TOKENS, NUMBERS, OR WORDS? BENCHMARKING HOW LARGE LANGUAGE MODELS EXPRESS UNCERTAINTY

Anonymous authors
Paper under double-blind review

ABSTRACT

As large language models (LLMs) are increasingly deployed in high-stakes applications, robust uncertainty estimation is essential for ensuring the safe and trustworthy deployment of LLMs. We present a large-scale study of uncertainty estimation in LLMs, evaluating 80 models spanning open- and closed-source families, dense and Mixture-of-Experts (MoE) architectures, reasoning and non-reasoning modes, quantization variants and parameter scales from 0.6B to 671B. Focusing on three representative single-pass methods that require no access to hidden states, including token probability-based uncertainty (TPU), numerical verbal uncertainty (NVU), and linguistic verbal uncertainty (LVU), we systematically evaluate uncertainty calibration and selective classification using the challenging MMLU-Pro benchmark, which covers both reasoning-intensive and knowledge-based tasks. Our results show that, on MMLU-Pro, LVU outperforms TPU and NVU on average and for most models, offering stronger calibration and discrimination. We also find that high accuracy does not imply reliable uncertainty, and that model scale, post-training and reasoning ability influence estimation performance, whereas the effects of quantization are small and method-dependent. Notably, LLMs exhibit better uncertainty estimates on reasoning tasks than on knowledge-heavy ones, and calibration and error ranking are only partially aligned. These findings highlight the need for multi-perspective evaluation and position LVU as a promising, though not universally superior, tool for improving the reliability of LLMs.

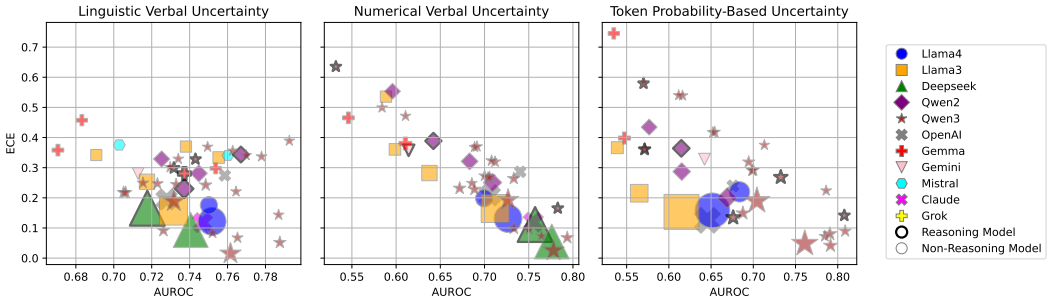

Figure 1: **AUROC vs. ECE across various LLMs, evaluated using Linguistic Verbal Uncertainty (LVU), Numerical Verbal Uncertainty (NVU) and Token Probability-based Uncertainty (TPU).** Model families are distinguished by color and marker shape. Reasoning-focused models are highlighted with thick black borders, while non-reasoning models are shown with dashed gray borders. Marker size reflects model size.

1 INTRODUCTION

As large language models (LLMs) become increasingly integrated into high-stakes applications such as medical decision support and legal consultation (Thirunavukarasu et al., 2023; Van Veen et al., 2024; Abacha et al., 2024; Hadi et al., 2023; Lai et al., 2024), growing concerns have emerged

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

around the public's over-reliance on their outputs (Passi & Vorvoreanu, 2022; ISACA, 2024). For instance, a New York lawyer relied on ChatGPT for legal research and subsequently submitted fabricated case citations generated by the model (BBC News, 2023). A promising way to mitigate such risks is to equip LLMs with *reliable uncertainty estimation*—enabling the model not only to quantify its uncertainty in a given response (*uncertainty calibration*), but also to distinguish between likely correct and incorrect predictions (*selective classification*). These capabilities support safer deployment: systems can abstain from uncertain predictions, flag high-risk outputs for human review, or adjust decision thresholds dynamically. In short, trustworthy LLMs must not only be accurate, but also well-calibrated and uncertainty-aware.

To extract uncertainty from large language models, a variety of methods have been proposed, which can be broadly categorized into five classes, as summarized in Table 1. (i) *Token Probability-Based Uncertainty (TPU)* estimates uncertainty from output token probabilities, such as perplexity (Duan et al., 2024; Lamb et al., 2025); (ii) *Internal state-based methods* leverage model internals like hidden states or attention maps (Kossen et al., 2024; Ji et al., 2025; Zhu et al., 2025), but require white-/gray-box access. In contrast, black-box-friendly approaches include: (iii) *Consistency-based methods*, which compute uncertainty from the disagreement across multiple generations (Kuhn et al., 2023; Manakul et al., 2023); (iv) *Numerical Verbal Uncertainty (NVU)*, where the model self-reports a numeric uncertainty score (Tian et al., 2023; Xiong et al., 2024); and (v) *Linguistic Verbal Uncertainty (LVU)*, which infers uncertainty from hedging language in responses, evaluated via a separate judging LLM (Yona et al., 2024; Belem et al., 2024; Ji et al., 2025). LVU expresses uncertainty in natural language, which is closer to how humans communicate uncertainty, although the same hedging expression can be interpreted differently by different readers (Belem et al., 2024).

Prior empirical studies have typically examined one uncertainty modality at a time. Xiong et al. (2024) study NVU with a focus on prompting and sampling strategies; Yona et al. (2024) study whether LLMs faithfully express uncertainty in words, on a small set of models; and Zhu et al. (2023) analyze token-probability calibration and the effect of instruction tuning. What remains missing is a head-to-head comparison of token-based, numerical, and linguistic uncertainty *under the same prompts, data, and metrics*, across enough models to separate the effect of the method from the effect of the model. Given the rapid evolution of LLMs (Guo et al., 2025; Qwen Team, 2025; AI, 2025; OpenAI, 2025b; xAI, 2025), the saturation of older benchmarks (Joshi et al., 2017; Hendrycks et al., 2021), and the emergence of novel uncertainty estimation techniques (Xiong et al., 2024; Yona et al., 2024), such a controlled comparison is needed to guide practitioners in choosing how to extract uncertainty from a given model.

In this work, we present such an evaluation, systematically studying uncertainty estimation in 80 state-of-the-art LLMs. Our benchmark covers both closed-source (e.g., OpenAI GPT, Anthropic Claude, Google Gemini) and open-source (e.g., Meta LLaMA, Qwen3, LLaMA-4, DeepSeek, Mistral) models, ranging from 0.6B to 671B parameters. We assess uncertainty through two complementary tasks: *uncertainty calibration*, which evaluates how well predicted uncertainty aligns with correctness (measured by ECE), and *uncertainty-based selective classification*, which assesses the model's ability to distinguish correct from incorrect predictions (measured by AUROC). All evaluations are conducted on *MMLU-Pro* (Wang et al., 2024), a recent and challenging benchmark combining reasoning-intensive (e.g., math, physics) and knowledge-heavy (e.g., law, history) multiple-choice tasks.

We exclude multi-generation approaches such as semantic entropy (Kuhn et al., 2023), which are computationally expensive and capture question-level rather than response-level uncertainty, and internal state-based methods, which require access to hidden activations. Instead, we focus on three single-pass uncertainty estimation methods: (i) **Token Probability-Based Uncertainty (TPU)** (Lamb et al., 2025), which estimates uncertainty from token-level probabilities using the inverse of perplexity over the generated response and therefore requires log-probability access (gray-box), which some APIs do not provide; (ii) **Numerical Verbal Uncertainty (NVU)** (Xiong et al., 2024), where the model is explicitly prompted to produce a self-rated uncertainty score (e.g., between 0 and 100) alongside its answer; and (iii) **Linguistic Verbal Uncertainty (LVU)** (Yona et al., 2024), which infers uncertainty from hedging expressions (e.g., "probably", "might") embedded in the response, judged by a separate LLM. NVU and LVU are fully black-box, requiring only the generated text.

Our evaluation reveals several key findings. First, on MMLU-Pro, LVU outperforms TPU and NVU on average in both calibration (ECE) and discrimination (AUROC), though not for every

Table 1: Taxonomy of uncertainty estimation methods for LLMs.

Category	Representative Work	Black-Box	Single Gen.	Description
Token Probability-Based Uncertainty	Duan et al. (2024); Lamb et al. (2025); Burns et al. (2023)	✗	✓	Computes uncertainty from token-level probabilities, such as perplexity over the response.
Internal State-Based Uncertainty	Kossen et al. (2024); Ji et al. (2025); Kadavath et al. (2022)	✗	✓	Extracts uncertainty from internal activations like hidden states or attention weights.
Consistency-Based Uncertainty	Kuhn et al. (2023); Manakul et al. (2023); Farquhar et al. (2024)	✓	✗	Estimates uncertainty via disagreement across multiple sampled generations.
Numerical Verbal Uncertainty	Xiong et al. (2024); Tian et al. (2023)	✓	✓	Prompts the model to output a numeric score, e.g., "Uncertainty score: 0.35".
Linguistic Verbal Uncertainty	Yona et al. (2024); Belem et al. (2024); Ji et al. (2025)	✓	✓	Infers uncertainty from hedging language in the response (e.g., "probably", "might"), judged by another model.

model. Second, higher accuracy does not imply better uncertainty estimates: some high-accuracy models (e.g., GPT-4.1) are poorly calibrated, while others (e.g., Qwen3-235B-A22B) achieve reliable uncertainty estimates despite lower accuracy. Third, scale, post-training, and reasoning mode affect uncertainty performance, with reasoning mode notably reducing overconfidence, while quantization has only small, method-dependent effects; MoE models may offer advantages over dense ones, though this requires further validation. Lastly, uncertainty estimation is task-sensitive (more reliable on reasoning-oriented than knowledge-seeking subjects), and calibration and selective classification are only partially aligned, highlighting the need for multi-metric evaluation.

