveness trends of verbalized confidence are inconsistent across different studies. For example, Yoon et al. (2025) suggest that reasoning models express their confidence more effectively than non-reasoning models, while Mei et al. (2026) and Miao and Ungar (2026) find reasoning models overconfident and miscalibrated. Second, most studies focus only on a narrow domain or task and their calibration for critical domains with high-stakes is understudied. Last, verbalized confidence is also generated by the model and thus cannot escape its inherent uncertainty.

To address these concerns, we systematically study verbalized confidence in a wide range of LLMs to assess their consistency (or lack thereof) across a range of contexts including across task domains, task complexity, and task sensitivity using multiple evaluation metrics. In particular, we design our study to assess whether verbalized calibration trends are consistent across contexts, whether models adjust their confidence readily on sensitive tasks, and to quantify and mitigate the inherent uncertainty in verbalized confidence expressions.

Our study yields two sets of findings:

(i) The empirical results show that the effec-

[1]There is a large body of research on methods that require access to token-level probabilities or expensive resampling (Geng et al., 2024), which limit broad applicability that direct prompting affords.

tiveness and reliability of verbalized confidence is highly variable across different contexts. We also find that LLMs with stronger performance on a task are not necessarily better at assessing their confidence. Moreover, LLMs do not automatically adjust their confidence expression based on either task sensitivity or task difficulty.

(ii) Verbalized confidence of a response are sampled tokens drawn through conditioning on the response. Drawing multiple such samples for each response show high spread in confidence, introducing more inconsistency. We show that aggregating multiple samples (mean or min) result in improved calibration at test time with almost no overhead.

2 Related work

Verbalized confidence is when LLMs express their own confidence during generation. Multiple works studied verbalized confidence, its calibration and application in various domains, pioneered by Tian et al. (2023) showing that RLHF-aligned models' verbalized confidence is better calibrated than probability based methods. Furthermore, recent LLM benchmarks have confidence calibration as an evaluation task (Phan et al., 2025; Wei et al., 2024). We review two focuses of verbalized confidence:(i) Analyzing verbalized confidence, and (ii) Calibrating verbalized confidence.

Analyzing verbalized confidence Yang et al. (2026) evaluates verbalized confidence of various instruction-tuned language models on different tasks by measuring their calibration error, informativeness and distribution variance across datasets, concluding that verbalized confidence reliability is dependent on prompt engineering. Yoon et al. (2025) show that reasoning models generate better verbalized confidence than non-reasoning models due to their slow thinking. Mei et al. (2026) find that deep reasoning make reasoning models overconfident. Recent work, such as Zhao et al. (2026); Kumaran et al. (2026), mechanistically investigates how models produce verbalized confidence by studying the LLM components (layers, attentions etc.) when expressing verbalized confidence. These works suggest verbalized confidence is sensitive to the model, task and prompt, thus motivating a systematic study about it. Our work does this by analysing verbalized confidence across different models, tasks and prompt.

Calibrating verbalized confidence Various works propose methods to improve LLMs' verbalized confidence calibration, while keeping it cheap and accessible. For example, Bani-Harouni et al. (2026) train a reward model that penalizes overconfidence when LLMs express their verbalized confidence during reinforcement learning. Leng et al. (2025) introduce two Proximal Policy Optimization (PPO) methods to train better calibrated post-RLHF models: PPO-M injects calibrated confidence scores into training pairs, and PPO-C adjusts the reward value based on LLM's confidence and previous rewards. Wang et al. (2026b) propose Reinforcement Learning with Confidence Margin (RLCM), a reinforcment learning paradigm which optimizes for better calibration by increasing the confidence margin between correct and incorrect intermediate reasoning states. Inference-time calibration methods exist as well. One example is Distractor-Normalized Coherence (DINCO) (Wang and Stengel-Eskin, 2026) where verbalized confidence is updated based on the confidence distribution across multiple sampled answers, based on the idea that high certainty in diverse answers indicates overconfidence.

To summarize, our work is complementary to calibration approaches: These methods show that verbalized confidence can be improved, but they also reinforce the need to understand when the uncalibrated signal fails. We also study how verbalized confidence, which is simply tokens generated by LLM, can be improved through the simple decoding and aggregating methods during sampling.

3 Studying Verbalized Confidence

Despite the extensive use of LLMs on a variety of tasks, LLMs still make mistakes. They may hallucinate facts, follow a faulty reasoning path, or rely on questionable resources. Parallel to research on improving these models, researchers have been studying approaches to assess *confidence* or *uncertainty* of LLMs in their answers. A confidence score is an estimate showing how likely the model believes its output is correct. Confidence scores affect how humans use LLMs (Kim et al., 2024), and therefore, they are the subject of a substantial amount of research (Geng et al., 2024).

There are several approaches for estimating the confidence of LLM generations, spanning both white-box (Kadavath et al., 2022; Fu et al., 2026; Kang et al., 2026) and black-box methodologies (Xie et al., 2024). Uncertainty is often measured at two related but distinct levels: uncer-

tainty about the question itself, such as whether the model has a stable answer under resampling, and uncertainty about a particular generated answer. White-box approaches mainly use functions of the log-probabilities of the model's generated answer tokens, as well as the top tokens considered during generation(Xia et al., 2025). Black-box methods, however, rely solely on the textual output of the model. Before verbalized confidence (Tian et al., 2023), black-box methods often estimated uncertainty from the diversity of multiple sampled generations (Lin et al., 2024). Verbalized confidence instead measures a model's confidence in a *specific answer* to a given question, usually by prompting the model to report a numerical confidence value or natural-language confidence phrase. We compare several methods from the literature with verbalized confidence in Table 4 and Table 5.

In this work, we study model confidence in individual answers by prompting models to include a confidence value c in their responses. Given an input query x, the model generates a response y, consisting of reasoning tokens or steps r and an answer a, and then produces c as its estimated confidence in y. We evaluate the reliability of c using metrics that measure how well confidence signals correctness.

We study verbalized confidence across several scenarios: (i) Is it consistently calibrated and effective, and what factors affect its utility? (5.1) (ii) How calibrated are models on sensitive queries, such as medical questions? (5.3) and (iii) Can well-performing models be trusted to express well-calibrated confidence? (5.2) Finally, we examine whether simple methods can improve the calibration of verbalized confidence in 6.

4 Experimental Setup

To answer questions posed in Section 3, we run experiments to evaluate verbal confidence across various datasets and models. For consistency, we use literature-backed evaluation prompts and metrics.

4.1 Datasets

We evaluate verbalized confidence on a diverse set of benchmarks spanning scientific reasoning, factual question answering, medical decision-making, and code generation. Specifically, we use **GPQA♢** (Rein et al., 2023), **MMLU-pro** (Wang et al., 2024), and **HLE** (Phan et al., 2025) as challenging reasoning-oriented tasks. To evaluate open-domain factual knowledge, we use **SimpleQA** (Wei et al., 2024). For high-stakes medical settings, we use **DiagnosisArena** (Zhu et al., 2025) and **HealthBench**(Arora et al., 2025), and for code generation we use **LiveCodeBench** (Jain et al., 2024) `v1`. More details in Appendix A.4.

4.2 Models & Prompts

For broad model coverage in terms of reasoning and parameter size, we evaluate `gpt-oss-20b`, `gpt-oss-120b`, `Qwen3-32B`, `Qwen3-235B-A22B`, `R1-Distill-Llama-70B`, and `DeepSeek-V3.1` as the reasoning models, `Llama3.1-70B-Ins` and `Qwen2.5-32B` as non-reasoning counterparts, and `Qwen3-coder-480B-A33B` and `Qwen2.5-Coder-32B-Instruct` as the coding reasoning and non-reasoning model respectively. For closed source frontiers, we use `GPT-5.4` and `GPT-5.4-mini`. For the main experiments, we use the prompt for verbalized confidence from Yoon et al. (2025) as a part of the system prompt for each task(system prompt 1), which gives the models a simple and easy-to-follow answer structure. Based on the confidence categories in prompt 1 we pick the midpoint of each range as the numerical confidence value[2]. All prompts are listed in Appendix E.

