THE ANATOMY OF UNCERTAINTY IN LLMS

Aditya Taparia[*] **Ransalu Senanayake**[*] **Kowshik Thopalli**[†] **Vivek Narayanaswamy**[†]

[*]School of Computing and Augmented Intelligence, Arizona State University
[†]Lawrence Livermore National Laboratory

ABSTRACT

Understanding *why* a large language model (LLM) is uncertain about the response is important for their reliable deployment. Current approaches, which either provide a single uncertainty score or rely on the classical aleatoric-epistemic dichotomy, fail to offer actionable insights for improving the generative model. Recent studies have also shown that such methods are not enough for understanding uncertainty in LLMs. In this work, we advocate for an uncertainty *decomposition* framework that dissects LLM uncertainty into three distinct *semantic* components: (i) input ambiguity, arising from ambiguous prompts; (ii) knowledge gaps, caused by insufficient parametric evidence; and (iii) decoding randomness, stemming from stochastic sampling. Through a series of experiments we demonstrate that the dominance of these components can shift across model size and task. Our framework provides a better understanding to audit LLM reliability and detect hallucinations, paving the way for targeted interventions and more trustworthy systems. **Code:** https://github.com/aditya-taparia/LLM-Uncertainty

1 INTRODUCTION

Large Language Models (LLMs) have achieved remarkable success in complex reasoning and generation tasks. Despite their great capabilities, they have the tendency to generate plausible-sounding but uncertain responses. Understanding when and why these models are uncertain in their response helps in detecting hallucination Manakul et al. (2023); Kadavath et al. (2022); Kuhn et al. (2023), improving response quality Ramírez et al. (2024), and optimizing tool calling Zubkova et al. (2025). Recent studies have shown that hallucinations in LLMs are triggered because the model often guesses when they are unsure about the final response Kalai et al. (2025). This makes it important to identify when the model is uncertain and the fundamental nature and origins of uncertainty.

Uncertainty in LLMs can originate from different sources. Consider the case of Gemma3 27B model Team et al. (2025), when asked a straightforward question from TriviaQA Joshi et al. (2017),

> *"What was Walter Matthau's first movie?"*

the model consistently responded with *"The Gangster,"* while the correct answer is *"The Kentuckian."* This discrepancy highlights two possible scenarios. First, the phrasing of the question introduces input ambiguity, since "first movie" could mean Matthau's first credited role or his earliest on-screen appearance. Second, the model's internal knowledge may be incomplete or imprecise, reflecting knowledge gaps in its training data. Similarly, another source of discrepancy in the response could be introduced during output decoding. If we look at another example from the same dataset,

> *"In Hanna and Barbera's TV cartoons base on The Addams Family who was the voice of Gomez?"*

the Gemma3 27B model consistently gives correct answer, *"John Astin"* when responses are generated using greedy decoding. But when temperature decoding is used, it sometimes responds with incorrect answer, *"Ted Cassidy."* Recent work has also demonstrated that decoding strategies influence both model outputs and the resulting uncertainty estimates Hashimoto et al. (2025).

Existing approaches focuses on quantifying these uncertainty using a single score Manakul et al. (2023); Kadavath et al. (2022); Kuhn et al. (2023). While these scores are useful for ranking re-