Our contributions can be summarized as follows:

1. **Comprehensive evaluation at scale:** We conduct, to our knowledge, the largest controlled comparison of single-pass uncertainty estimation methods, across 80 LLMs covering open- and closed-source models, dense and MoE architectures, and parameter scales ranging from 0.6B to 671B.
2. **Large-scale analysis of linguistic uncertainty:** We present an extensive evaluation of LVU, showing that on MMLU-Pro it outperforms TPU and NVU on average in both calibration and selective classification, making it a strong candidate for future work on uncertainty-aware LLMs.
3. **Empirical insights to guide future development:** Our analysis uncovers a range of actionable empirical findings such as the effect of reasoning mode, model scale, architecture, quantization and task type on uncertainty estimation, offering valuable guidance for future method design and theoretical analysis.

2 LLM Uncertainty Estimation and Evaluation

Table 1 summarizes the five main families of LLM uncertainty estimation methods introduced above; further related work is discussed in Appendix A.

2.1 LLM Uncertainty Estimation

In this work, we define *uncertainty* as a scalar in $[0, 1]$, where 0 indicates complete certainty and 1 indicates maximum uncertainty. Formally, we write $u_\phi(\mathbf{y} \mid \mathbf{x}) = 1 - p_\phi(\mathbf{y} \mid \mathbf{x})$, where $\mathbf{x}$ is the input, $\mathbf{y}$ is the generated output, and p_ϕ is the model's confidence. Our study focuses on *pointwise uncertainty*—the model's uncertainty about its actual response in a single generation—rather than

question-level uncertainty aggregated across multiple outputs. Thus, we adopt TPU, NVU, and LVU as our baseline methods for systematic evaluation.

Token Probability-Based Uncertainty (TPU) This method estimates uncertainty using the model's output token probabilities. Let $\mathcal{V}$ denote the vocabulary, and $\mathbf{x} \in \mathcal{V}^l$ be an input prompt of l tokens. The model generates a response $\mathbf{y} = (y_1, \ldots, y_n) \in \mathcal{V}^n$, where n is the number of tokens in the generated response. To normalize for sequence length, we compute the average autoregressive log-likelihood:

$$\bar{\ell}(\mathbf{y} \mid \mathbf{x}) = \frac{1}{n} \log p_\phi(\mathbf{y} \mid \mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \log p_\phi(y_i \mid \mathbf{y}_{<i}, \mathbf{x}), \quad \text{where } \mathbf{y}_{<i} = (y_1, \ldots, y_{i-1}). \tag{1}$$

Following Lamb et al. (2025), we define the *Token Probability-Based Uncertainty* as the complement of the exponentiated average log-likelihood of the generated response:

$$u_\phi^{\text{TPU}}(\mathbf{y} \mid \mathbf{x}) := 1 - \exp\left(\bar{\ell}(\mathbf{y} \mid \mathbf{x})\right) = 1 - p_\phi(\mathbf{y} \mid \mathbf{x})^{1/n} \in [0, 1], \tag{2}$$

where lower values indicate greater confidence. This method is broadly applicable—even to gray-box models—and serves as a strong baseline for uncertainty estimation.

Numerical Verbal Uncertainty (NVU) NVU elicits explicit uncertainty estimates directly from the LLM. The model is prompted to output a numeric uncertainty score alongside its response, offering an interpretable form of subjective uncertainty. We incorporate chain-of-thought (CoT) prompting to extract better uncertainty estimation following Xiong et al. (2024). The complete NVU prompting templates are provided in Appendix F.

Linguistic Verbal Uncertainty (LVU) In contrast to NVU, LVU captures uncertainty implicitly through hedging language embedded in the model's response—e.g., phrases like "probably", "might", or "possibly". Rather than providing a numeric score, the model expresses its uncertainty through natural language, aligning with how humans convey uncertainty in conversation. To quantify these implicit signals, we adopt the approach of Yona et al. (2024), using `LLaMA-4-Maverick-17B-128E-Instruct` as a separate evaluator to interpret and score linguistic uncertainty. This decouples generation from evaluation: the primary model produces a response, and an external model assesses the level of uncertainty conveyed linguistically. We provide an empirical analysis of the LLM judge's effectiveness in the Appendix H, along with details of the generation and evaluation prompt templates in Appendix F.

2.2 Evaluation of Uncertainty Estimation

To assess the quality of uncertainty estimation in large language models (LLMs), we consider two widely adopted evaluation tasks: *uncertainty calibration* and *uncertainty-based selective classification*. **Uncertainty calibration** measures how well predicted uncertainty values reflect the true likelihood of correctness, typically using the *Expected Calibration Error* (ECE) (Guo et al., 2017). In contrast, **selective classification** evaluates how well uncertainty scores distinguish correct from incorrect predictions, using the *Area Under the Receiver Operating Characteristic curve* (AUROC) (Mucsányi et al., 2024).

Uncertainty Calibration Uncertainty calibration evaluates whether a model's predicted uncertainty values correspond to actual correctness likelihoods. For example, among predictions with uncertainty 0.2 (i.e., 0.8 confidence), approximately 80% should be correct for a well-calibrated model. Formally, perfect calibration satisfies:

$$P(\hat{Y} = Y \mid U = u) = 1 - u, \quad \forall u \in [0, 1], \tag{3}$$

where $\hat{Y}$ is the model's prediction, Y is the ground truth, and U is the predicted uncertainty. In practice, calibration is quantified using *Expected Calibration Error (ECE)*, which partitions predictions into M uncertainty bins and computes the weighted average absolute difference between empirical accuracy and the implied certainty $(1 - \text{uncertainty})$:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} \left|\text{acc}(B_m) - (1 - \text{unc}(B_m))\right|, \tag{4}$$

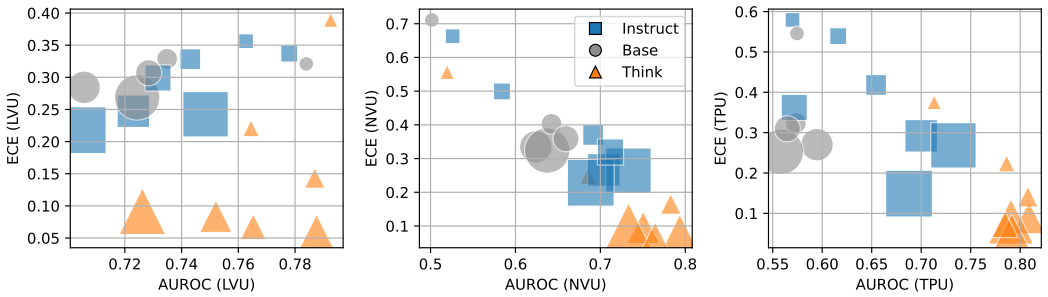

Figure 2: Uncertainty estimation performance of Qwen3 across three variants: no post-training (Base), with post-training (Instruct), and reasoning-enhanced (Think). Point size reflects model scale.

where $|B_m|$ is the number of predictions in bin m, n is the total number of predictions, and $\text{unc}(B_m)$ is the average uncertainty in bin m. Lower ECE indicates better calibration. We use $M = 10$ bins. We also report *adaptive ECE* (Nixon et al., 2019), which is robust to skewed uncertainty distributions, in Appendix D. Reliability diagrams (Niculescu-Mizil & Caruana, 2005) are also used for qualitative assessment.

Selective Classification Selective classification assesses the discriminative power of uncertainty scores in ranking correct versus incorrect predictions. The most common metric is AUROC, which measures the probability that a randomly chosen correct prediction is assigned lower uncertainty than a randomly chosen incorrect one. An AUROC of 0.5 corresponds to random guessing, while a value closer to 1.0 indicates stronger discrimination.

Models We evaluate 80 state-of-the-art LLMs, including both open- and closed-source models, with parameter scales ranging from 0.6B to 671B. Representative models include: **Open-source**—Meta LLaMA 3 (1B–405B) (Grattafiori et al., 2024), LLaMA 4 (109B, 400B) (AI, 2025), DeepSeek (Guo et al., 2025; Liu et al., 2024a), Qwen2.5/Qwen3 (0.6B–235B) (Yang et al., 2024; Qwen Team, 2025), Mistral (Jiang et al., 2023), and Google Gemma (Team et al., 2024b); **Closed-source**—OpenAI GPT-4.1/GPT-4o/o3/o4 (OpenAI, 2024; 2025a;b), Anthropic Claude 3 (Anthropic, 2024), Google Gemini (Team et al., 2024a), and xAI Grok (xAI, 2025). A complete model list is in Appendix I.

Dataset We conduct all evaluations on MMLU-Pro (Wang et al., 2024), a comprehensive benchmark released in June 2024 that includes both reasoning-centric tasks (e.g., math, physics, engineering) and knowledge-centric tasks (e.g., law, history, psychology). Compared to prior benchmarks, MMLU-Pro is more challenging and less saturated, making it a robust testbed for evaluating uncertainty estimation.

Prompt Design To enable consistent and fair uncertainty extraction across methods, we adopt a unified prompt template that combines numerical verbal uncertainty and linguistic verbal uncertainty formats (Xiong et al., 2024; Yona et al., 2024), with an additional concise CoT component. This design allows fair comparison across all models and uncertainty modalities. Detailed prompt formats are provided in Appendix F.

Uncertainty Extraction All models are prompted to produce answers in a predefined format. We use regular expressions to extract uncertainty scores. Outputs that deviate from the expected formats are excluded from analysis; since such failures may not occur at random, we discuss this choice as a limitation in Section 4. Extraction scripts and format rules are described in Appendix F.