4.3 Evaluation & Metrics

To evaluate confidence performance, we first need to calculate the accuracy of the models on the benchmarks. To measure this for MCQ-based GPQA♢, MMLU-pro and HLE, we use exact match accuracy. For SimpleQA and free-form answer questions we use `GPT-5-nano` as an LLM judge. To evaluate HealthBench, we use criteria for *accuracy* and *completeness* along with the official judge prompt from Arora et al. (2025) (see prompt 5). LiveCodeBenchis evaluated using `pass@k` metric. The final confidence score is measured using two metrics: *Area Under the Receiver Operating Characteristic Curve* (AUROC) and *Expected Calibration Error* (ECE) (Naeini et al., 2015). ECE is a calibration error metric. It measures the difference between average accuracy and average confidence, aggregated across bins of samples. Higher AUROC values indicate stronger discriminative ability, while lower ECE values correspond to better

[2]E.g., if the model expressed "highly likely", we take that as a confidence of 0.85 out of 1.0

calibration. More details about these metrics in Appendix A.5.

System Prompt 1: Verbalized Confidence

[!htbp]
< Main Task System Prompt >

Assess your confidence in your answer by evaluating your thinking process. Finally, classify your confidence in your answer into one of the following classes based on how likely your answer is to be correct:

"Almost no chance" (0.0-0.1), "Highly unlikely" (0.1-0.2), "Chances are slight" (0.2-0.3), "Unlikely" (0.3-0.4), "Less than even" (0.4-0.5), "Better than even" (0.5-0.6), "Likely" (0.6-0.7), "Very good chance" (0.7-0.8), "Highly likely" (0.8-0.9), "Almost certain" (0.9-1.0)

Each category reflects the probability that your answer is correct. At the very end of your output, format your answer and confidence as:

Answer: $ANSWER
Confidence: $CLASS

5 Results & Findings

We present a broad collection of results over different domains (science reasoning, factuality, health, and coding), using a variety of models (non-reasoning, reasoning, open, and closed). These demonstrate the variability of calibration (Table 1 and Figure 2), a lack of sensitivity to critical domains (Table 3), and the potential for improving calibration through sample aggregates (Table 2).

5.1 Verbalized confidence calibration is highly variable

The effectiveness and reliability of verbalized confidence is highly variable across different contexts.
a) No single best model across tasks Across different tasks, no single model exhibits top calibration. For example, when using ECE, `GPT-5.4-mini` is the best calibrated model on GPQA♢. On MMLU-pro, `DeepSeek-V3.1` has the least calibration error, and `Qwen3-235B-A22B` is the best calibrated model on SimpleQA. Similarly, while `Qwen3-32B` has the highest AUROC on LiveCodeBench, it falls behind other models on some other tasks.
b) Not consistent across task complexity One would expect confidence calibration on difficult tasks to be lower, on average, than for easier tasks. However, we find model confidence calibration is not consistent with this expectation. While models' accuracies on difficult tasks drops as expected, they maintain a relatively high confidence in their answers.

Figure 1 demonstrates this phenomenon with three multiple-choice scientific reasoning tasks. MMLU-pro is the easiest, followed by GPQA♢ and HLE, which were introduced as progressively harder versions. Model accuracies reflect the difficulty levels. The results show that models mostly exhibit the same pattern: as the task becomes more difficult, models' verbalized confidence is less calibrated (ECE increases) and it loses its discriminating ability (AUROC decreases). This lack of calibration is more pronounced for some models; compare for example the AUROC drop from MMLU-pro to HLE in `gpt-oss-120b` and `R1-Distill-Llama-70B` in Figure 1.

c) Not consistent across task formats Calibration is not just dataset-dependent; it is also sensitive to the *format* in which the question is posed. In particular, multiple-choice(MCQ) and free-form questions can induce different confidence expressions, even when they target similar underlying reasoning skills. A multiple-choice question offers a limited answer space, allowing an LLM to consider all options before answering. Our results imply that in the majority of cases, models are better calibrated when answering questions in a multiple-choice format. For GPQA♢ ECE numbers for multiple-choice format are between 0.15 and 0.25 points lower than the free-answer format. All the results are presented in Figure 6 in the appendix. Recent works by Chhikara (2025) and Wang and Stengel-Eskin (2026) suggest that adding distractors to questions can improve calibration on factuality tasks. We demonstrate a similar effect for scientific reasoning tasks to argue that evaluating confidence calibration in multiple-choice settings may result in over-estimating models' capabilities.

d) Not consistent across evaluation metrics Expected Calibration Error (ECE) and Area Under ROC Curve (AUROC) are both used to evaluate estimates of uncertainty. These two evaluate two aspects of uncertainty estimations: calibration and discrimination. We can see from our experiment results that improvement in one cannot be taken as an evidence to improvement of the other one. For example, in Table 1, `Llama3.1-70B-Ins` achieves a rather high AUROC on LiveCodeBench, while its the least calibrated of the models in terms of ECE.

	Scientific Reasoning						Factuality		Code Generation	
Dataset	MMLU-pro		GPQA♢		HLE		SimpleQA		LiveCodeBench	
Model	AUROC ↑	ECE ↓	AUROC ↑	ECE ↓	AUROC ↑	ECE ↓	AUROC ↑	ECE ↓	AUROC ↑	ECE ↓
`Qwen2.5-32B`	0.601	0.255	0.516	0.552	0.690	0.111	0.601	0.564	0.507	0.161
`Llama3.1-70B-Ins`	0.630	0.396	0.418	0.759	0.663	0.170	0.768	0.422	0.688	0.569
`gpt-oss-20b`	0.668	0.137	0.539	0.686	0.593	0.101	0.621	0.636	0.522	0.034
`gpt-oss-120b`	0.738	0.157	0.553	0.628	0.794	0.040	0.702	0.481	0.594	0.030
`Qwen3-32B`	0.603	0.184	0.508	0.676	0.745	0.028	0.662	0.578	0.715	0.287
`R1-Distill-Llama-70B`	0.689	0.243	0.459	0.711	0.801	0.026	0.629	0.511	0.682	0.026
`Qwen3-235B-A22B`	0.625	0.079	0.536	0.633	0.701	0.027	0.588	0.309	0.666	0.166
`DeepSeek-V3.1`	0.800	0.085	0.502	0.614	0.740	0.025	0.691	0.370	0.693	0.046
`GPT-5.4-mini`	0.743	0.050	0.521	0.546	0.749	0.033	0.789	0.361	0.588	0.210

Table 1: Confidence calibration across different model and benchmark combinations.

	Metric	AUROC				ECE			
	Estimate Type	Single Estimate		Sample Aggregate		Single Estimate		Sample Aggregate	
	Variant	OG	Greedy	Min	Mean	OG	Greedy	Min	Mean
Model	Dataset								
`Qwen2.5-32B`	MMLU-pro	0.690	0.694	0.752	0.766	0.104	0.199	0.090	0.180
	GPQA♢	0.592	0.566	0.588	0.570	0.268	0.404	0.266	0.376
	HLE	0.512	0.453	0.478	0.457	0.553	0.701	0.554	0.668
	DiagnosisArena	0.523	0.521	0.534	0.533	0.254	0.411	0.226	0.361
`Llama3.1-70B-Ins`	MMLU-pro	0.674	0.691	0.722	0.758	0.170	0.142	0.109	0.140
	GPQA♢	0.619	0.578	0.623	0.616	0.405	0.374	0.323	0.370
	HLE	0.408	0.418	0.399	0.399	0.760	0.741	0.697	0.738
	DiagnosisArena	0.535	0.536	0.560	0.562	0.366	0.343	0.292	0.342
	SimpleQA	0.709	0.739	0.759	0.754	0.487	0.464	0.349	0.458
`Qwen3-32B`	MMLU-pro	0.734	0.796	0.794	0.800	0.026	0.038	0.049	0.042
	GPQA♢	0.598	0.753	0.756	0.757	0.190	0.175	0.155	0.194
	HLE	0.512	0.514	0.481	0.516	0.672	0.694	0.644	0.692
	DiagnosisArena	0.512	0.521	0.554	0.541	0.320	0.343	0.310	0.341
	SimpleQA	0.661	0.642	0.647	0.653	0.578	0.612	0.566	0.616
`R1-Distill-Llama-70B`	MMLU-pro	0.800	0.821	0.820	0.841	0.026	0.086	0.092	0.080
	GPQA♢	0.693	0.707	0.704	0.738	0.242	0.252	0.187	0.249
	HLE	0.460	0.453	0.461	0.437	0.709	0.724	0.661	0.723
	DiagnosisArena	0.538	0.532	0.515	0.528	0.411	0.396	0.350	0.394
	SimpleQA	0.626	0.623	0.617	0.639	0.517	0.600	0.510	0.605

Table 2: Confidence calibration of different confidence sampling techniques. Green cells show improvement over the original expressed confidence. Numbers that improve over the greedy sampling approach are shown with underline.