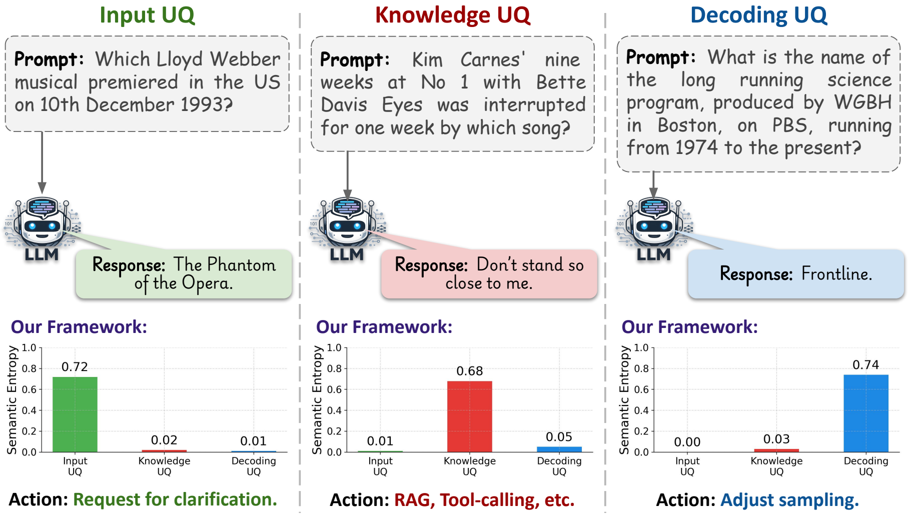

Input UQ
Prompt: Which Lloyd Webber musical premiered in the US on 10th December 1993?
LLM
Response: The Phantom of the Opera.
Our Framework:
Semantic Entropy
1.0
0.8
0.6
0.4
0.2
0.0
0.72
0.02
0.01
Input UQ
Knowledge UQ
Decoding UQ
Action: Request for clarification.
Knowledge UQ
Prompt: Kim Carnes' nine weeks at No 1 with Bette Davis Eyes was interrupted for one week by which song?
LLM
Response: Don't stand so close to me.
Our Framework:
0.01
0.68
0.05
Action: RAG, Tool-calling, etc.
Decoding UQ
Prompt: What is the name of the long running science program, produced by WGBH in Boston, on PBS, running from 1974 to the present?
LLM
Response: Frontline.
Our Framework:
0.00
0.03
0.74
Action: Adjust sampling.

Figure 1: Uncertainty decomposition on TriviaQA examples. Our framework identifies the dominant source of uncertainty—input ambiguity (left), knowledge gaps (middle), or decoding randomness (right)—and maps each to a targeted mitigation action.

sponses and guiding abstention, they are not actionable because they fail to diagnose the root cause of uncertainty. It remains unclear what intervention might reduce the uncertainty in these systems. Some works have shown how these uncertainties can be decomposed into classical aleatoric-epistemic dichotomy Senanayake (2024); Hou et al. (2024). However, recent studies Kirchhof et al. (2025); Huang et al. (2024); Xie et al. (2025); Bakman et al. (2025) have highlighted that this dichotomy is not sufficient in case of LLMs.

To address these issues, in this work we propose a framework that decomposes uncertainty in LLMs into three distinct *semantic* components: (i) input ambiguity, stemming from prompts with multiple valid interpretations; (ii) knowledge gaps, caused by insufficient training coverage or outdated information; and (iii) decoding randomness, introduced by the sampling process itself (see Figure 1). We advocate that this decomposition provides a more faithful description of uncertainty in generative models and offers actionable insights for system design. For example, high input-driven uncertainty suggests clarifying questions; high knowledge uncertainty suggests retrieval or data augmentation; and high decoding uncertainty suggests adjusting sampling strategy. Our contributions are:

1. We propose a framework for decomposing LLM uncertainty into three distinct *semantic* components: input ambiguity, knowledge gaps, and decoding randomness.
2. We empirically demonstrate how the dominant source of uncertainty shifts across different tasks and model scales.