3 Results and Discussion

3.1 Larger Models Generally Yield More Reliable Uncertainty Estimates

Model scale plays a critical role in shaping uncertainty estimation. While earlier studies in conventional deep learning suggest that larger models tend to be more overconfident and poorly cali-

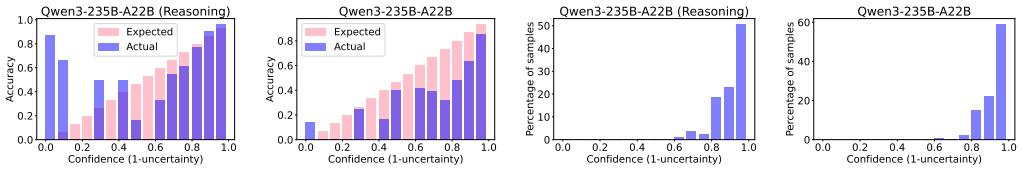

Figure 3: Reliability diagram and bin density between reasoning and non-reasoning LLMs.

brated (Guo et al., 2017), our findings in the LLM setting suggest otherwise. Overall, we observe that larger models tend to produce more calibrated and discriminative uncertainty scores. This trend is evident in Figure 1, especially under NVU and TPU.

To isolate the effect of model size, we examine the Qwen3 series trained under similar conditions, ranging from 0.6B to 32B parameters. As shown in Figure 2, larger models such as Qwen3-14B and Qwen3-32B consistently outperform their smaller counterparts in both calibration (lower ECE) and selective classification (higher AUROC). However, we also find that performance gains saturate at scale, and that mid-sized models like Qwen3-8B can achieve competitive results. In contrast, lightweight models such as Qwen3-0.6B and Qwen3-1.7B struggle to produce meaningful uncertainty scores, often approaching random guessing. This underscores the practical limitations of compact LLMs in scenarios that demand reliable uncertainty quantification.

Interestingly, under LVU, this trend appears partially reversed when evaluating AUROC. Upon further inspection, we find that smaller models tend to generate responses that poorly follow instructions or omit uncertainty cues altogether. The judge model tends to assign high LVU uncertainty to such non-compliant outputs, many of which are indeed incorrect. This results in strong discriminative uncertainty for smaller models—not because their uncertainty estimates are intrinsically better, but because instruction violations correlate strongly with failure.

3.2 Post-Training Enhances Uncertainty Estimation

Modern post-training techniques, including instruction fine-tuning and Direct Preference Optimization (DPO), are designed to align LLM outputs with human expectations and communication styles. Our results show that these techniques also substantially enhance uncertainty estimation. By comparing Qwen3-Base (no post-training) and Qwen3-Instruct (with post-training) models as shown in Figure 2, we observe consistent improvements across all three methods: post-trained models achieve lower ECE and higher AUROC, indicating both better calibration and stronger ability to distinguish between correct and incorrect predictions. This improvement is robust across model sizes. These findings suggest that post-training helps models not only generate more helpful outputs but also express uncertainty more reliably. For all other model series, we include only post-trained variants in our main evaluation to ensure fair comparisons. This aligns with and extends the findings of Zhu et al. (2023), and highlights post-training as an effective mechanism for enhancing LLM uncertainty awareness.

3.3 Reasoning Mode Improves Uncertainty Without Needing Higher Accuracy

Qwen3 supports both standard and reasoning-enhanced modes (denoted as 'Instruct' and 'Think'). Comparing these variants across model sizes (Figure 2), we find that reasoning mode consistently improves uncertainty estimation. For example, Qwen3-4B in reasoning mode matches or exceeds the uncertainty performance of much larger models like GPT-4.1, as shown in Figure 1. Figure 4 (left) shows reasoning-enhanced models have marked gains in calibration and modest improvements in selective classification, compared to non-reasoning models (averaged over all evaluated models).

To understand this effect in depth, we analyze reliability diagrams and bin density plots for representative models (Figure 3). Reasoning variants exhibit better alignment between predicted and actual correctness, particularly in high-confidence regions. Moreover, reasoning reduces the proportion of highly overconfident predictions by more than 20%, indicating its effectiveness in curbing model overconfidence. These results suggest that reasoning mode improves the model's ability to estimate uncertainty, particularly by curbing overconfidence.

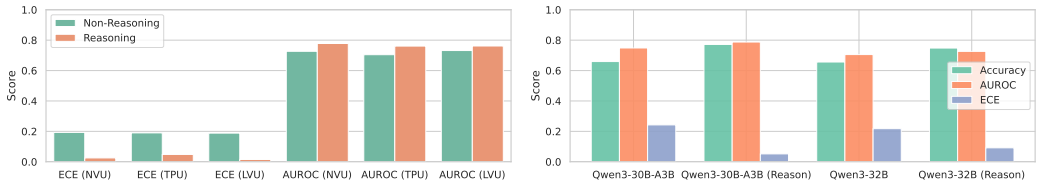

Figure 4: **Left:** Comparison of Reasoning vs. Non-Reasoning on ECE and AUROC. Results are averaged over all reasoning/non-reasoning models. **Right:** Comparison between MoE and dense models within the Qwen3 series under LVU.

3.4 MoE Models May Offer Improved Uncertainty Estimation over Dense Models

Mixture-of-Experts (MoE) has emerged as a compelling architectural paradigm for scaling LLMs efficiently, with recent adoption in models such as Qwen3-MoE, LLaMA-4, Grok-3, and DeepSeek. Unlike fully dense models, MoE models activate only a subset of parameters per input, potentially enabling more adaptive and specialized behavior. However, due to their recent emergence, head-to-head comparisons with dense counterparts remain limited.

One relatively fair comparison is between Qwen3-30B-A3B (MoE) and Qwen3-32B (dense), which share similar total parameter counts and follow closely matched training protocols (Yang et al., 2025). The Qwen3-30B-A3B model activates 3B parameters per forward pass, while Qwen3-32B is fully dense.

As shown in Figure 4 (right), Qwen3-30B-A3B outperforms its dense counterpart in both reasoning and non-reasoning tasks when evaluated using LVU, achieving better calibration and more reliable selective classification. This result suggests that the conditional computation of MoE may facilitate more nuanced uncertainty estimation. Nevertheless, the sample size of comparable MoE–dense model pairs remains small, and further systematic comparisons are needed to confirm whether MoE architectures generally confer an advantage in uncertainty estimation.

3.5 High Accuracy Does Not Guarantee Reliable Uncertainty Estimation

While recent advancements in LLM development have predominantly emphasized improving accuracy, our results reveal that high accuracy does not necessarily imply reliable uncertainty estimation. As shown in Figure 5 left, several top-performing models—including LLaMA-4-Maverick, LLaMA-3.1-405B, and GPT-4.1—exhibit strong accuracy but poor uncertainty estimation under LVU, as indicated by elevated ECE and reduced AUROC scores. We observe similar discrepancies under NVU and TPU, detailed in the Appendix E.

Notably, `Qwen3-235B-A22B (Reasoning)` achieves only moderate accuracy (67%)—about 15 points lower than models such as `DeepSeek-R1`, `GPT-4.1`, or `Grok-3`—yet consistently delivers top-tier uncertainty performance across all three methods. This suggests that reliable uncertainty estimation is not merely a byproduct of better accuracy, but a distinct capability shaped by model architecture, training, and prompting.

3.6 Linguistic Verbal Uncertainty Shows High Potential for Reliable Uncertainty Estimation

To compare the overall effectiveness of different uncertainty estimation methods, we compute average AUROC and ECE scores across all 80 evaluated models. As shown in Figure 5 right, LVU outperforms both NVU and TPU on average, with roughly 10% improvement in both AUROC and ECE over the second-best methods. Because TPU requires token log-probabilities, it is unavailable for models whose APIs do not expose them (marked "/" in Appendix I), so TPU averages are computed over the subset of models with TPU scores. The advantage of LVU is not universal: per-model results (Appendix I) show that it attains the lowest ECE for 54 models and the highest AUROC for 64 models, i.e., for most, but not all, of them. LVU also communicates uncertainty through hedging language such as "probably" and "might," which is natural for users; however, hedges are perceived differently

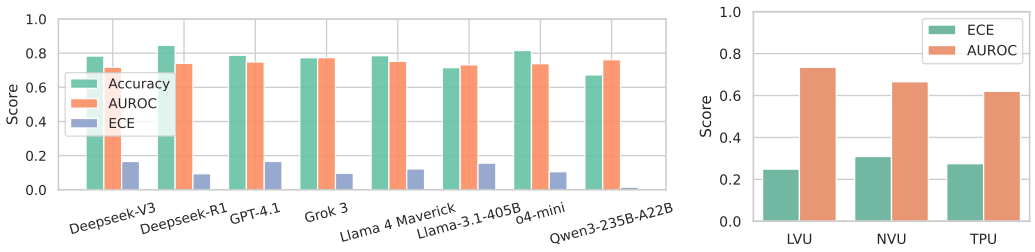

Figure 5: **Left:** Comparison of top-tier models under LVU. **Right:** Comparison of different uncertainty estimation methods.