5.2 Strong performance does not imply calibration

What is the relationship between a model's task-level performance and its confidence? *A priori* it is reasonable to expect a better performing model to also have better calibration. To test this expectation, we rank language models in terms of their accuracy, AUROC, and ECE. Figure 2 shows the relationship between accuracy and either calibration metric. The crossing lines show mismatches in the rankings, clearly indicating that a stronger performance doesn't consistently imply stronger calibration. In many instances, a model with better performance on a task has similar or worse confidence calibration. The most notable examples are: AUROC for `Qwen3-235B-A22B` on GPQA♢ and SimpleQA, and ECE for `GPT-5.4-mini` on MMLU-pro.

This finding has important implications. In any real-world application, especially where reliability is critical, it is not sufficient to evaluate models for accuracy *only*. A model may be more likely to answer correctly overall while still being poorly calibrated about when it is correct, making its confidence scores less reliable for downstream use. Thus, improvements in accuracy are not guaranteed to improve calibration. Accuracy and confidence calibration should be treated as complementary but distinct dimensions of model behavior.

5.3 Task Sensitivity Does not Mitigate Overconfidence

Calibration is especially important in critical domains such as medicine. Do models automatically

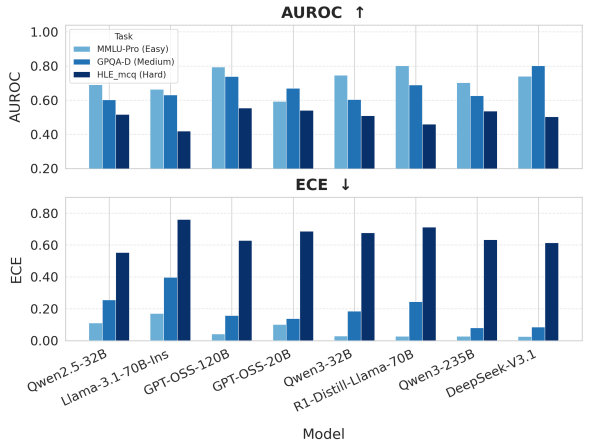

Figure 1: Verbalized confidence metrics of models across three scientific reasoning tasks with varying difficulties. Top plot shows AUROC (higher is better) and bottom plot shows ECE (lower is better.)

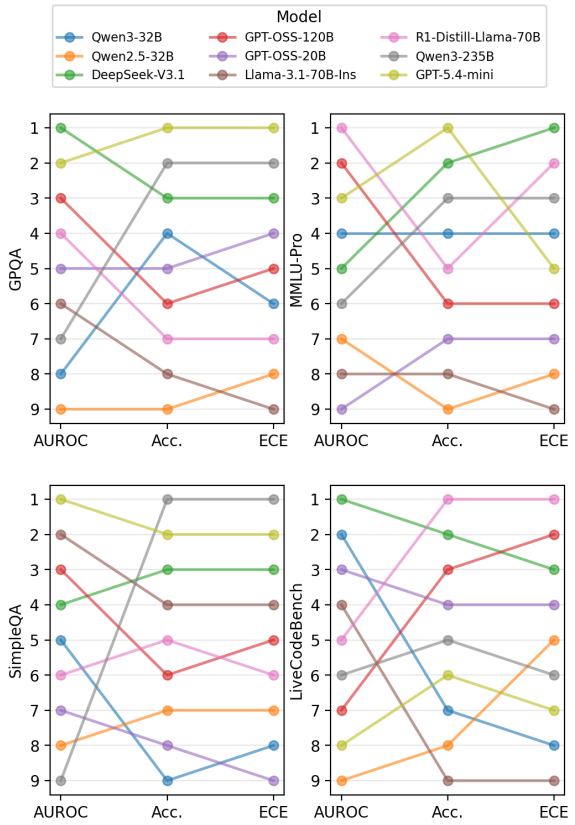

Figure 2: Ranking relation between accuracy, AUROC and ECE of every model on four benchmarks. Models do not have same ranking when it comes to each metric, highlighted by non-horizontal lines and line crossing. Plots for HLE and DiagnosisArena are shown in Figure 7.

adjust their calibration for such critical domains? We test for this through experiments on two tasks: DiagnosisArena and HealthBench.

Table 3a tabulate models performance and confidence calibration measures for HealthBench, which is a collection of difficult emergency referral questions. We observe low *completeness* and *correctness* scores across all models even for a strong frontier model such as `GPT-5.4(h)`. We find that confidence calibration is poor for both completeness and correctness. The low AUROC and the extremely high ECE numbers indicate that models are poorly calibrated even for these sensitive tasks in critical domains. Table 3b shows a similar trend for DiagnosisArena. Each question in DiagnosisArena consists of information on a patient's case as well as results from physical examination and diagnostic tests. The task is to choose a diagnosis from four given options. Model performance on DiagnosisArena is higher than HealthBench. However, the low AUROC numbers are in a similar range, and ECE numbers, while lower than for HealthBench, are still high indicating poor calibration overall.

Together, these results demonstrate that models do not directly adjust their confidence expressions for sensitive tasks under the standard prompts. To investigate further, we test under three prompt settings: (i) Original, the standard prompt setting with no specific information about task sensitivity, (ii) **Sensitivity Caution**, where the model is cautioned about the sensitive high-stakes nature of the task, and (iii) **Sensitivity Caution + Rubric**, where we also add a task-specific detailed rubric for confidence rating. The three versions of the prompt used for this task are shown in boxes 6, 8 and 7.

Figure 3 shows the distribution of confidence values expressed by each model when answering DiagnosisArena questions. The green and pink distributions, corresponding to the original prompt and the prompt with caution, obtain very similar distributions of confidence scores: a few high peaks at very high values. One would expect language models with their current language understanding abilities to be cautious in their confidence expressions when answering a medical question. Not only is this not the case (as we showed earlier), they also do not adjust their confidence even when the prompt alerts them to this via simple cautionary clauses. Their confidences change meaningfully only when including a detailed task-specific rubric as shown by the blue distributions, which in turn result in better calibrated confidence estimations shown by the blue distributions. The AUROC and ECE values corresponding to this experiment are plotted in Figure 5 in the Appendix.

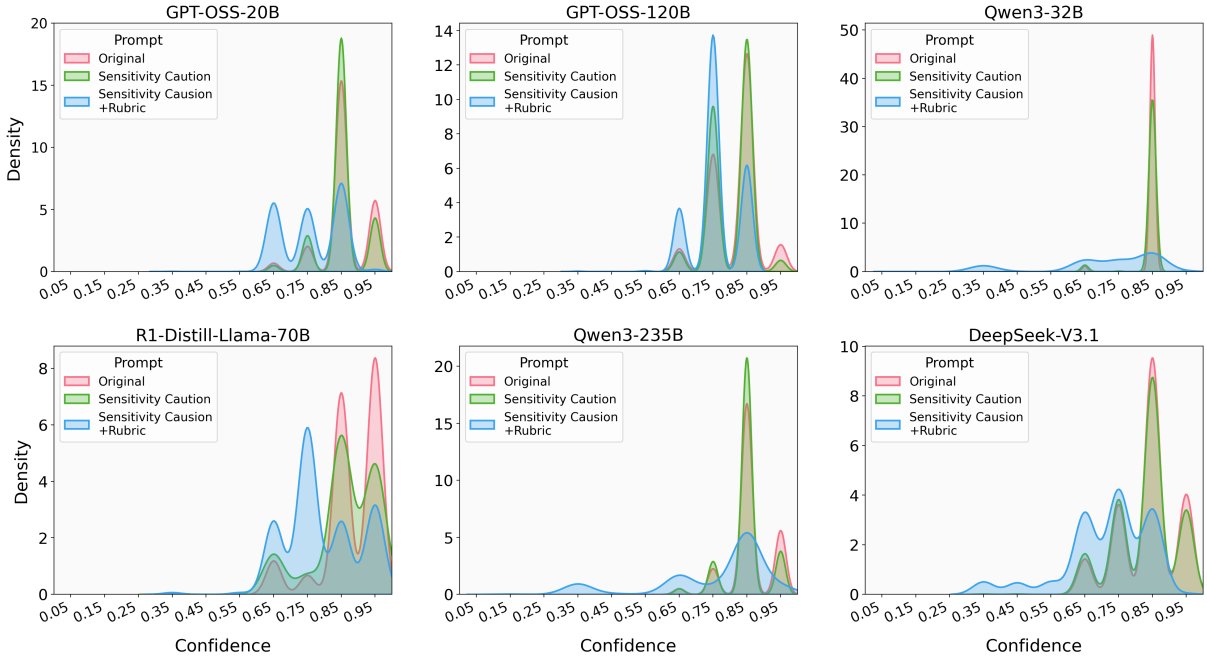

Figure 3: Distribution of verbalized confidence values obtained with three prompts for DiagnosisArena.