2 Related Works

Uncertainty Quantification in LLMs. Prior work has developed various techniques to assign uncertainty scores to LLM outputs. Manakul et al. (2023) propose SelfCheckGPT, a sampling-based method that compares multiple model generations. They show that, if a fact is truly known, the samples agree, whereas hallucinated facts cause divergent answers. Similarly, Kadavath et al. (2022) uses LLM itself to estimate the probability that their own answers are correct (a "P(True)" confidence). Another approach by Kuhn et al. (2023) defines a semantic entropy score that accounts for linguistic paraphrases (shared meaning) to better predict uncertainty. These uncertainty scores have been useful for improving tasks like hallucination detection and inference efficiency. For instance, Ramírez et al. (2024) show that using a small model's uncertainty to decide whether to invoke a larger model yields an effective two-tier cascade. But they do not explain why the LLM is uncertain about a particular response and how we can improve them. With our approach, we aim to bridge this gap by decomposing uncertainty into interpretable sources.

Towards Decomposing Uncertainty in LLMs. Decomposing uncertainty into meaningful components has a long history in Bayesian and reinforcement learning Charpentier et al. (2022). Inspired by this, recent work has begun exploring uncertainty decomposition in large language models. Hou et al. (2024) introduce an input-clarification ensembling framework that generate multiple disambiguated versions of each prompt and ensemble the outputs. However, recent researches note that the simple aleatoric-epistemic split is not ideal for LLMs. Kirchhof et al. (2025) argue that classical definitions of aleatoric vs. epistemic uncertainty "contradict each other and lose their meaning" in open-ended, interactive language tasks. In particular, assigning fixed aleatoric and epistemic scores to each output cannot capture the nuanced, multi-turn uncertainty arising from under specified prompts. Motivated by these observations, we take a more fine-grained view and explicitly separate input ambiguity, knowledge gaps, and decoding randomness as distinct uncertainty sources.

3 Anatomy of Uncertainty in LLMs

We propose a formal framework for decomposing the uncertainty in a LLM's response along three distinct axes: input ambiguity, knowledge gaps, and decoding randomness. These components correspond to the core stages of the generation pipeline: the user prompt (Input), the model's learned parameters (Knowledge), and the generation procedure (Decoding).

Let X and Y represent the input and output spaces of a generative task. An LLM, parameterized by θ, defines a conditional probability distribution $p(Y \mid x, \theta)$ over the output space for a given input $x \in X$. A specific response $y \in Y$ is generated by sampling from this distribution according to a decoding strategy τ (e.g., greedy, temperature, or nucleus sampling). The uncertainty of the model for a given input x is defined as the entropy of its output distribution,

$$U_{\text{total}}(x) = \mathcal{H}\left(p(Y \mid x, \theta, \tau)\right). \tag{1}$$

Each of the three uncertainty components is defined as follows,

Input Ambiguity (U_{input}). It quantifies the model's sensitivity to prompt phrasing. We isolate this by holding θ and τ fixed while varying the input across a set of K semantically equivalent paraphrases $P(x) = \{x^k\}_{k=1}^{K}$. We generate outputs y^k for each paraphrase and compute the semantic entropy over C equivalence clusters:

$$U_{\text{input}}(P, \theta, \tau) = -\sum_{c} \hat{p}(c) \log \hat{p}(c) \quad \text{where} \quad \hat{p}(c) = \sum_{y^k \in c} p(y^k \mid x^k, \theta, \tau) \tag{2}$$

High U_{input} indicates that the output distribution is unstable across paraphrases, signaling that the prompt is underspecified and requires clarification.

Knowledge Gaps ($U_{knowledge}$). It measure uncertainty stemming from the model parameters θ. We approximate this by creating an ensemble of M LoRA-finetuned instances $\Theta = \{\theta^m\}_{m=1}^{M}$. Holding x and τ fixed, we generate responses y^m from each instance and compute the semantic entropy:

$$U_{\text{knowledge}}(x, \Theta, \tau) = -\sum_{c} \hat{p}(c) \log \hat{p}(c) \quad \text{where} \quad \hat{p}(c) = \sum_{y^m \in c} p(y^m \mid x, \theta^m, \tau) \tag{3}$$

High $U_{\text{knowledge}}$ signifies ensemble disagreement, indicating a lack of parametric evidence that warrants external retrieval (RAG) or tool-calling.