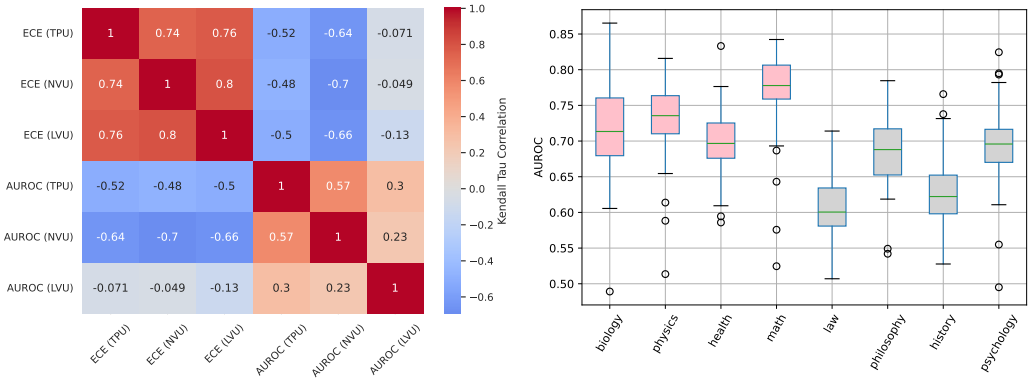

Figure 6: **Left:** Kendall Rank Correlation among Uncertainty Estimation Methods. **Right:** LLMs exhibit better uncertainty estimates in reasoning tasks, as measured by AUROC using LVU.

across readers (Belem et al., 2024), so this form of uncertainty is less precise than a numeric score, and LVU requires an additional call to a judge model. We therefore view LVU as a promising but task- and cost-dependent choice rather than a universally superior method.

To better understand their relationship, we compute Kendall rank correlations (Abdi, 2007) between ECE and AUROC across the three uncertainty estimation methods. As shown in Figure 6 left, we observe high internal consistency within each metric—particularly between LVU and NVU on ECE—indicating agreement in how well each method captures calibrated uncertainty. However, the correlation between AUROC values across methods is notably weaker (τ between 0.23 and 0.57), suggesting that different approaches capture distinct signals relevant to ranking performance. We also include the comparisons with multi-generation based uncertainty (Kuhn et al., 2023; Farquhar et al., 2024; Manakul et al., 2023) in the Appendix B for a comprehensive understanding.

3.7 Calibration and Selective Classification Capture Distinct Dimensions of Uncertainty

Figure 6 (left) shows that the two metrics are only partially aligned. Within the same method, ECE and AUROC are moderately negatively correlated for NVU ($\tau = -0.70$) and TPU ($\tau = -0.52$)—as expected, since lower ECE and higher AUROC both indicate better uncertainty—but nearly uncorrelated for LVU ($\tau = -0.13$). Thus, although better-calibrated models tend to rank errors better under NVU and TPU, calibration is far from a substitute for discrimination, and for LVU the two are largely independent. These findings reinforce the need for multi-metric evaluation in uncertainty estimation, as each metric provides a complementary view of uncertainty reliability. In practice, selecting a model or method by ECE alone may overlook poor error ranking, which matters most for abstention and human-review workflows; we therefore recommend reporting ECE and AUROC together with reliability diagrams.

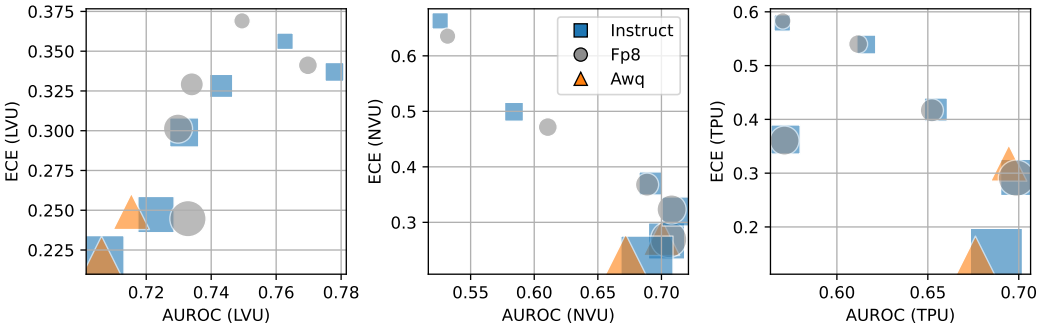

Figure 7: Uncertainty estimation performance of Qwen3 across quantization variants: no quantization (Instruct), FP8, and Activation-aware Weight Quantization (Awq). Point size reflects model scale.

3.8 LLMs Estimate Uncertainty More Reliably on Reasoning Tasks

We further investigate how uncertainty estimation varies across task types. MMLU-Pro offers a diverse set of subjects, covering both reasoning-intensive domains (e.g., math, physics) and knowledge-seeking ones (e.g., law, history). To compare model performance across these categories, we report AUROC scores for 80 models using LVU, as shown in Figure 6 right, where reasoning-related subjects are marked in red and knowledge-heavy subjects in light gray.

We group reasoning-intensive tasks as *math, biology, health*, and *physics*, and knowledge-seeking tasks as *law, philosophy, history*, and *psychology*. Our analysis reveals that models estimate uncertainty more reliably on reasoning tasks, with an average AUROC improvement of over 10% compared to knowledge-based tasks. We observe the same direction for all three uncertainty estimation methods, suggesting that the structured nature of reasoning problems may allow LLMs to form more consistent and discriminative uncertainty estimates. Conversely, uncertainty estimates in knowledge-heavy domains such as law, where errors can be costly, should be treated with extra caution.

3.9 Quantization Has Small, Method-Dependent Effects on Uncertainty Estimation

Quantization is widely used to deploy LLMs in resource-constrained environments. We analyze quantized variants (FP8 and AWQ) of the Qwen3 series. As shown in Figure 7, quantized and full-precision variants generally lie close to each other, and we do not observe a consistent degradation trend. Under LVU and TPU, quantized variants are often slightly worse, whereas under NVU, quantization yields marginal improvements in small models and small drops at larger scales; for some model–method pairs, metrics even improve after quantization. Overall, the impact is modest, typically within a 5% margin, suggesting that FP8 and AWQ quantization can be adopted in uncertainty-sensitive deployments without a substantial loss in uncertainty quality.

4 Conclusion and Future Work

We benchmarked TPU, NVU, and LVU across 80 LLMs on MMLU-Pro. LVU performs best on average, accuracy is not a reliable proxy for uncertainty quality, and calibration and discrimination are only partially aligned. Scale, post-training, and reasoning mode improve uncertainty estimation, whereas quantization has only small effects. Future work could leverage LVU to reduce user over-reliance (Kim et al., 2024) and develop objectives that jointly optimize calibration and discrimination.

Limitations Our conclusions are drawn mainly from multiple-choice evaluation on MMLU-Pro and may not transfer to other tasks. LVU relies on a single external judge, which is less precise for negative hedges (Appendix H), and an extra inference call; and outputs that violate the required format are excluded, which may bias results for weaker models.

Ethics Statement

This work evaluates publicly available LLMs on a public benchmark (MMLU-Pro) and involves no human subjects or private data. Our goal is to improve the reliability of LLM deployment; we note that uncertainty estimates can be miscalibrated, and should complement rather than replace human oversight in high-stakes settings.

Reproducibility Statement

The evaluated models are listed with full results in Appendix I. Prompt templates, the LVU judge prompts, and the regular expressions used for answer and confidence extraction are given in Appendix F; metric definitions are given in Section 2.2. Code and resources are available through an anonymous repository (Appendix C).

AI Use Statement

We used generative AI tools for editing the writing and for LaTeX formatting. LLMs are also part of our methodology: they are the models under evaluation, and `LLaMA-4-Maverick` serves as the judge for LVU, as described in Section 2. We have reviewed all AI-assisted work and take responsibility for the final content of this paper.

References

Asma Ben Abacha, Wen-wai Yim, Yujuan Fu, Zhaoyi Sun, Meliha Yetisgen, Fei Xia, and Thomas Lin. MEDEC: A benchmark for medical error detection and correction in clinical notes. *arXiv preprint arXiv:2412.19260*, 2024.

Hervé Abdi. The Kendall rank correlation coefficient. *Encyclopedia of measurement and statistics*, 2: 508–510, 2007.

Meta AI. Llama 4: Multimodal intelligence. `https://ai.meta.com/blog/llama-4-multimodal-intelligence/`, April 2025. Accessed: 2025-05-10.

Anthropic. Introducing the next generation of Claude. `https://www.anthropic.com/news/claude-3-family`, 2024. Accessed: 2025-05-12.

Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, 2023.

BBC News. ChatGPT: US lawyer admits using AI for case research. `https://www.bbc.com/news/world-us-canada-65735769`, May 2023. Published 27 May 2023. Accessed: 2025-05-10.

Catarina G Belem, Markelle Kelly, Mark Steyvers, Sameer Singh, and Padhraic Smyth. Perceptions of linguistic uncertainty by language models and humans. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations*, 2023.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 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)*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Muhammad Usman Hadi, Rizwan Qureshi, Abbas Shah, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, Seyedali Mirjalili, et al. Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects. *Authorea Preprints*, 1:1–26, 2023.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

Hsiu-Yuan Huang, Yutong Yang, Zhaoxi Zhang, Sanwoo Lee, and Yunfang Wu. A survey of uncertainty estimation in LLMs: Theory meets practice. *arXiv preprint arXiv:2410.15326*, 2024.

ISACA. Understanding the EU AI Act: Requirements and next steps. `https://www.isaca.org/resources/white-papers/2024/understanding-the-eu-ai-act`, October 2024. Accessed: 2025-05-12.

Ziwei Ji, Lei Yu, Yeskendir Koishekenov, Yejin Bang, Anthony Hartshorn, Alan Schelten, Cheng Zhang, Pascale Fung, and Nicola Cancedda. Calibrating verbal uncertainty as a linear feature to reduce hallucinations. *arXiv preprint arXiv:2503.14477*, 2025.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL `https://aclanthology.org/P17-1147/`.