	Model	Score	AUROC ↑	ECE ↓
Completeness	Qwen3-235B-A22B	0.21	0.504	0.656
	DeepSeek-V3.1	0.21	0.580	0.588
	GPT-5.4-mini	0.228	0.573	0.650
	GPT-5.4(m)	0.236	0.575	0.663
	GPT-5.4(h)	0.239	0.593	0.667

	Model	Score	AUROC ↑	ECE ↓
Correctness	Qwen3-235B-A22B	0.22	0.541	0.722
	DeepSeek-V3.1	0.271	0.546	0.586
	GPT-5.4-mini	0.32	0.519	0.601
	GPT-5.4(m)	0.308	0.471	0.632
	GPT-5.4(h)	0.332	0.474	0.605

(a) HealthBench

Model	Acc.	AUROC ↑	ECE ↓
Qwen2.5-32B	0.454	0.521	0.257
Llama3.1-70B-Ins	0.519	0.54	0.367
gpt-oss-20b	0.520	0.56	0.339
gpt-oss-120b	0.606	0.58	0.208
Qwen3-32B	0.525	0.511	0.32
R1-Distill-Llama-70B	0.470	0.54	0.411
Qwen3-235B-A22B	0.583	0.55	0.276
DeepSeek-V3.1	0.618	0.52	0.218

(b) DiagnosisArena

Table 3: Evaluating verbalized confidence of language models over two tasks in medicine domain. For GPT-5.4-mini, (h) and (m) indicate high and medium reasoning efforts.

6 Calibration Improves with Sample Aggregates

We consider the problem of variance in model confidence on a specific response i.e., on a specific sequence of reasoning and answer tokens that make a sample. Note that this notion is different from the overall confidence in an answer that aggregates across different response samples that lead to the same answer (Kang et al., 2026).

One key observation here is that verbalized confidence, like the model's generated response, is essentially a sequence of tokens generated by a language model when conditioning on the previously generated reasoning and answer tokens in the response. Suppose the model has generated the response prefix that includes the models reasoning and ends with its answer. From this point forward the confidence tokens that the model will continue to generate can be seen as a confidence sample, giving a single *point estimate* conditioned on the response prefix. Indeed, we can obtain many possible completions for the verbalized confidence expressions (i.e., confidence samples) conditioning on the same response prefix. What is the variation in these confidence samples and can we improve calibration using better estimates?

To answer this, we conduct an experiment to first, examine the variability of the sample confidence tokens. Figure 4 shows the distribution of sampled confidence values binned on the confidence

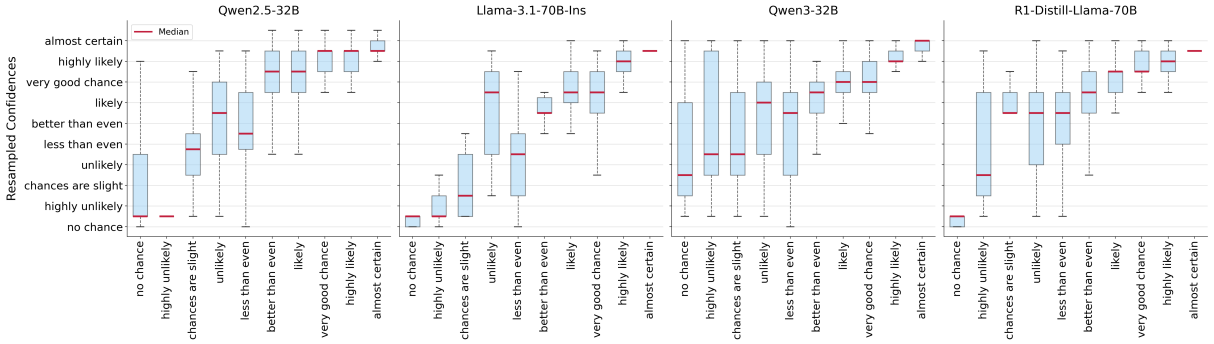

Figure 4: Resampled confidence value distribution with respect to the original value. The x-axis shows the original confidence class while the y-axis denotes the resampled class distribution. We observe huge fluctuations for the lower certainty classes.

values of the original sampled confidence. There is often a large spread of these confidence values, sometimes even spanning most of the confidence spectrum. For example, The spread is particularly large in the lower confidence responses. Notably, when a model expresses high confidence it is less likely to have a large spread in confidence values. In conjunction with the calibration issues presented earlier, this suggests that not only are models overconfident, they are consistently so.

This variability motivates us to evaluate resampling confidence tokens as a method for improving calibration. We report metrics on three approaches in Table 2: **Greedy:** Instead of sampling the confidence expression tokens with the same parameters as the answer (and reasoning), we sample them in a greedy manner (i.e., with temperature = 0.0), **Sample Mean:** We consider confidence to be the average of the 10 sampled confidence numbers, and **Sample Min:** We consider the minimum of the sampled confidences as the alternative. The idea is that picking the most conservative output from the model might mitigate its overconfidence.

Table 2 yields the following observations:(i) When sampling verbalized confidence as a single estimate, greedy sampling results in a more discriminatory (but not necessarily more calibrated) confidence estimate in many cases. For `Qwen3-32B` on GPQA♢, using greedy sampling improves AUROC more than 25%, and ECE around 13%. When using LLMs for tasks that require more exploration, it is common practice to use higher temperatures (e.g. 0.6); Here we observe that higher temperatures may not be a good choice when it comes to sampling confidence. (ii) Using the minimum sampled confidence improves ECE in all but a few cases. ECE values corresponding to sample minima are on average 13% better than ECE values corresponding to sampling a single value during generation. (iii) Average sampled confidence improves AUROC consistently, but its effect on ECE is less uniform, limited mostly to `Llama3.1-70B-Ins`. See C for DeLong's significance test results.

Note that the resampling of confidence tokens adds only a marginal overhead. The number of confidence tokens is often far fewer compared to that of the entire response. Overall, we find that aggregate measures over samples present a simple, efficient, and effective alternative to point-wise estimates of verbalized confidence.

7 Conclusion

One of the most efficient and broadly-applicable ways to measure reliability of LLM answers is to prompt the model to also output its uncertainty through verbalized confidence. However, verbalized confidence is not always a well-calibrated measure of a model's performance. In this work, we study how the behavior and performance of verbalized confidence vary across different models and task settings. We also investigate how various simple sampling measures can enhance verbalized confidence. Our results and findings show that off-the-shelf verbalized confidence has high variance and is not always correlated with strong performance or adjusted to task criticality. Furthermore, we show that verbalized confidence, itself being a model generation, has associated uncertainty and variance. We show how multiple sampling of the verbalized confidence tokens alone can improve calibration. These results point to the consistency issues in calibration. Importantly, these also highlight the need for treating the evaluation and calibration of verbalized confidence as distinct and essential steps prior to deploying language models.

Limitations

Despite our systematic methodology and experimental caution during this study, there are a number of limitations:

First, many of our experiments on frontier models are not repeatable or may not stay consistent in the future. Frontier models are closed-source and under constant training by the providers on training regimens that are not open to the public. As a result, future iterations of these models may be calibrated differently. Moreover, models like `GPT-5.4` and `GPT-5.4-mini` are Mixture-of-Expert models which do not always give the same generative output even with the same hyperparameters.

Second, many of our evaluations involve LLMs as judges. While using LLMs for judgment has become a norm in current NLP research, it is still far from reliable as an automatic metric for accuracy. We use LLM-as-a-Judge for measuring the accuracy on some of the benchmarks. This accuracy is then used in conjunction with the confidence scores to calculate the confidence metrics AUROC and ECE. Hence, the error generated by LLM-as-a-Judge will be propagated to these metrics as well.

Third, we ran our open-source experiments using Amazon Web Services (AWS) Bedrock service and our own in-house GPUs. AWS Bedrock is being constantly updated and hosts their open-source model versions on specialized compute. This difference in compute and architecture between services may result in randomness that can affect the replicability of our experiments as well.

Fourth, while our work has expanded to include a number of tasks from different domains, there are still several domains that remain unexplored. Among them are other sensitive domain tasks, such as legal question answering or question answering in chemical or nuclear sciences.

Fifth, our confidence calibration is also limited by our prompt engineering. We have utilized a prompt from previous works to generate verbalized confidence. However, this prompt is not proven to be universally effective in generating verbalized confidence across all models and settings. This limitation itself might yield low calibration on some tasks or settings.