Decoding Randomness (U_{dec}). It captures uncertainty arising from the stochastic sampling process. We isolate this by holding x and θ constant while generating N responses $\{y^n\}_{n=1}^{N}$ using a specific decoding strategy $\tau \in T$. The semantic entropy is given by:

$$U_{\text{dec}}(x, \theta, \tau) = -\sum_{c} \hat{p}(c) \log \hat{p}(c) \quad \text{where} \quad \hat{p}(c) = \sum_{y^n \in c} p(y^n \mid x, \theta, \tau) \tag{4}$$

Comparing U_{dec} across different strategies (e.g., greedy vs. temperature sampling) reveals the model's inherent generation stability.

Table 1: Failure prediction performance (AUROC) of each uncertainty component on TriviaQA (fact-retrieval) and GSM8K (reasoning). Higher values indicate a stronger ability to predict incorrect model responses. The results show that the most uncertainty source is task-dependent.

Dataset	**Model**	**Input UQ**		**Knowledge UQ**		**Decoding UQ**	
		AUROC	ECE	AUROC	ECE	AUROC	ECE
TriviaQA	Llama 3 (8B)	0.705	0.340	0.499	0.513	0.731	0.364
	Gemma 3 (27B)	0.761	0.223	0.498	0.514	0.636	0.458
GSM8K	Llama 3 (8B)	0.518	0.926	0.598	0.810	0.533	0.843
	Gemma 3 (27B)	0.334	0.920	0.500	0.827	0.383	0.861

Although we treat these axes as distinct to enable analysis, they are not strictly orthogonal in practice. For instance, an ambiguous inputs can increase both input variability (U_{input}) and decoding variability (U_{dec}) by flattening the output distribution across multiple plausible interpretations. Despite such interactions, this decomposition helps identify the dominant source of uncertainty for a given query, enabling targeted system interventions.

4 Experiments

We empirically validate our framework by addressing two research questions: **(RQ1)** Can decomposed uncertainty scores effectively predict model failures across tasks? and **(RQ2)** How do dominant uncertainty sources shift with model scale and task type?

Setup. We evaluate our framework on **TriviaQA** (fact-retrieval) and **GSM8K** (reasoning) using Llama 3 (8B) and Gemma 3 (270M–27B). For full implementation details, datasets, and hyperparameters, please refer to Appendix A.

4.1 Disentangling Uncertainty for Hallucination Detection

Table 1 reveals that uncertainty dynamics are strongly **task- and model-dependent**. On TriviaQA, Llama 3 (8B) failures are best predicted by U_{dec} (AUROC 0.731), while Gemma 3 (27B) relies on U_{input} (AUROC 0.761). This indicates a shift from generation noise to prompt sensitivity as models scale. Conversely, on GSM8K, all signals weaken, though $U_{\text{knowledge}}$ remains comparatively stable. This suggests reasoning errors stem primarily from confident, incorrect internal trajectories rather than ambiguity or sampling variance. We further analyze the interaction between these uncertainty sources in Appendix C.

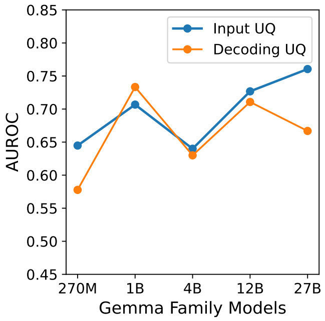


Figure 2: Failure prediction (AUROC) across Gemma 3. Input UQ reliability improves with scale.

4.2 Analysis of Scaling and Decoding Effects

To answer RQ2, we analyze how uncertainty sources evolve with model scale. Figure 2 illustrates the performance of input and decoding uncertainty for the Gemma 3 model family on TriviaQA. We observe no clear monotonic trend; the predictive power of both uncertainty types fluctuates with model size. However, a notable pattern emerges: for smaller models (1B), decoding uncertainty is a stronger predictor, while for larger models (12B, 27B), input ambiguity becomes the more reliable signal. This reinforces our finding from Table 1: as models grow, their sensitivity to input phrasing becomes a more prominent failure mode than simple generation variability.