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

Sunnie SY Kim, Q Vera Liao, Mihaela Vorvoreanu, Stephanie Ballard, and Jennifer Wortman Vaughan. "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*, pp. 822–835, 2024.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. Semantic entropy probes: Robust and cheap hallucination detection in LLMs. *arXiv preprint arXiv:2406.15927*, 2024.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations*, 2023.

Jinqi Lai, Wensheng Gan, Jiayang Wu, Zhenlian Qi, and Philip S Yu. Large language models in law: A survey. *AI Open*, 2024.

Tom A Lamb, Desi R Ivanova, Philip Torr, and Tim GJ Rudner. Semantic calibration of LLMs through the lens of temperature scaling. In *ICLR Workshop: Quantify Uncertainty and Hallucination in Foundation Models: The Next Frontier in Reliable AI*, 2025.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, 2023.

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

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*, 2024a.

Linyu Liu, Yu Pan, Xiaocheng Li, and Guanting Chen. Uncertainty estimation and quantification for LLMs: A simple supervised approach. *arXiv preprint arXiv:2404.15993*, 2024b.

Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. Uncertainty quantification and confidence calibration in large language models: A survey. *arXiv preprint arXiv:2503.15850*, 2025.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Bálint Mucsányi, Michael Kirchhof, and Seong Joon Oh. Benchmarking uncertainty disentanglement: Specialized uncertainties for specialized tasks. *Advances in neural information processing systems*, 37:50972–51038, 2024.

Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pp. 625–632, 2005.

Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. Measuring calibration in deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

OpenAI. Hello GPT-4o. `https://openai.com/index/hello-gpt-4o/`, 2024. Accessed: 2025-05-12.

OpenAI. Introducing GPT-4.1 in the API. `https://openai.com/index/gpt-4-1/`, 2025a. Accessed: 2025-05-12.

OpenAI. Introducing OpenAI o3 and o4-mini. `https://openai.com/index/introducing-o3-and-o4-mini/`, 2025b. Accessed: 2025-05-12.

Samir Passi and Mihaela Vorvoreanu. Overreliance on AI: Literature review. Technical report, Microsoft Research, 2022.

Qwen Team. Qwen3: Think deeper, act faster. `https://qwenlm.github.io/blog/qwen3/`, 2025. Accessed: 2025-05-12.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024a.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on Gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024b.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. Large language models in medicine. *Nature medicine*, 29(8): 1930–1940, 2023.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 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*, 2023.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerová, et al. Adapted large language models can outperform medical experts in clinical text summarization. *Nature medicine*, 30(4):1134–1142, 2024.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eighth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

xAI. Grok 3 beta — the age of reasoning agents. `https://x.ai/news/grok-3`, 2025. Accessed: 2025-05-12.

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

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Gal Yona, Roee Aharoni, and Mor Geva. Can large language models faithfully express their intrinsic uncertainty in words? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024.

Chiwei Zhu, Benfeng Xu, Quan Wang, Yongdong Zhang, and Zhendong Mao. On the calibration of large language models and alignment. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

Younan Zhu, Linwei Tao, Minjing Dong, and Chang Xu. Mitigating object hallucinations in large vision-language models via attention calibration. *arXiv preprint arXiv:2502.01969*, 2025.

A Related Work

A.1 More Uncertainty Estimation Approaches

Multi-Generation Consistency-Based Methods Recent advancements in uncertainty estimation for large language models have introduced several methods leveraging multi-generation consistency. SelfCheckGPT (Manakul et al., 2023) identifies hallucinations by generating multiple responses to the same prompt and assessing their consistency using metrics like BERTScore, n-gram overlap, and natural language inference (NLI). Inconsistencies among responses suggest potential hallucinations. Generating with Confidence (Lin et al., 2024) proposes evaluating the variability among multiple generated outputs to estimate uncertainty. Semantic Entropy (Kuhn et al., 2023; Farquhar et al., 2024) addresses the challenge of semantic equivalence in natural language by clustering semantically similar responses and computing entropy over these clusters, providing a more accurate measure of uncertainty for open-ended questions.

Internal State-Based Uncertainty Recent research has also explored leveraging the internal states of LLMs to estimate uncertainty and detect hallucinations. Azaria & Mitchell (2023) demonstrate that hidden layer activations can be utilized to predict the truthfulness of generated statements by training a classifier on these activations. Li et al. (2023) introduce Inference-Time Intervention (ITI), a technique that adjusts model activations during inference to enhance truthfulness, significantly improving performance on benchmarks like TruthfulQA. Kossen et al. (2024) propose Semantic Entropy Probes (SEPs), which estimate semantic uncertainty directly from hidden states without requiring multiple generations, offering a computationally efficient method for hallucination detection. Ji et al. (2025) identify a linear feature within the model's representation space that governs verbal uncertainty, enabling calibration to reduce overconfident hallucinations. Burns et al. (2023) present an unsupervised approach to uncover latent knowledge by identifying directions in activation space that correspond to logical consistency, allowing accurate yes-no question answering without labeled data. Lastly, Liu et al. (2024b) propose a supervised method that leverages labeled datasets to estimate uncertainty from hidden activations, demonstrating robust performance across various tasks and model access levels.

A.2 Study on Uncertainty Estimation for LLMs

Recent research has delved into the challenges and methodologies of uncertainty estimation in LLMs. Geng et al. (2024) provide a comprehensive survey on confidence estimation and calibration techniques. Huang et al. (2024) offer a hierarchical categorization of uncertainty estimation methods and drawing relations with traditional machine learning approaches. Zhu et al. (2023) examine token probability-based uncertainty, analyzing the effects of model size and instruction tuning, though they focus on a single estimation method and do not include modern LLM design elements like Direct Preference Optimization (DPO) and Mixture-of-Experts (MoE) architectures. Liu et al. (2025) present a recent survey exploring the sources of uncertainty in LLMs and methods for uncertainty quantification and confidence calibration. Further, Mucsányi et al. (2024) benchmark a broad set of uncertainty estimators across multiple uncertainty tasks, showing that different tasks call for different estimators. Studies by Yona et al. (2024) and Xiong et al. (2024) investigate the expression of uncertainty in LLMs, focusing on linguistic verbal uncertainty and numerical verbal confidence, respectively, under specific conditions.

B Comparison with Multi-Generation Based Uncertainty

We primarily compare two multi-generation based uncertainty estimation methods: (1) *consistency* (Kuhn et al., 2023; Manakul et al., 2023; Yona et al., 2024), which measures the rate of contradiction between the anchor response and a set of sampled responses; and (2) *majority vote* (Farquhar et al., 2024), which estimates prediction confidence as the proportion of sampled responses that agree with the majority answer.

Since our evaluation is conducted under a multiple-choice setting, we focus only on the predicted label from each response, disregarding its full textual content.

Consistency Formally, given a question Q, a generated response $\hat{R}$, and its corresponding predicted answer $\hat{A}$, we sample k additional responses $\{R_1, \ldots, R_k\}$ and extract their respective predictions $\{A_1, \ldots, A_k\}$. The model's uncertainty in $\hat{A}$ is then quantified as the fraction of sampled predictions that contradict $\hat{A}$:

$$\text{unc}(\hat{A}) \equiv \frac{1}{k} \sum_{i=1}^{k} \mathbf{1}[\hat{A} \text{ contradicts } A_i] \tag{5}$$

To ensure a fair comparison, we use the same response $\hat{R}$ as those employed in LVU, NVU, and TPU, and set $k = 10$.

Majority Vote As an additional baseline, we adopt a majority vote strategy following (Farquhar et al., 2024). In our multiple-choice setting, we omit the semantic clustering step and define the predicted label as the most frequent answer among the k sampled responses. The proportion of responses that agree with this majority label is treated as the model's confidence.

Note that for models that fail to follow instructions (i.e., do not produce a valid prediction), we exclude the corresponding instances from evaluation.

We conduct experiments on the Qwen3 model series, ranging from 0.6B to 32B parameters, post-trained models (with instruction fine-tuning and DPO) and quantized variants (FP8). As shown in Figure 8, we compare post-trained Qwen3 series using multi-generation-based methods (Consistency and Majority Vote) and single-generation methods (LVU, NVU, and TPU). Consistency achieves the best calibration at every model size, with an ECE as low as 0.08 at the 32B scale. For AUROC, the picture depends on scale: LVU is better for small models (0.6B–4B), whereas Consistency is better from 8B onward. Multi-generation methods, however, require k additional generations per question.

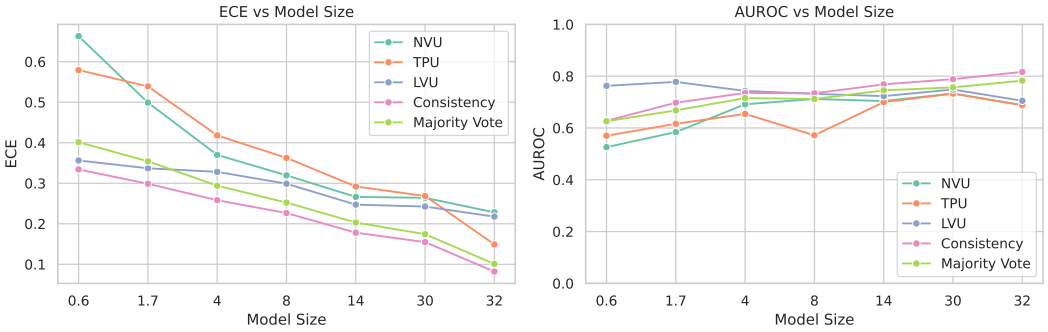

Figure 8: Comparison with multi-generation based uncertainty estimation.

Full evaluation results on multi-generation based uncertainty are provided in Appendix B.1.