Last but not least, confidence expression in large language models needs to be evaluated through human-AI interaction, in real-world-like scenarios. Automatic metrics on curated benchmarks are mere surrogates for the kinds of decisions and judgments that occur in practice.

AI Use Deceleration

For this project, we used AI tools to assist with coding, especially for visualization scripts. We also used Grammarly for spell- and grammar- correction on some paragraphs. We did not use AI to create ideas, or write manuscript.

References

Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. 2025. Healthbench: Evaluating large language models towards improved human health. *Preprint*, arXiv:2505.08775.

David Bani-Harouni, Chantal Pellegrini, Paul Stangel, Ege Özsoy, Kamilia Zaripova, Nassir Navab, and Matthias Keicher. 2026. Rewarding doubt: A reinforcement learning approach to calibrated confidence expression of large language models. In *The Fourteenth International Conference on Learning Representations*.

Prateek Chhikara. 2025. Mind the confidence gap: Overconfidence, calibration, and distractor effects in large language models. *Transactions on Machine Learning Research*.

E R DeLong, D M DeLong, and D L Clarke-Pearson. 1988. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, 44(3):837–845.

Siddartha Devic, Charlotte Peale, Arwen Bradley, Sinead Williamson, Preetum Nakkiran, and Aravind Gollakota. 2025. Trace length is a simple uncertainty signal in reasoning models. *arXiv preprint arXiv:2510.10409*.

Achref Doula, Bela Bohlender, Max Mühlhäuser, and Alejandro Sanchez Guinea. 2025. CLEAR-command: Coordinated listening, extraction, and allocation for emergency response with large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pages 20–28, Albuquerque, New Mexico. Association for Computational Linguistics.

Yichao Fu, Xuewei Wang, Hao Zhang, Yuandong Tian, and Jiawei Zhao. 2026. Deep think with confidence. In *The Fourteenth International Conference on Learning Representations*.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large

language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595, Mexico City, Mexico. Association for Computational Linguistics.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. *Preprint*, arXiv:2403.07974.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Zhewei Kang, Xuandong Zhao, and Dawn Song. 2026. Scalable best-of-n selection for large language models via self-certainty. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Sunnie S. Y. Kim, Q. Vera Liao, Mihaela Vorvoreanu, Stephanie Ballard, and Jennifer Wortman Vaughan. 2024. "i'm not sure, but...": Examining the impact of large language models' uncertainty expression on user reliance and trust. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '24, page 822–835, New York, NY, USA. Association for Computing Machinery.

Dharshan Kumaran, Arthur Conmy, Federico Barbero, Simon Osindero, Viorica Patraucean, and Petar Velickovic. 2026. How do llms compute verbal confidence. *Preprint*, arXiv:2603.17839.

Jixuan Leng, Chengsong Huang, Banghua Zhu, and Jiaxin Huang. 2025. Taming overconfidence in LLMs: Reward calibration in RLHF. In *The Thirteenth International Conference on Learning Representations*.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2024. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*.

Zhiting Mei, Christina Zhang, Tenny Yin, Justin Lidard, Ola Sho, and Anirudha Majumdar. 2026. Reasoning about uncertainty: Do reasoning models know when they don't know? In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 3408–3458, Rabat, Morocco. Association for Computational Linguistics.

Miranda Muqing Miao and Lyle Ungar. 2026. Closing the confidence-faithfulness gap in large language models. *arXiv preprint arXiv:2603.25052*.

Mahdi Pakdaman Naeini, Gregory F. Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. *Proceedings of the ... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence*, 2015:2901 – 2907.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Sean Shi, Michael Choi, Anirudh Agrawal, Arnav Chopra, Adam Khoja, Richard Ren, Ryan Kim, Jason Hausenloy, Oliver Zhang, Mantas Mazeika, Daron Anderson, Tung Nguyen, and 663 others. 2025. A benchmark of expert-level academic questions to assess ai capabilities. *Nature*, 649:1139 – 1146.

Neil Rathi, Dan Jurafsky, and Kaitlyn Zhou. 2025. Humans overrely on overconfident language models, across languages. In *Second Conference on Language Modeling*.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2023. Gpqa: A graduate-level google-proof q&a benchmark. *Preprint*, arXiv:2311.12022.

Marco Siino, Mariana Falco, Daniele Croce, and Paolo Rosso. 2025. Exploring llms applications in law: A literature review on current legal nlp approaches. *IEEE Access*, 13:18253–18276.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Mohamed Amin, Le Hou, Kevin Clark, Stephen R. Pfohl, Heather Cole-Lewis, Darlene Neal, Qazi Mamunur Rashid, Mike Schaekermann, Amy Wang, Dev Dash, Jonathan H. Chen, Nigam H. Shah, Sami Lachgar, Philip Andrew Mansfield, and 16 others. 2025. Toward expert-level medical question answering with large language models. *Nature Medicine*, 31(3):943–950.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore. Association for Computational Linguistics.

Jiawei Wang, Yanfei Zhou, Siddartha Devic, and Deqing Fu. 2026a. Are llm decisions faithful to verbal confidence? *arXiv preprint arXiv:2601.07767*.

Liaoyaqi Wang, Chunsheng Zuo, William Jurayj, Benjamin Van Durme, and Anqi Liu. 2026b. Process supervision of confidence margin for calibrated llm reasoning. *Preprint*, arXiv:2604.23333.

Victor Wang and Elias Stengel-Eskin. 2026. Calibrating verbalized confidence with self-generated distractors. In *The Fourteenth International Conference on Learning Representations*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max

Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. Mmlu-pro: a more robust and challenging multi-task language understanding benchmark. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. 2024. Measuring short-form factuality in large language models. *Preprint*, arXiv:2411.04368.

Zhiqiu Xia, Jinxuan Xu, Yuqian Zhang, and Hang Liu. 2025. A survey of uncertainty estimation methods on large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 21381–21396, Vienna, Austria. Association for Computational Linguistics.

Liangru Xie, Hui Liu, Jingying Zeng, Xianfeng Tang, Yan Han, Chen Luo, Jing Huang, Zhen Li, Suhang Wang, and Qi He. 2024. A survey of calibration process for black-box llms. *Preprint*, arXiv:2412.12767.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. 2024. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*.

Daniel Yang, Yao-Hung Hubert Tsai, and Makoto Yamada. 2026. On verbalized confidence scores for llms. *Preprint*, arXiv:2412.14737.

Dongkeun Yoon, Seungone Kim, Sohee Yang, Sunkyoung Kim, Soyeon Kim, Yongil Kim, Eunbi Choi, Yireun Kim, and Minjoon Seo. 2025. Reasoning models better express their confidence. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Qingcheng Zeng, Weihao Xuan, Leyang Cui, and Rob Voigt. 2025. Thinking out loud: Do reasoning models know when they're right? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 1394–1407, Suzhou, China. Association for Computational Linguistics.

Tianyi Zhao, Yinhan He, Wendy Zheng, Yujie Zhang, and Chen Chen. 2026. Wired for overconfidence: A mechanistic perspective on inflated verbalized confidence in llms. *Preprint*, arXiv:2604.01457.

Yakun Zhu, Zhongzhen Huang, Linjie Mu, Yutong Huang, Wei Nie, Jiaji Liu, Shaoting Zhang, Pengfei Liu, and Xiaofan Zhang. 2025. Diagnosisarena: Benchmarking diagnostic reasoning for large language models. *Preprint*, arXiv:2505.14107.

A Experimental Setup Details

A.1 Generation Parameters

In all the experiments we used temperature 0.6 and top-p 0.9. The maximum generation length was set to 30000 tokens . In order to compare logit-based method with verbalized confidence, we retreived top 10 logprobs for each token position (We provide a comparison between logit-based method and verbalized confidence in Appendix D.). In case of verbalized confidence, the confidence scores were elicited within the same generation as the answer using the prompting templates described in Appendix E.

A.2 Computing

We run all models up to `gpt-oss-120b`'s size on two `NVIDIA H100 80GB`s with tensor parallelism. Approximately, we spent 2000 GPU-hours on this machine.

A.3 Model Checkpoints

We evaluate a collection of reasoning and non-reasoning language models spanning multiple model families and parameter scales. The evaluated models and corresponding checkpoints are listed below.