We also investigated the impact of different decoding strategies (e.g., Greedy vs. Top-k) on failure detection. We consistently find that stochastic decoding methods are significantly more effective at revealing uncertainty than deterministic ones. A detailed comparison and analysis of these strategies is provided in Appendix B.

5 Discussion: Actionable Uncertainty Decomposition

Uncertainty decomposition is valuable not only for measuring confidence but also for understanding why LLMs fail in the first place. When uncertainty is represented by a single scalar score, it only indicates that the model is unsure, without revealing the underlying reason. As a result, the only practical response is often abstention or fallback generation. In contrast, separating uncertainty into interpretable components helps diagnose the source of failure. For example, some failures arise because the prompt itself is ambiguous and admits multiple valid interpretations. Recent work shows that LLMs can improve reliability by detecting such underspecified queries and asking clarifying questions before generating a response Li et al. (2025); Yang et al. (2025).

Other failures can originate from gaps in the model's internal knowledge. Retrieval-based systems address this by augmenting the model with external information when parametric knowledge is insufficient. Self-Routing RAG Wu et al. (2025), for example, uses uncertainty signals to decide whether a query should be answered using the model's internal knowledge or through external retrieval. However, recent analysis argues that standard predictive entropy fails to capture knowledge-related uncertainty in retrieval-augmented pipelines (Soudani et al., 2025). This further supports the importance of identifying the source of uncertainty.

Uncertainty can also arise during the generation process itself. In structured tasks such as code generation, reliability can depend strongly on token choices during decoding. Frameworks such as AdaDec therefore monitor token-level entropy and trigger additional search or reranking when decoding uncertainty becomes high He et al. (2025). In summary, these works illustrate how decomposing uncertainty turns it into a practical signal for identifying model failures and improving LLM behavior, enabling targeted interventions such as clarification, retrieval, or adaptive decoding.

6 Conclusion

We present **a unified framework for decomposing LLM uncertainty** into three distinct **semantic (not probabilistic)** components: input, knowledge, and decoding. Each is formalized over semantic equivalence classes, enabling direct comparison via a common semantic entropy measure. Systematic evaluation demonstrates that the **dominant source of uncertainty is task- and model-dependent**. Interestingly on TriviaQA, uncertainty decomposition reveals that smaller models exhibit stronger decoding-driven uncertainty, while larger models are more sensitive to prompt phrasing (input ambiguity). Our findings challenge the common practice of relying on a single uncertainty estimate and highlight fundamental differences in how semantic uncertainty manifests across task and model types.

Ethics Statement

This research focuses on analyzing the uncertainty and reliability of existing open-weights LLMs using publicly available datasets (TriviaQA, GSM8K). The study does not involve human subjects, nor does it utilize private or sensitive data. By developing a framework to decompose and diagnose uncertainty, this work aims to improve the safety and transparency of generative AI systems, potentially mitigating risks associated with hallucinations in real-world deployments. In this work, LLMs were utilized primarily as the subjects of our analysis (Llama 3, Gemma 3).

Reproducibility Statement

To ensure the reproducibility of our results, we have provided detailed descriptions of our experimental setup in Appendix A. All experiments rely on publicly available models and datasets accessible via HuggingFace. To further facilitate reproduction, we have also released our codebase.

Acknowledgments and Disclosure of Funding

This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract No. DE-AC52-07NA27344, Lawrence Livermore

National Security, LLC. This material is based upon work supported by the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research Program.LLNL-CONF-2015712.