B.1 FULL EVALUATION RESULTS OF MULTI-GENERATION BASED UNCERTAINTY

Table 2: Multi-generation based uncertainty estimation evaluation for Qwen3 series.

Model	Accuracy	Accuracy (Maj. Vote)	ECE					AUROC				
			NVU	TPU	LVU	Consi.	Maj. Vote	NVU	TPU	LVU	Consi.	Maj. Vote
Qwen3-0.6b	0.248	0.250	0.663	0.579	0.356	0.334	0.401	0.526	0.570	0.763	0.628	0.626
Qwen3-1.7b	0.399	0.418	0.499	0.539	0.337	0.299	0.354	0.584	0.616	0.778	0.698	0.668
Qwen3-4b	0.529	0.544	0.370	0.418	0.328	0.258	0.294	0.691	0.654	0.743	0.735	0.716
Qwen3-8b	0.587	0.602	0.320	0.363	0.299	0.226	0.252	0.712	0.572	0.732	0.734	0.712
Qwen3-14b	0.632	0.649	0.266	0.292	0.247	0.178	0.203	0.704	0.700	0.723	0.768	0.745
Qwen3-30b-a3b	0.662	0.686	0.264	0.269	0.243	0.155	0.174	0.733	0.733	0.748	0.789	0.757
Qwen3-32b	0.697	0.723	0.228	0.149	0.218	0.082	0.101	0.689	0.688	0.705	0.817	0.783
Qwen3-0.6b-fp8	0.250	0.252	0.635	0.583	0.369	0.311	0.398	0.532	0.570	0.749	0.631	0.627
Qwen3-1.7b-fp8	0.395	0.416	0.472	0.540	0.341	0.277	0.349	0.611	0.612	0.770	0.704	0.671
Qwen3-4b-fp8	0.530	0.551	0.368	0.417	0.329	0.243	0.284	0.689	0.652	0.734	0.736	0.705
Qwen3-8b-fp8	0.588	0.613	0.323	0.361	0.301	0.208	0.238	0.708	0.571	0.730	0.750	0.714
Qwen3-14b-fp8	0.630	0.649	0.269	0.291	0.245	0.172	0.199	0.706	0.699	0.733	0.768	0.741

C CODE AND RESOURCES

Our code and resources are anonymously available at: `https://anonymous.4open.science/r/LLM-Calibration-Study-203D/`

D EVALUATION WITH ADAPTIVE ECE

We also adopt Adaptive-ECE (Nixon et al., 2019) as an alternative metric to evaluate calibration performance. The results of Adaptive-ECE closely align with those of ECE, demonstrating the robustness of our calibration evaluation using ECE across different binning strategies in this work.

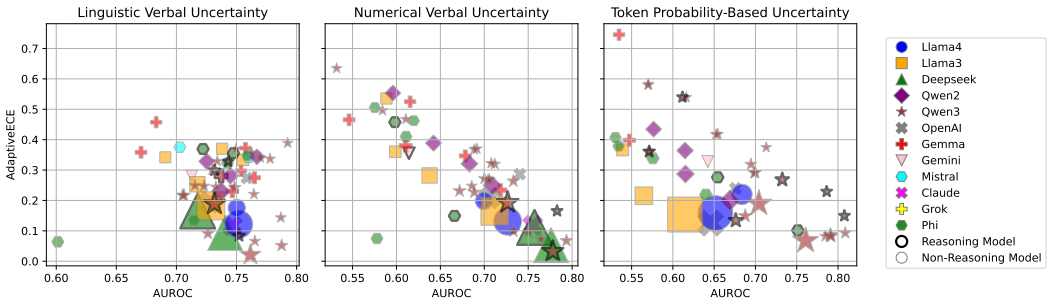

Figure 9: AdaptiveECE vs AUROC across all methods.

E HIGH ACCURACY DOES NOT GUARANTEE RELIABLE UNCERTAINTY ESTIMATION

As discussed in Section 3.5, our experiments using LVU reveal that high accuracy does not necessarily imply reliable uncertainty estimation. We observe similar trends under NVU and TPU, as illustrated in Figures 10 and 11. Notably, although `Qwen3-235B-A22B` has the lowest accuracy among these top-tier models, it achieves the lowest ECE and an AUROC comparable to the best models across all three uncertainty estimation methods. Note that the DeepSeek series does not provide valid token-level probabilities; hence, its AUROC and ECE values under TPU are omitted due to invalidity.

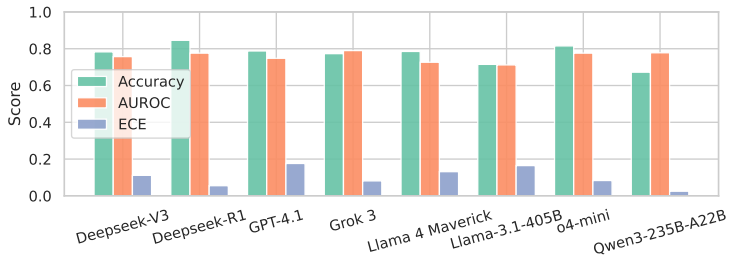

Figure 10: Comparison of top-tier models under NVU.

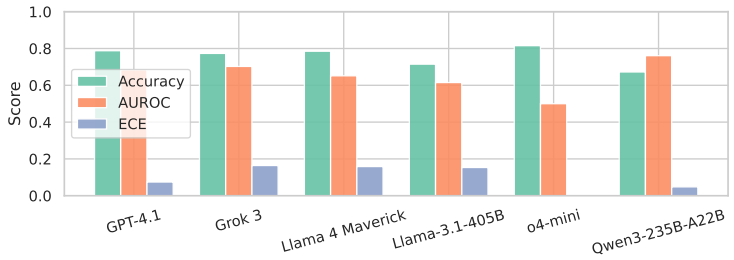

Figure 11: Comparison of top-tier models under TPU.

F PROMPTS AND REGULAR EXPRESSIONS

F.1 SHARED PROMPT FOR MMLU-PRO RESPONSE SAMPLING

The following prompt is used to sample responses, incorporating NVU and LVU elicitation as well as a concise CoT, where `{Letters}` and `{Question}` are placeholders. A single response from this prompt is shared by all uncertainty methods.

Shared prompt for MMLU-Pro response sampling

Read the question, analyze step by step, provide your answer and your confidence in this answer. Note: The confidence indicates how likely you think your answer is true.

If you are uncertain about your answer to the question, convey this uncertainty linguistically by precisely hedging this answer. Use the following format to answer:

Explanation: [insert short step-by-step analysis here]
Answer: [ONLY the option LETTER, where the LETTER is one of `{Letters}`]
Confidence (0-100): [Your confidence level, please only include the numerical number in the range of 0-100]%

Only give me the reply according to this format, don't give me any other words.

`{Question}`

F.2 Prompts for Linguistic Uncertainty (LVU) Judge

The following details the few-shot prompt used to instruct the LVU judge, where `{Question}` and `{Response}` are placeholders. The response sampled from the shared prompt has been post-processed to remove numerical verbal confidence to avoid leading or biased LVU judgment.

Prompt for the LVU judge

You will be given a question and a proposed answer generated by a language model. Your task is to output an confidence score: a number in [0,1] representing how certain the language model is about the truthfulness of its generated answer, judged solely based on the provided answer. You should pay special attention to the usage of any hedging modifiers in the original answer, used to convey uncertainty in the truthfulness of the answer. A score close to 1.0 indicates that the model is very certain about the truthfulness answer. And a score close to 0.0 means that the model is highly uncertain, and the answer likely contains some hedging expressions. If the proposed answer punts the question, the confidence score should be 0.0.

Here are some examples:

Question: criminal minds episode where jj becomes a profiler
Proposed answer: I think JJ officially becomes a profiler in Criminal Minds season 6, episode 2.
Confidence score: 0.8

Question: criminal minds episode where jj becomes a profiler
Proposed answer: JJ officially becomes a profiler in Criminal Minds season 6, in either the first or second episode.
Confidence score: 0.5

Question: criminal minds episode where jj becomes a profiler
Proposed answer: I'm not really sure about this, but I think the episode in which JJ officially becomes a profiler in Criminal Minds may be episode 2 in season 6.
Confidence score: 0.6

Question: criminal minds episode where jj becomes a profiler
Proposed answer: I don't know which episode you're referring to.
Confidence score: 0.0

Here are the question and proposed answer to be judged:
Question:
`{Question}`

Proposed Answer:
`{Response}`

Provide your response only in the following format:
Confidence score: [confidence score (0-1)].

F.3 Answer and Numerical Verbal Confidence Extraction Regular Expressions

The following presents the regex collection used to extract the answer from the LLM responses.

Multiple-choice answer extraction regex

```
r"[Aa]nswer:?[\s]*[\n]*([A-J])",
r"[Aa]nswer:[\s]*[\n]*\(?([A-J])\)?",
r"[Aa]nswer:[\s]*[\n]*\[?([A-J])\]?",
r"[Aa]nswer:[\s]*[\n]*([A-J])[,)]",
r"[Aa]nswer:[\s]*[\n]*([A-J])\s*,?.*",
r"Answer:\n([A-J])\nConfidence",
r"answer is\s*\[?\(?([A-J])\]?\)?",
r"answer should be\s*\[?\(?([A-J])\]?\)?",
r"best option is \(?([A-J])\)?",
r"best match is option \(?([A-J])\)?",
r"the closest is \(?([A-J])\)?",
r"Answer:\n*^([A-J])$",
r"^([A-J])$"
```

The following regex collection is used to extract and remove Numerical Verbal Confidence from the responses.