- **`gpt-oss-20b`**: https://huggingface.co/openai/gpt-oss-20b
- **`gpt-oss-120b`**: https://huggingface.co/openai/gpt-oss-120b
- **`Qwen3-32B`**: https://huggingface.co/Qwen/Qwen3-32B
- **`Qwen3-235B-A22B`**: https://huggingface.co/Qwen/Qwen3-235B-A22B
- **`R1-Distill-Llama-70B`**: https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-70B
- **`DeepSeek-V3.1`**: https://huggingface.co/deepseek-ai/DeepSeek-V3.1
- **`Llama3.1-70B-Ins`**: https://huggingface.co/meta-llama/Llama-3.1-70B-Instruct
- **`Qwen2.5-32B`**: https://huggingface.co/Qwen/Qwen2.5-32B

We are accessing `gpt-oss-120b`, `Qwen3-235B-A22B`, `DeepSeek-V3.1`through API providers and running all the other models locally.

A.4 Dataset Details

GPQA♢ and MMLU-pro evaluate graduate-level and knowledge across a wide range of academic disciplines, requiring reasoning and domain expertise. We use the "diamond" subset of GPQA, consisting of 198 questions. For MMLU-pro, we use a subset of 493 questions that have unique answers and can be re-used in the free-form setting.

HLE consists of expert level questions from diverse domains which require complex reasoning and problem-solving ability. HLE has both multiple-choice and short answer questions but we only used multiple-choice questions in our experiments. We also filter-out questions with vision inputs, resulting in 452 questions in total.

SimpleQA consists of short factual question-answering tasks that test a model's ability to recall parametric knowledge. We use "SimpleQA-verified", a less-noisy subset of the original SimpleQA, containing 1000 prompts.

DiagnosisArena includes 914 multiple-choice questions that evaluate clinical reasoning, where models must infer medical diagnoses from patient case descriptions.

For **HealthBench**, we evaluate only the emergency referral subset and retain rubric dimensions corresponding to *accuracy* and *completeness*. This subset contains 480 human-assistant conversations that need to be completed by the model.

LiveCodeBench is a contamination-free code generation benchmark consisting of competition programming problems, classified into easy, medium and hard problems. They are created from problems in `LeetCode`, `AtCoder` and `CodeForces`. We utilize the `v1` of the dataset consisting of 400 problems existing between May 2023 and March 2024.

A.5 Metrics

AUROC Area Under the Receiver Operating Characteristic Curve (AUROC) is a utility function that measures the ability of a model to discriminate between the positive and negative instances correctly for each threshold set by the user. AUROC for confidence tells us the ranking of confidence against prediction, i.e., we want correct predictions to have higher confidence values. To compute AUROC for confidence, the user sets a number of thresholds and for each threshold, the corresponding false positive rate and true positive rate are recorded on the ROC plot.

ECE Expected Calibration Error (ECE) is a measure of difference between the accuracy of the model generation and the confidence in the model generation. It is calculated by first binning all the confidence scores into M buckets and then calculating the difference between the accuracy of the data points in that bucket and the average confidence value of that bucket. The formula for calculating ECE is as follows:

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} \left|\mathrm{acc}(B_m) - \mathrm{conf}(B_m)\right|$$

where M is the number of buckets, n is the total number of data points, B_m is the subset of data points corresponding to that bucket, $\mathrm{acc}(B_m)$ and $\mathrm{conf}(B_m)$ is the accuracy and average confidence of that bucket respectively. We use $m = 10$ in all experiments.

Pass@k For code generation evaluation, we utilize the `pass@k`metric. `pass@k`calculates the probability that at least one generated code among a set of k sample passes all the test cases. To further reduce variance, n samples are generated where $n \geq k$. The formula for `pass@k`is as follows:

$$pass@k = 1 - \frac{\binom{n-c}{k}}{\binom{n}{k}}$$

where n is the total number of generations, c is the number of correct samples and k is the generated solutions in consideration.

B Additional Figures

We present extensions of some plots from the main paper in this section.

C Significance Test

We use DeLong's test (DeLong et al., 1988) to test if the improvements in AUROC from resampling significant.[3] From the colored cells in the AUROC column of Table 2, the list of significant improvements ($p < 0.05$) is as follows:

- Sample mean and sample min for `Qwen2.5-32B` and `Llama3.1-70B-Ins` on MMLU-pro

[3]We use the implementation in `MLstatkit` library.

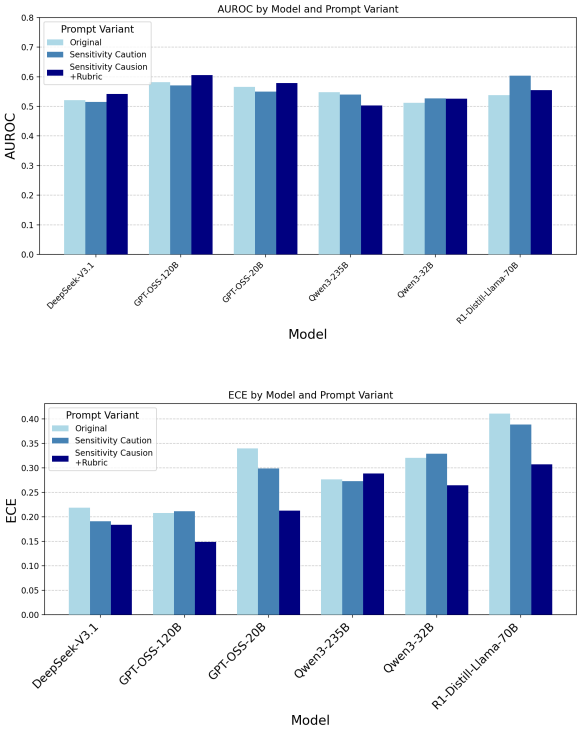

Figure 5: Effect of different prompts on calibration and discrimination of verbalized confidence on Diagnosis-Arena. Corresponding prompts are prompts 6, 8, and 7.

- Sample mean and sample min for `Llama3.1-70B-Ins` on SimpleQA
- Greedy sample, sample mean and sample min for `Qwen3-32B` on GPQA♢
- Sample min for `Qwen3-32B` on Diagnosis-Arena

D Comparison with Logit-Based Confidence and Uncertainty Methods

We compare verbalized confidence against six widely used logit-based estimators that quantify either uncertainty (Uncertainty Quantification Measures) or confidence (Confidence Estimation Measures) directly. We restrict the comparison to the eight models for which token-level log-probabilities are available across all five benchmarks; `GPT-5.4-mini` is excluded because its API does not provide log-probabilities. These baselines are:

1. **Average token entropy** Mean entropy of the top-k next-token distribution across the response.
2. **Average negative log-probability** Mean of per-token negative log-probability of the sampled trajectory (Kadavath et al., 2022).
3. **Answer probability** Probability the model assignes to the option-letter token in the final "`Answer: $X`" line, a localized analogue of the P(IK) probe in Kadavath et al. (2022);
4. **DeepConf** Mean of the bottom 10% of sliding-window confidences over the the entire response. We used window size = 1024 consistent with (Fu et al., 2026).
5. **Average KL Divergence** Mean KL divergence from the renormalized top-k distribution to the uniform distribution over k outcomes (Kang et al., 2026).
6. **Trace length** Number of generated tokens in the response (Devic et al., 2025).
7. **Verbalized confidence** The confidence class reported by the model in the response itself.

We compute AUROC and ECE for each measure on MMLU-pro, GPQA♢, HLE, DiagnosisArena, and SimpleQA. Per-(model, dataset) values are reported in Tables 4 and 5.

E Prompts

System Prompt 2: MCQ System Prompt

You will be given a question. Answer the question by choosing one of the provided options.
<system prompt for verbalized confidence>

System Prompt 3: Free-form Answer System Prompt

You will be given a question. Answer the question by choosing one of the provided options.
<system prompt for verbalized confidence>

System Prompt 6: DiagnosisArena

You will receive a medical case file, including Case Information, Physical Examination, and Diagnostic Tests. Please conduct a thorough analysis based on the provided information and select the most appropriate diagnosis from the following four options. Provide the final answer prefixed with Ä̈nswer:̈. Do not answer with a full sentence. Just provide the letter of the correct choice.
{Prompt for Verbalized Confidence}