References

Yavuz Faruk Bakman, Duygu Nur Yaldiz, Sungmin Kang, Tuo Zhang, Baturalp Buyukates, Salman Avestimehr, and Sai Praneeth Karimireddy. Reconsidering llm uncertainty estimation methods in the wild. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29531–29556, 2025.

Bertrand Charpentier, Ransalu Senanayake, Mykel Kochenderfer, and Stephan Günnemann. Disentangling epistemic and aleatoric uncertainty in reinforcement learning. *arXiv preprint arXiv:2206.01558*, 2022.

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.

Wataru Hashimoto, Hidetaka Kamigaito, and Taro Watanabe. Decoding uncertainty: The impact of decoding strategies for uncertainty estimation in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 14601–14613, 2025.

Kaifeng He, Mingwei Liu, Chong Wang, Zike Li, Yanlin Wang, Xin Peng, and Zibin Zheng. Towards better code generation: Adaptive decoding with uncertainty guidance. *arXiv preprint arXiv:2506.08980*, 2025.

Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. Decomposing uncertainty for large language models through input clarification ensembling. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 19023–19042, 2024.

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.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.

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.

Adam Tauman Kalai, Ofir Nachum, Santosh S Vempala, and Edwin Zhang. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*, 2025.

Michael Kirchhof, Gjergji Kasneci, and Enkelejda Kasneci. Position: Uncertainty quantification needs reassessment for large-language model agents. *arXiv preprint arXiv:2505.22655*, 2025.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

Belinda Z Li, Been Kim, and Zi Wang. Questbench: Can llms ask the right question to acquire information in reasoning tasks? *arXiv preprint arXiv:2503.22674*, 2025.

Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd annual meeting of the association for computational linguistics (ACL-04)*, pp. 605–612, 2004.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*, 2023.

Guillem Ramírez, Alexandra Birch, and Ivan Titov. Optimising calls to large language models with uncertainty-based two-tier selection. *arXiv preprint arXiv:2405.02134*, 2024.

Ransalu Senanayake. The role of predictive uncertainty and diversity in embodied ai and robot learning. In *Metacognitive Artificial Intellegence, Cambridge University Press*, 2024.

Heydar Soudani, Evangelos Kanoulas, and Faegheh Hasibi. Why uncertainty estimation methods fall short in rag: An axiomatic analysis. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 16596–16616, 2025.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Di Wu, Jia-Chen Gu, Kai-Wei Chang, and Nanyun Peng. Self-routing rag: Binding selective retrieval with knowledge verbalization. *arXiv preprint arXiv:2504.01018*, 2025.

Qiujie Xie, Qingqiu Li, Zhuohao Yu, Yuejie Zhang, Yue Zhang, and Linyi Yang. An empirical analysis of uncertainty in large language model evaluations. *arXiv preprint arXiv:2502.10709*, 2025.

Chenyang Yang, Yike Shi, Qianou Ma, Michael Xieyang Liu, Christian Kästner, and Tongshuang Wu. What prompts don't say: Understanding and managing underspecification in llm prompts. *arXiv preprint arXiv:2505.13360*, 2025.

Hanna Zubkova, Ji-Hoon Park, and Seong-Whan Lee. Sugar: Leveraging contextual confidence for smarter retrieval. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025.

A Experiment Setup

In this section, we outline the components of our experimental design, including the datasets, models, and evaluation metrics used to validate our framework.

A.1 Tasks and Datasets

We evaluate our framework on two distinct datasets to analyze uncertainty under different tasks, focusing on the LLM's accuracy in zero-shot prediction. For factual question answering, we use TriviaQA, a dataset that requires models to provide factually correct responses, thereby testing their learned knowledge and ability to generate precise information. We also use GSM8K to evaluate the model on mathematical reasoning, as this dataset of grade-school math word problems assesses multi-step reasoning where errors may arise from either misinterpretation of the problem statement or flaws in the logical chain.