Numerical Verbal Confidence extraction and removal regex

```
r"[Cc]onfidence\s*\(0-100\):\s*[\(]?[\[]?(\d+)[\)]?[\]]?%?",
r"[Cc]onfidence[:]?\s*(\d+)%?",
r"[Cc]onfidence [\(0-100\)]?:\s*\[(\d+)%?\]"
r"[Cc]onfidence [Ll]evel\s*\(0-100\):\s*(\d+)%?",
r"[Cc]onfidence [Ll]evel[:]?\s*(\d+)%?",
r"[Cc]onfidence [Ll]evel[\(0-100\)]?:\s*\[(\d+)%?\]",
r"[Cc]onfidence \(100\):\s*\w*,\s*(\d+)%?",
r"[Cc]onfidence\s*\(\d+\)\s*:\s*(\d+)%?",
r"[Cc]onfidence\s*[\(]?(\d+)[\)]?%?"
```

Any output falling outside the predefined regex patterns is treated as non-response, even if the underlying content may be partially or fully accurate. As a result, correct answers and successful confidence reporting are contingent upon both factual correctness and adherence to basic instruction-following behavior.

Our observations suggest that answer and confidence extraction failures are mainly caused by poor instruction following, with the following typical behaviors:

- Implicit answers: the content of the response is correct, and yet the answer is not explicitly stated.
- Maximum token length exceeded: especially evident in reasoning models, the models have not finished outputting their chain of thought before hitting the maximum token length. i.e. the answer and/or confidence have not yet been provided.
- Non-response error: due to model internal issues, null responses are produced arbitrarily.
- General regex mismatches: the answer output does not adhere to the allowed regex formats.

For subsequent analyses, all successful confidence extractions will be rescaled between 0 - 1. All failed extractions will be dropped. On average, approximately 17% of the responses are dropped due to the answer, numerical confidence and/or linguistic confidence extraction failure.

G Efficient Confidence Evaluation: Stratified Sampling on MMLU-Pro

Since instruction following can be challenging for less-capable models such as Qwen3-0.6B, we remove instances where the model fails to adhere to the instructions or produces nonsensical responses. For example, Qwen3-0.6B has up to 5,000 such invalid responses excluded from evaluation. To ensure that our confidence estimation remains reliable despite the reduced number of valid responses, we conduct additional experiments demonstrating that the evaluation performance remains consistent even with fewer samples.

Although MMLU-Pro comprises a substantial total of 12,032 questions spanning a diverse range of academic disciplines, we are interested in the possibility of evaluating confidence on a subset to reduce computational overhead without significantly compromising the reliability of the results.

To investigate this, we employ GPT-4.1-Mini to generate responses for the entire MMLU-Pro dataset and construct stratified samples at proportions from 10% to 90%. Each sample is generated through stratified sampling, ensuring an equal number of questions from each field of study. For each sampling proportion, we repeat the procedure 50 times to account for sampling variability and to obtain robust performance estimates. The resulting distributions of ECE and AUROC are presented in Figure 12. The results demonstrate that stratified samples comprising at least 50% of the dataset produce ECE and AUROC distributions that are closely aligned with those of the full dataset, suggesting that stratified subsets can be used for efficient calibration analysis.

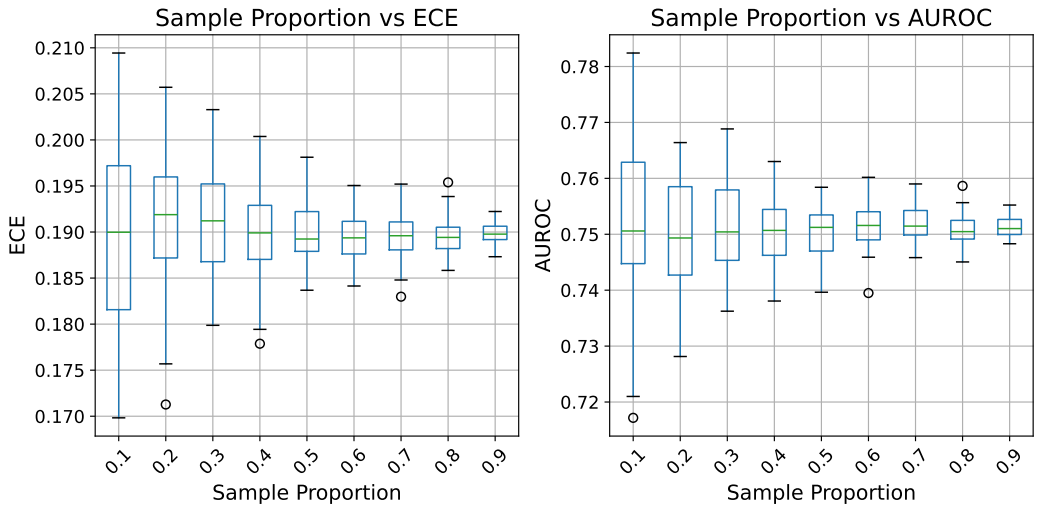

Figure 12: **GPT-4.1-Mini's spread of confidence distributions from different stratified subsets sizes.** Each box represents 50 stratified samples. The AUROC and ECE distributions start to stabilize and trend towards those of the full set when the sample proportion exceeds 50%.

H LLM AS A JUDGE FOR LVU

We employ LLaMA-4-Maverick-17B-128E-Instruct (AI, 2025) as the judge for LVU in all evaluations. We replicate the experiment carried out by Belem et al. (2024) to explore the judge's mapping between various hedging words and their perceived numerical confidence (note that confidence and uncertainty are complementary, i.e., uncertainty $= 1 -$ confidence). The experiment involves evaluating a collection of sentences, each of which consists of a single hedging word, and outputting a corresponding probability of truth ascertained by the model. There are 20 non-verifiable statements on daily life events, where non-verifiable statements are statements for which there is insufficient contextual information for external observers to confidently believe whether they are true or false. This is designed to linguistically assess the model's sensitivity to hedging words rather than its knowledge.

H.1 SENTENCE CONSTRUCTION

Each sentence has the following structure:

Sentence structure

`[NAME]` believes it is `[HEDGING WORD]` that `[STATEMENT]`

where the name, hedging word, and statement are sourced from the following. The judge then evaluates all the possible combinations of names, hedging words, and statements.

Possible options for [NAME]

"Brendan", "Bruce", "David", "Gary", "Isaac", "Jeffery", "Joey", "Johnnie", "Kenny", "Lance", "Marco", "Mike", "Nathan", "Nick", "Raul", "Amanda", "Bonnie", "Camille", "Catherine", "Cheri", "Ethel", "Gabriela", "Jacquelyn", "Jessica", "Laura", "Olga", "Roxanne", "Silvia", "Tara", "Violet"

Possible options for [HEDGING WORD]

"almost certain", "highly likely", "very likely", "probable", "somewhat likely", "possible", "uncertain", "somewhat unlikely", "unlikely", "not likely", "doubtful", "very unlikely", "highly unlikely"

Possible options for [STATEMENT]

"they will buy a new watch this Thanksgiving weekend.",
"they will be offered a promotion this fall.",
"the company will have another round of layoffs by mid-July.",
"there will be vegetarian options at the barbecue.",
"they will visit New York over winter break.",
"the restaurant near their apartment accepts reservations.",
"the new museum is offering complimentary admission.",
"there is a yoga studio within 2 miles of their workplace.",
"there are more than eighty students in the auditorium right now.",
"the temperature in the office is at least 72 degrees Fahrenheit.",
"their boss owns a blue car.",
"their friend has a leather jacket.",
"their cousin has a vegetable garden.",
"their classmate owns a guitar.",
"their boss has a stereo amplifier.",
"their cousin prefers spinach over broccoli.",
"their boss prefers coffee over tea.",
"their friend prefers running over cycling.",
"their neighbor prefers the beach over the mountains.",
"their coworker prefers reading books over watching movies."

H.2 Prompt Construction and Uncertainty Value Extraction

The following prompt is used to elicit the judge to output its perceived uncertainty with the presence of a hedging word in the sentence.

Prompt to elicit LLMs' perceived confidence

```
[SENTENCE]
```

From their perspective, considering the use of hedging words, what is the probability (from 0 to 100) that the statement is true? Concisely provide your response only in the following format:

Probability: [0-100]%

The probability is then extracted using the following regular expression:

Regex to extract perceived confidence

```
r"\b(?:Probability|Prob(?:\.|ability)?|P)\s*[:=]?\s*(\d+)%"
```

H.3 Results

The prompt and regex above successfully extract a probability for 99.85% of the sentences, indicating that the judge follows the required format reliably.

As shown in Figure 13, the judge maps positive hedges consistently and in the expected order: "almost certain" receives the highest confidence, followed by "highly likely" and "very likely" (which are not distinguished from each other) and "probable". Negative hedges such as "unlikely", "very unlikely", and "highly unlikely" receive lower median confidence, but their scores vary widely across statements, and some mid-range expressions are not ordered as expected (e.g., "doubtful" is rated higher than "somewhat unlikely"). The judge therefore captures the coarse gradation of hedging language, especially for confident expressions, but is less precise for negative and intermediate hedges, which

is consistent with the variability observed in human interpretations of hedging expressions (Belem et al., 2024).