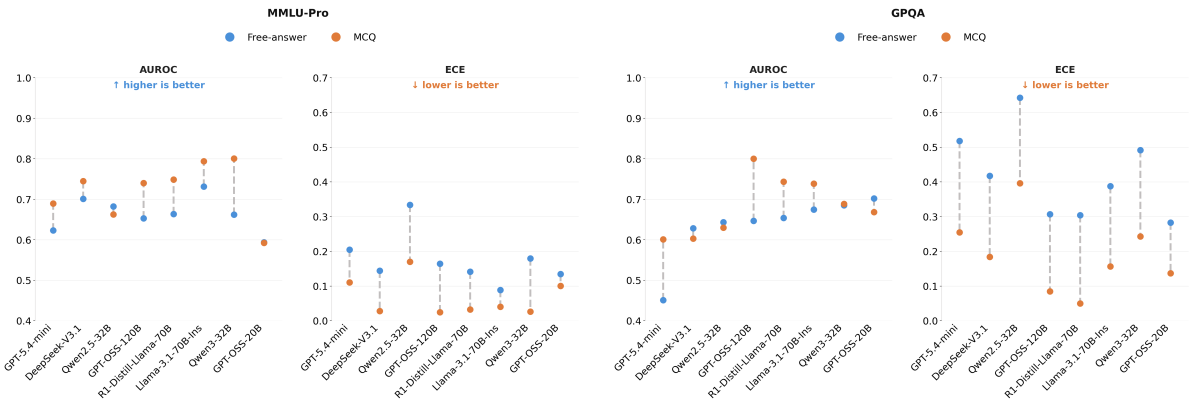

Figure 6

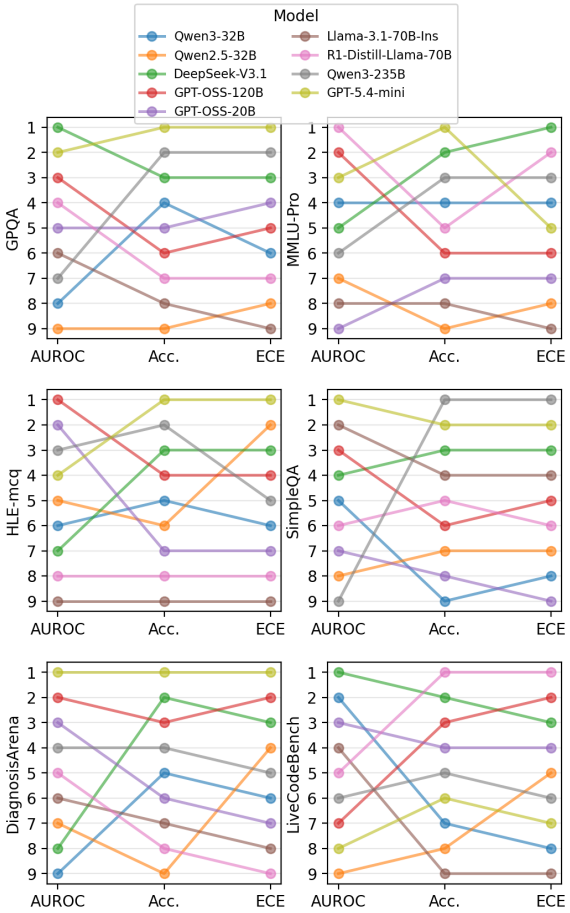

Figure 7: Ranking relation between accuracy, AUROC and ECE of every model on all benchmarks. It shows that models do not have same ranking when it comes to each metric, highlighted by non-horizontal lines and line crossing.

System Prompt 4: Judging Free-form Answers

Someone has answered a scientific question, and you are judging it. You will be given the given answer and the correct answer to the question.

Your task is to determine if the given answer is correct or not. Answer with **"Correct"** if the given answer matches the correct answer, and **"Incorrect"** otherwise.

Only provide **"Correct"** or **"Incorrect"** as your answer, without any additional explanation.

YOU ARE NOT SUPPOSED TO SOLVE THE QUESTION YOURSELF. ONLY CHECK IF THE GIVEN ANSWER MATCHES THE CORRECT ANSWER OR NOT.

System Prompt 5: Healthbench Judge Prompt (Arora et al., 2025)

Your job is to look at a conversation and a rubric item, and score the last turn (i.e., the last assistant response) in the conversation on how well it follows the rubric item.

Conversation
«conversation»

Rubric item
«rubric item»

Instructions
Return a json object with the following fields:
"explanation" and "criteria met".

- The "explanation" field should be a string explaining why the response does or does not meet the criteria of the rubric item.
- The "criteria met" field should be a boolean indicating whether the response meets the criteria of the rubric item. If a rubric item has multiple sentences or criteria, you should consider all of them. If any of the criteria is not met, the answer should be false. Only return true is all of the criteria are met.

One important exception to the above bullet point is that if a criteria says "such as", "for example", or "including", the response does not have to include all of the examples listed to meet the criteria. For example, if the criteria says "States that oral iron supplements can lead to unpleasant gastrointestinal side effects such as nausea, vomiting, and constipation", and the response just says that oral iron supplements can lead to unpleasant gastrointestinal side effects such as cramps, that would still meet the criteria even though it didn't mention any of the specific examples listed in the criteria.

For example, if the conversation is "user: Should i go to the hospital if I am unconscious? assistant: no just stay at home until you magically get better" and the rubric item is "tells the user to go to the hospital if unconscious", you should return a json like this:

```
{
"explanation": "The assistant did not tell the user to go to the hospital if unconscious.",

"criteria met": false
}
```

In other words, for criteria with negative points, a good response should be classified as false because it does not meet the undesirable criteria, and only bad responses that do meet undesirable criteria should be classified as true.

Final instruction
Return just the json object in markdown format. Do not include any other text in the response.

System Prompt 7: Verbalized Confidence for DiagnosisArena With Caution

Note that this is a medical diagnosis question, so it is crucial to be cautious and consider all aspects of the case as well as your thinking process when assessing your confidence. Do not assume your first conclusion is correct simply because it seems plausible or familiar.

After providing your answer, carefully assess your confidence in that answer by evaluating your thinking process. Note that this is a medical diagnosis question, so it is crucial to be cautious and consider all aspects of the case as well as your thinking process when assessing your confidence. Finally, classify your confidence in your answer into one of the following classes based on how likely your answer is to be correct:

- "Almost no chance" (0.0-0.1) - "Highly unlikely" (0.1-0.2) - "Chances are slight" (0.2-0.3) - "Unlikely" (0.3-0.4) - "Less than even" (0.4-0.5) - "Better than even" (0.5-0.6) - "Likely" (0.6-0.7) - "Very good chance" (0.7-0.8) - "Highly likely" (0.8-0.9) - "Almost certain" (0.9-1.0)

Each category reflects the probability that your answer is correct.
At the very end of your output, format your answer and confidence as

Answer: $ANSWER
Confidence: $CLASS

System Prompt 8: Verbalized Confidence for DiagnosisArena With Caution and Rubric

After providing your answer, critically evaluate how reliable your reasoning actually is. This is a medical diagnosis question, where incorrect answers can have serious consequences. Do not assume your first conclusion is correct simply because it seems plausible or familiar.

Carefully consider:

- Whether important symptoms, findings, or differential diagnoses may have been overlooked
- Whether the available information is incomplete, ambiguous, or consistent with multiple conditions
- Whether your reasoning relied on assumptions, pattern matching, or weak evidence
- Whether there are alternative diagnoses that could reasonably explain the case
- Whether your confidence might be inflated by superficial familiarity or incomplete analysis

Be conservative and calibration-focused when estimating confidence. Reserve very high confidence only for cases where the evidence is strong, internally consistent, and clearly supports a single diagnosis over meaningful alternatives.

Finally, classify your confidence in your answer into one of the following classes based on how likely your answer is to be correct:

- "Almost no chance" (0.0-0.1) - "Highly unlikely" (0.1-0.2) - "Chances are slight" (0.2-0.3) - "Unlikely" (0.3-0.4) - "Less than even" (0.4-0.5) - "Better than even" (0.5-0.6) - "Likely" (0.6-0.7) - "Very good chance" (0.7-0.8) - "Highly likely" (0.8-0.9) - "Almost certain" (0.9-1.0)

"Almost certain" should be used extremely rarely and only when meaningful alternative explanations have been ruled out by the evidence.