A.2 Models

We conduct experiments across a range of model families and sizes, including Llama 3 (8B) Grattafiori et al. (2024) and Gemma 3 (270M, 1B, 4B, 12B and 27B) Team et al. (2025). Input and decoding based uncertainty estimation were tested on all these models, but model based uncertainty was only tested on Llama 3 8B and Gemma 3 27B.

A.3 Implementation Details

Below we describe the implementation detail for each component of uncertainty.

Input Ambiguity ($U_{\textbf{input}}$). To evaluate uncertainty based on input, we generate $K = 5$ semantically similar paraphrases for each prompt using GPT5-nano. For each paraphrase, we obtain a response from the target LLM using greedy decoding. We then compute the semantic entropy over this set of 5 responses, as described in equation. 2.

Knowledge Gaps ($U_{\textbf{knowledge}}$). We create an ensemble of $M = 5$ model instances for Llama 3 8B and Gemma 3 27B by fine-tuning with different random seeds on the train set of underlying dataset using LoRA. For a given prompt, we generate one response from each of the 5 models using greedy decoding and compute the semantic entropy over the resulting set of responses, as described in equation 3.

Decoding Randomness ($U_{\textbf{dec}}$). For each prompt, we generate $N = 5$ responses using different decoding methods. We then compute the semantic entropy over this set of 5 diverse responses to quantify the decoding uncertainty over the chosen decoding method, as described in equation 4.

A.4 Evaluation Metrics

To assess how effectively each decomposed uncertainty component predicts model failures (hallucinations), we formulate the problem as a binary classification task that distinguishes correct from incorrect model outputs. Generation correctness is determined using a fuzzy matching criterion: an output is labeled correct if its Rouge-L score, which measures the length of the longest common subsequence with respect to the reference answer, is greater than or equal to 0.3. Although using a Rouge-L threshold (≥ 0.3) as a proxy for semantic correctness can introduce noise, we retain this threshold to remain consistent with evaluation protocols commonly adopted in recent LLM uncertainty quantification literature (Lin & Och, 2004; Kuhn et al., 2023; Manakul et al., 2023).

Once correctness is established, we evaluate predictive performance using the Area Under the Receiver Operating Characteristic curve (AUROC), where higher values indicate stronger alignment between uncertainty scores and actual model errors. In addition, we report the Expected Calibration Error (ECE) to quantify how well the uncertainty scores correspond to true error rates.

B Impact of Decoding Strategies

In Section 4.3, we briefly noted the importance of decoding strategies. Here, we provide the full analysis of how different decoding methods influence the effectiveness of uncertainty estimation.

Figure 3 explores the failure prediction performance (AUROC) of Decoding Uncertainty (U_{dec}) when calculated using varying strategies: Beam Search, Greedy Search, Top-k Sampling, Top-p Sampling, and Temperature Sampling.