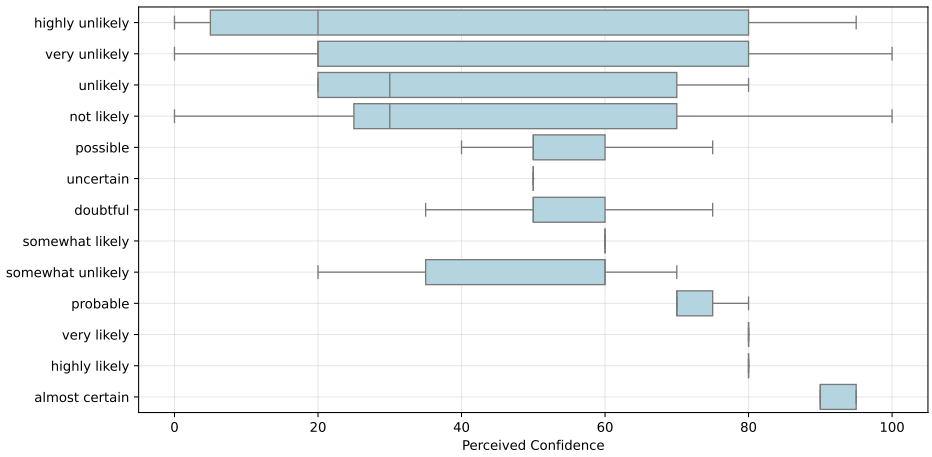

Figure 13: LLaMA-4-Maverick-17B-128E-Instruct's perceived confidence for sentences containing different hedging words. Positive hedges are mapped consistently, whereas negative hedges show high variance.

I FULL EVALUATION RESULTS

Table 3: **Full Evaluation Results (Part 1)** Model naming format follows HuggingFace.

Model	Accuracy	ECE			AUROC		
		NVU	TPU	LVU	NVU	TPU	LVU
Claude-3-5-haiku-20241022	0.603	0.297	/	0.272	0.676	/	0.718
Claude-3-7-sonnet-20250219	0.782	0.119	/	0.118	0.755	/	0.747
Claude-3-haiku-20240307	0.421	0.498	/	0.397	0.525	/	0.649
Deepseek-chat	0.784	0.112	/	0.166	0.757	/	0.718
Deepseek-reasoner	0.846	0.055	/	0.094	0.776	/	0.741
Gemini-1.5-flash	0.650	0.273	0.275	0.238	0.687	0.683	0.724
Gemini-1.5-flash-8b	0.542	0.358	0.328	0.281	0.614	0.642	0.713
Gemini-1.5-pro	0.741	0.207	0.189	0.188	0.674	0.661	0.705
Gemini-2.0-flash	0.785	0.177	0.091	0.162	0.721	0.707	0.744
Gemini-2.0-flash-lite	0.723	0.207	0.138	0.182	0.720	0.724	0.756
Gemma-2-27b-it	0.533	0.380	0.351	0.281	0.611	0.524	0.737
Gemma-2-9b-it	0.470	0.465	0.398	0.358	0.546	0.547	0.671
Gemma-2b-it	0.148	0.767	0.745	0.457	0.504	0.535	0.683
Gemma-3-12b-it	0.607	0.347	/	0.275	0.678	/	0.765
Gemma-3-1b-it	0.175	0.782	0.694	0.298	0.513	0.514	0.754
Gemma-3-27b-it	0.667	0.234	/	0.230	0.718	/	0.747
Gemma-3-4b-it	0.430	0.525	/	0.372	0.616	/	0.757
Gpt-4.1	0.791	0.176	0.074	0.166	0.748	0.684	0.748
Gpt-4.1-mini	0.761	0.168	0.130	0.184	0.765	0.727	0.744
Gpt-4.1-nano	0.635	0.243	0.214	0.252	0.712	0.698	0.781
Gpt-4o-2024-08-06	0.726	0.208	0.125	0.194	0.706	0.646	0.728
Gpt-4o-mini-2024-07-18	0.601	0.286	0.240	0.274	0.740	0.676	0.759
Grok-2-1212	0.726	0.215	0.200	0.198	0.670	0.613	0.712
Grok-3-beta	0.774	0.081	0.164	0.097	0.790	0.703	0.774
Grok-3-mini-beta	0.829	0.102	0.138	0.101	0.791	0.676	0.783
Llama-3.1-nemotron-nano-4b-v1.1	0.502	0.332	0.350	0.285	0.721	0.662	0.762
Llama-3.1-nemotron-nano-8b-v1	0.401	0.349	0.298	0.302	0.637	0.628	0.707
Llama-3.2-1b-instruct	0.166	0.510	0.573	0.342	0.525	0.463	0.691
Llama-3.2-3b-instruct	0.317	0.536	0.476	0.370	0.589	0.521	0.738
Llama-3.1-8b-instruct	0.449	0.361	0.366	0.334	0.599	0.539	0.755
Llama-3.1-70b-instruct	0.610	0.282	0.215	0.253	0.638	0.565	0.718
Llama-3.1-405b-instruct	0.722	0.165	0.153	0.183	0.712	0.615	0.729
Llama-4-maverick-17b-128e-instruct-fp8	0.787	0.131	0.158	0.122	0.727	0.652	0.752
Llama-4-scout-17b-16e-instruct	0.740	0.198	0.220	0.176	0.700	0.684	0.750
Mistral-7b-instruct-v0.1	0.263	0.677	0.606	0.341	0.495	0.486	0.760
Mistral-nemo-instruct-2407	0.405	0.580	0.446	0.375	0.522	0.510	0.703
Mistral-small-24b-instruct-2501	0.598	0.309	0.275	0.283	0.651	0.544	0.696
Mistral-small-3.1-24b-base-2503	0.545	0.354	0.308	0.244	0.552	0.471	0.624
O1-mini-2024-09-12	0.713	0.218	/	0.210	0.776	/	0.777
O3-mini-2025-01-31	0.797	0.161	/	0.152	0.757	/	0.752
O4-mini-2025-04-16	0.832	0.083	/	0.106	0.776	/	0.738
Phi-3-medium-128k-instruct	0.504	0.410	0.339	0.323	0.611	0.576	0.742
Phi-3-mini-128k-instruct	0.399	0.506	0.407	0.356	0.575	0.530	0.748
Phi-3.5-mini-instruct	0.442	0.462	0.379	0.370	0.620	0.534	0.722
Phi-3.5-moe-instruct	0.518	0.383	0.307	0.313	0.649	0.581	0.746
Phi-4-mini-instruct	0.460	0.458	0.276	0.348	0.598	0.654	0.759
Phi-4-mini-reasoning	0.801	0.149	0.094	0.135	0.666	0.751	0.715
Phi-4-reasoning	0.915	0.075	0.219	0.064	0.578	0.640	0.601

Table 4: **Full Evaluation Results (Part 2)** Model naming format follows HuggingFace.

Model	Accuracy	ECE			AUROC		
		NVU	TPU	LVU	NVU	TPU	LVU
Qwen2.5-14b-instruct	0.571	0.321	0.287	0.281	0.683	0.615	0.745
Qwen2.5-32b-instruct	0.663	0.251	0.200	0.230	0.709	0.669	0.737
Qwen2.5-3b-instruct	0.396	0.554	0.435	0.344	0.596	0.577	0.767
Qwen2.5-7b-instruct	0.508	0.389	0.364	0.329	0.642	0.615	0.725
Qwen3-0.6b	0.248	0.663	0.579	0.356	0.526	0.570	0.763
Qwen3-0.6b-base	0.275	0.711	0.546	0.321	0.501	0.575	0.784
Qwen3-0.6b-fp8	0.250	0.635	0.583	0.369	0.532	0.570	0.749
Qwen3-0.6b-think	0.415	0.556	0.375	0.389	0.519	0.713	0.793
Qwen3-1.7b	0.399	0.499	0.539	0.337	0.584	0.616	0.778
Qwen3-1.7b-base	0.374	0.539	0.435	0.334	0.538	0.560	0.776
Qwen3-1.7b-fp8	0.395	0.472	0.540	0.341	0.611	0.612	0.770
Qwen3-1.7b-think	0.695	0.249	0.224	0.221	0.686	0.786	0.764
Qwen3-14b	0.632	0.266	0.292	0.247	0.704	0.700	0.723
Qwen3-14b-awq	0.615	0.273	0.319	0.249	0.700	0.695	0.715
Qwen3-14b-base	0.593	0.335	0.270	0.284	0.624	0.595	0.706
Qwen3-14b-fp8	0.630	0.269	0.291	0.245	0.706	0.699	0.733
Qwen3-14b-think	0.829	0.096	0.090	0.084	0.750	0.809	0.752
Qwen3-235b-a22b-fp8-tput	0.728	0.193	0.190	0.188	0.727	0.704	0.732
Qwen3-235b-a22b-fp8-tput-think	0.880	0.025	0.048	0.015	0.778	0.761	0.762
Qwen3-30b-a3b	0.662	0.264	0.269	0.243	0.733	0.733	0.748
Qwen3-30b-a3b-base	0.604	0.324	0.253	0.269	0.637	0.558	0.724
Qwen3-30b-a3b-think	0.861	0.068	0.080	0.052	0.794	0.791	0.788
Qwen3-32b	0.697	0.228	0.149	0.218	0.689	0.688	0.705
Qwen3-32b-awq	0.711	0.232	0.135	0.217	0.672	0.676	0.706
Qwen3-32b-think	0.841	0.099	0.042	0.092	0.733	0.791	0.726
Qwen3-4b	0.529	0.370	0.418	0.328	0.691	0.654	0.743
Qwen3-4b-base	0.526	0.403	0.324	0.329	0.643	0.573	0.735
Qwen3-4b-fp8	0.530	0.368	0.417	0.329	0.689	0.652	0.734
Qwen3-4b-think	0.766	0.165	0.142	0.144	0.783	0.808	0.787
Qwen3-8b	0.587	0.320	0.363	0.299	0.712	0.572	0.732
Qwen3-8b-base	0.556	0.359	0.310	0.307	0.660	0.565	0.728
Qwen3-8b-fp8	0.588	0.323	0.361	0.301	0.708	0.571	0.730
Qwen3-8b-think	0.840	0.073	0.072	0.068	0.764	0.785	0.765