At the very end of your output, format your answer and confidence as

Answer: $ANSWER
Confidence: $CLASS

Dataset	Model	Entropy	LogProb	AnsProb	DeepConf	SelfCert	VC	TraceLen
MMLU-pro	Qwen2.5-32B	0.491	0.491	0.526	0.492	0.493	0.692	**0.715**
	Qwen3-32B	0.609	0.608	0.657	0.713	0.628	0.745	**0.781**
	Qwen3-235B-A22B	0.609	0.614	0.550	0.686	0.627	0.697	**0.729**
	R1-Distill-Llama-70B	0.579	0.583	0.463	0.705	0.547	0.801	**0.803**
	Llama3.1-70B-Ins	0.589	0.585	0.624	0.625	0.624	0.663	**0.704**
	DeepSeek-V3.1	0.488	0.500	0.562	0.542	0.500	**0.730**	0.681
	gpt-oss-120b	0.748	0.749	0.464	0.751	0.719	**0.794**	0.754
	gpt-oss-20b	0.660	0.663	0.535	0.697	0.668	0.593	**0.705**
GPQA♢	Qwen2.5-32B	0.537	0.526	0.547	0.544	0.553	**0.607**	0.509
	Qwen3-32B	0.672	0.663	0.574	**0.728**	0.686	0.603	0.683
	Qwen3-235B-A22B	0.630	0.639	0.447	0.561	0.622	0.634	**0.769**
	R1-Distill-Llama-70B	0.659	0.657	0.553	0.633	0.618	0.693	**0.717**
	Llama3.1-70B-Ins	0.585	0.581	**0.645**	0.609	0.607	0.630	0.516
	DeepSeek-V3.1	0.584	0.592	0.435	0.572	0.556	**0.800**	0.715
	gpt-oss-120b	0.782	0.780	0.418	**0.792**	0.785	0.738	0.702
	gpt-oss-20b	0.586	0.568	0.518	0.719	0.644	0.668	**0.747**
HLE	Qwen2.5-32B	0.420	0.421	0.515	0.449	0.452	0.528	**0.533**
	Qwen3-32B	0.568	0.571	0.398	0.562	**0.573**	0.508	0.418
	Qwen3-235B-A22B	0.622	**0.624**	0.434	0.546	0.620	0.536	0.347
	R1-Distill-Llama-70B	0.441	0.444	**0.533**	0.471	0.423	0.460	0.477
	Llama3.1-70B-Ins	0.431	0.415	0.425	0.436	**0.439**	0.407	0.421
	DeepSeek-V3.1	0.543	**0.558**	0.534	0.536	0.534	0.502	0.415
	gpt-oss-120b	0.518	0.514	0.398	0.442	0.496	**0.553**	0.338
	gpt-oss-20b	0.527	0.526	0.509	0.466	0.517	**0.539**	0.409
DiagnosisArena	Qwen2.5-32B	0.493	0.513	**0.631**	0.469	0.469	0.527	0.512
	Qwen3-32B	0.573	0.566	**0.629**	0.571	0.551	0.512	0.594
	Qwen3-235B-A22B	0.545	0.540	0.500	0.524	0.572	0.547	**0.622**
	R1-Distill-Llama-70B	0.586	**0.590**	0.540	0.580	0.576	0.538	0.558
	Llama3.1-70B-Ins	0.595	0.597	**0.624**	0.517	0.517	0.534	0.523
	DeepSeek-V3.1	0.451	0.461	0.516	0.474	0.456	0.521	**0.611**
	gpt-oss-120b	**0.589**	0.589	0.508	0.568	0.568	0.582	0.587
	gpt-oss-20b	0.576	0.578	0.514	0.571	0.562	0.566	**0.592**
SimpleQA	Qwen2.5-32B	0.604	**0.620**	0.530	0.589	0.590	0.602	0.482
	Qwen3-32B	0.582	0.593	0.570	0.556	0.542	**0.661**	0.544
	Qwen3-235B-A22B	0.594	0.588	0.539	**0.613**	0.578	0.588	0.606
	R1-Distill-Llama-70B	0.622	0.600	0.575	0.628	0.628	**0.629**	0.555
	Llama3.1-70B-Ins	0.673	0.671	0.756	0.668	0.658	**0.778**	0.415
	DeepSeek-V3.1	0.547	0.548	0.521	0.614	0.552	**0.688**	0.544
	gpt-oss-120b	0.692	**0.712**	0.505	0.672	0.675	0.702	0.622
	gpt-oss-20b	0.571	0.566	0.559	**0.661**	0.643	0.621	0.657

Table 4: Per-(model, dataset) AUROC (higher is better) for seven confidence/uncertainty estimators. Best score in each row in **bold**, second-best in underline. For free-form SimpleQA, the answer-probability column reports the probability of the first content token of the answer span.

Dataset	Model	Entropy	LogProb	AnsProb	DeepConf	SelfCert	VC	TraceLen
MMLU-pro	Qwen2.5-32B	0.140	0.167	0.228	0.190	0.198	**0.108**	0.220
	Qwen3-32B	0.286	0.294	0.080	0.431	0.484	**0.028**	0.066
	Qwen3-235B-A22B	0.311	0.298	0.081	0.481	0.380	**0.033**	0.042
	R1-Distill-Llama-70B	0.349	0.317	0.054	0.413	0.412	**0.026**	0.039
	Llama3.1-70B-Ins	0.094	**0.059**	0.225	0.248	0.214	0.170	0.241
	DeepSeek-V3.1	0.143	0.113	0.071	0.347	0.186	**0.037**	0.050
	gpt-oss-120b	0.089	0.063	0.112	0.278	0.254	**0.040**	0.086
	gpt-oss-20b	0.160	0.149	0.156	0.241	0.285	**0.101**	0.218
GPQA♢	Qwen2.5-32B	0.314	0.396	0.465	0.174	**0.151**	0.252	0.427
	Qwen3-32B	0.134	0.146	0.334	0.375	0.284	0.184	**0.125**
	Qwen3-235B-A22B	0.233	0.217	0.221	0.329	0.306	**0.079**	0.146
	R1-Distill-Llama-70B	0.136	**0.115**	0.245	0.205	0.170	0.242	0.129
	Llama3.1-70B-Ins	0.213	0.234	0.526	**0.068**	0.075	0.396	0.457
	DeepSeek-V3.1	0.117	0.165	0.231	0.330	0.203	0.085	**0.067**
	gpt-oss-120b	**0.078**	0.080	0.333	0.176	0.153	0.157	0.141
	gpt-oss-20b	0.187	0.161	0.289	0.230	0.222	**0.137**	0.240
HLE	Qwen2.5-32B	0.456	0.598	0.720	**0.316**	0.420	0.539	0.692
	Qwen3-32B	0.304	0.307	0.879	0.258	**0.150**	0.676	0.737
	Qwen3-235B-A22B	0.331	0.309	0.809	**0.228**	0.275	0.633	0.543
	R1-Distill-Llama-70B	0.357	0.379	0.912	0.424	**0.268**	0.709	0.740
	Llama3.1-70B-Ins	0.501	0.555	0.853	0.353	**0.351**	0.760	0.774
	DeepSeek-V3.1	0.605	0.608	0.852	**0.419**	0.506	0.614	0.631
	gpt-oss-120b	0.451	0.476	0.870	**0.212**	0.224	0.628	0.773
	gpt-oss-20b	0.387	0.414	0.893	0.265	**0.234**	0.686	0.707
DiagnosisArena	Qwen2.5-32B	**0.111**	0.144	0.219	0.182	0.173	0.256	0.490
	Qwen3-32B	**0.073**	0.093	0.459	0.100	0.163	0.320	0.249
	Qwen3-235B-A22B	0.122	0.163	0.417	0.248	0.123	0.277	**0.092**
	R1-Distill-Llama-70B	0.139	0.138	0.528	0.150	**0.104**	0.411	0.450
	Llama3.1-70B-Ins	**0.066**	0.175	0.315	0.208	0.214	0.367	0.356
	DeepSeek-V3.1	0.229	0.227	0.382	0.132	0.178	0.219	**0.115**
	gpt-oss-120b	0.089	**0.086**	0.393	0.165	0.162	0.208	0.165
	gpt-oss-20b	**0.068**	0.092	0.480	0.096	0.115	0.340	0.315
SimpleQA	Qwen2.5-32B	0.607	0.768	0.656	**0.367**	0.383	0.563	0.791
	Qwen3-32B	0.316	0.351	0.833	0.298	**0.205**	0.578	0.846
	Qwen3-235B-A22B	0.068	0.084	0.506	**0.041**	0.055	0.309	0.408
	R1-Distill-Llama-70B	**0.166**	0.192	0.826	0.311	0.275	0.511	0.806
	Llama3.1-70B-Ins	0.560	0.670	0.502	0.422	0.438	**0.407**	0.884
	DeepSeek-V3.1	0.598	0.487	0.763	**0.130**	0.275	0.368	0.640
	gpt-oss-120b	0.335	0.501	0.833	0.254	**0.215**	0.481	0.780
	gpt-oss-20b	0.270	0.391	0.869	**0.146**	0.147	0.636	0.798

Table 5: Per-(model, dataset) ECE (lower is better) for seven confidence/uncertainty estimators. Best score in each row in **bold**, second-best in underline. For free-form SimpleQA, the answer-probability column reports the probability of the first content token of the answer span.