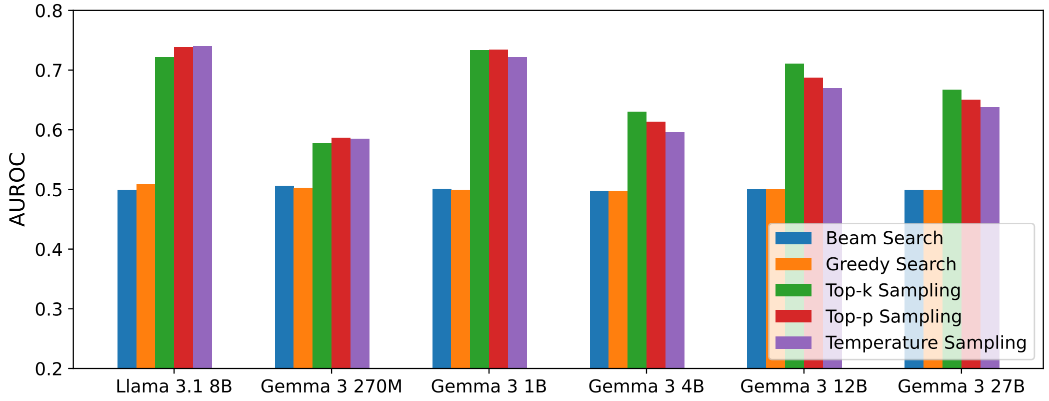
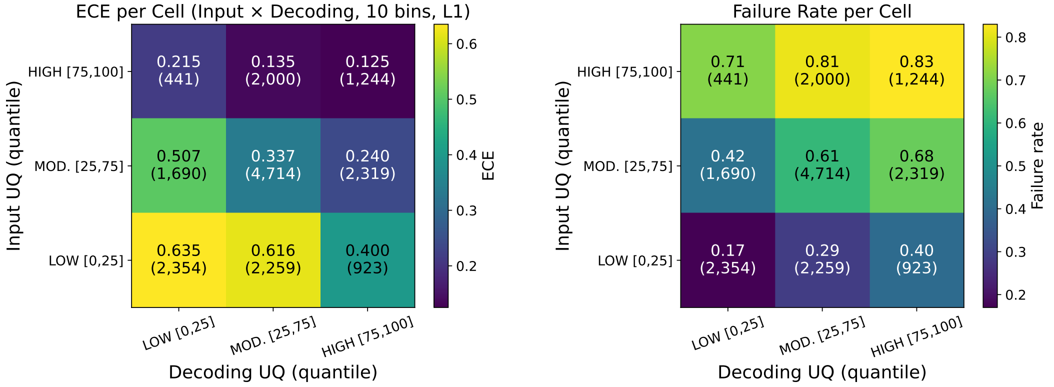


Figure 3: Comparison of failure prediction AUROC for Decoding Uncertainty when calculated using different decoding strategies. Stochastic methods (e.g., Top-k, Top-p) are significantly more effective at revealing uncertainty than deterministic ones (e.g., Greedy).



(a)

(b)

Figure 4: Joint analysis of Input Ambiguity (U_{input}) and Decoding Randomness (U_{dec}) on TriviaQA, partitioned by uncertainty quantiles. The heatmaps reveal an important insight about overconfidence: while the failure rate (b) increases with uncertainty, the model is most poorly calibrated (highest ECE in a) when it appears most confident (low uncertainty)

A consistent and striking pattern emerges across all models: stochastic decoding methods (Top-k, Top-p, and Temperature Sampling) yield significantly higher AUROC scores than deterministic methods (Beam Search, Greedy Search). This demonstrates that allowing the model to explore a diverse set of potential answers is important for revealing its underlying uncertainty. Deterministic methods, which force the model to commit to a single path, can mask this uncertainty, often leading to confidently incorrect answers.

C Interaction of Uncertainty Sources

To better understand how different uncertainty sources interact, we performed a joint analysis of input ambiguity and decoding randomness. We partitioned the TriviaQA test set into a 3x3 grid based on low, moderate, and high quantiles of U_{input} and U_{dec}. We then computed the average model failure rate and ECE within each cell.

Figure 4(b) shows a clear and intuitive trend: the model's failure rate increases monotonically with both input and decoding uncertainty. The lowest failure rate (0.17) occurs when both uncertainty scores are low, while the highest rate (0.83) occurs when both are high. This confirms that both components are meaningful indicators of correctness, and their combined effect is even stronger.

However, Figure 4(a) reveals a more surprising relationship with calibration. The model is most poorly calibrated (highest ECE of 0.635) when it appears most confident (low input and decoding uncertainty). Conversely, it is best calibrated (lowest ECE of 0.125) when it is most uncertain. This suggests that the model is often underconfident. When the model signals low uncertainty on both axes, it is only wrong 17% of the time, but its confidence level is disproportionately low, leading to poor calibration. This highlights a critical failure mode: the model's confidence is least trustworthy precisely when it should appear most certain.