

When Response Variation Does Not Indicate Uncertainty: Hallucination Detection for Multi-Answer Queries

Hongliang Dai^{1,2*}, Qiuyuan Li^{1,2}, Piji Li^{1,2}

¹College of Artificial Intelligence,
Nanjing University of Aeronautics and Astronautics, Nanjing, 211106, China

²The Key Laboratory of Brain-Machine Intelligence Technology,
Ministry of Education, Nanjing, 211106, China
{hongldai,qiuyli,pjli}@nuaa.edu.cn

Abstract

Hallucination detection in LLMs often relies on output consistency or model-internal signals. While output-based methods have the advantage of applicability to closed-source models, existing studies often overlook a key limitation: they struggle on tasks that admit multiple valid answers, where variation across generations may reflect legitimate alternative responses rather than hallucinations. In this paper, we propose a question-type-aware hallucination detection framework that distinguishes single-answer and multi-answer regimes. Given a predicted question-type signal, we derive type-conditioned diagnostic features from low- and high-temperature generations, and build two alternative hallucination detectors based on these features: a calibrated type-aware uncertainty estimator and a feature-verbalized few-shot LLM judge. Experiments on NQ-Open, WebQA, and AmbigQA across both open-source and closed-source LLMs demonstrate that our method consistently outperforms conventional semantic clustering approaches and representative baselines in hallucination detection.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable performance in generating fluent and informative responses across a wide range of tasks (Brown et al., 2020; Ouyang et al., 2022). However, they frequently generate hallucinated content that appears plausible but is factually incorrect or unsupported by evidence (Ji et al., 2023; Huang et al., 2025). Detecting such errors without external references is therefore important for reliable deployment. A common strategy is to sample multiple responses to the same input and estimate hallucination risk from their consistency (Manakul et al., 2023; Farquhar et al., 2024). For long-form

*Corresponding author.

Motivation: When Response Variation Does Not Indicate Uncertainty

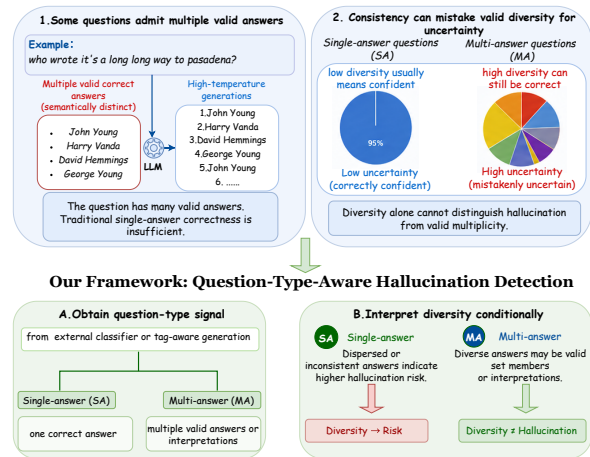


Figure 1: Motivation for type-aware hallucination detection. The same answer diversity signal indicates uncertainty under SA but legitimate variation under MA, motivating type-conditioned uncertainty estimation.

responses, it can be applied by decomposing outputs into atomic claims paired with corresponding input questions.

While this output-based strategy has the advantage of being applicable to both open- and closed-source models, we identify an important limitation that remains largely overlooked in prior work. Under this approach, diverse or contradictory sampled generations are interpreted as evidence of uncertainty. Although this assumption is generally valid for single-answer input questions (Manakul et al., 2023; Kuhn et al., 2023; Farquhar et al., 2024), in multi-answer settings, variation across generations may instead reflect legitimate alternative responses rather than hallucinations (Min et al., 2020; Stelmakh et al., 2022). As a result, uniformly interpreting diversity as evidence of hallucination can inflate uncertainty estimates and incorrectly reject valid outputs (analysis and case studies can be found in Appendix F.1 and I).

To address this limitation, we argue that output-based uncertainty estimation should be conditioned

on the input question type. For single-answer (SA) questions, semantic dispersion across sampled responses remains a strong indicator of uncertainty (Manakul et al., 2023; Farquhar et al., 2024). In contrast, for multi-answer (MA) questions, response diversity should not be directly treated as evidence of hallucination. Instead, uncertainty estimation should focus on whether the sampled responses are valid and non-contradictory (Min et al., 2020; Stelmakh et al., 2022).

Specifically, we propose a question-type-aware hallucination detection framework that conditions output-based uncertainty on whether the input admits a single answer or multiple valid answers. Given an inferred question-type signal, our method extracts type-conditioned diagnostic features from low- and high-temperature generations. These features capture semantic dispersion, answer validity, cross-answer contradiction, and type stability. Based on these features, we design two hallucination detectors: a calibrated type-aware uncertainty estimator and a feature-verbalized few-shot LLM judge (Liu et al., 2023; Zheng et al., 2023). Both operate solely on generated outputs and inferred question type information, without requiring access to model internals.

We evaluate our method on NQ-Open, WebQA, and AmbigQA (Kwiatkowski et al., 2019; Berant et al., 2013; Min et al., 2020). Across multiple open- and closed-source LLMs, our framework consistently improves hallucination detection over semantic clustering and representative baselines (Manakul et al., 2023; Farquhar et al., 2024). Ablations further show that meaningful SA/MA type signals are important for robust detection. Our code is available at <https://github.com/liqiuyuan318/MA-Hallu>.

In summary, our core contributions are:

- We identify and address a key limitation of consistency-based hallucination detection: misinterpreting legitimate multi-answer diversity as hallucination evidence.
- We propose a question-type-aware hallucination detection framework that distinguishes SA and MA regimes and derives type-conditioned diagnostic features from generated outputs.
- We build two alternative detectors, calibrated type-aware uncertainty estimation and feature-verbalized LLM judging, and validate them

across open- and closed-source LLMs.

2 Related Work

2.1 Consistency-based Uncertainty

A common line of reference-free hallucination detection estimates uncertainty from the consistency of sampled model outputs. SelfCheckGPT compares multiple responses from the same model, assuming that factual knowledge yields stable generations whereas hallucinated content tends to diverge or contradict (Manakul et al., 2023). Later work extends this idea with fact-level consistency checking, question perturbations, and cross-model agreement (Gupta et al., 2025; Zhang et al., 2023). Semantic uncertainty methods further cluster sampled generations by meaning and compute uncertainty over semantic clusters (Kuhn et al., 2023; Farquhar et al., 2024). Compared with surface-form matching, these methods better handle aliases, paraphrases, and lexical variants because semantically equivalent answers can be grouped together. However, they still generally interpret dispersion across semantic clusters as uncertainty. This assumption is effective when the input has a single expected answer, but can over-penalize inputs with multiple semantically distinct valid answers.

2.2 Open-domain Question Answering

Open-domain QA benchmarks such as NQ-Open and WebQA require models to answer questions without a provided context, using parametric knowledge, retrieval, or both (Lee et al., 2019; Kwiatkowski et al., 2019; Berant et al., 2013). However, many open-domain questions are ambiguous, under-specified, or set-valued. AmbigQA explicitly captures this phenomenon by pairing ambiguous questions with multiple plausible answers and disambiguated rewrites (Min et al., 2020), and ASQA further studies long-form answers to ambiguous questions (Stelmakh et al., 2022). Such multi-answer structure creates a challenge for hallucination detection: answer diversity may signal uncertainty for single-answer questions, but legitimate variation for multi-answer questions. Existing QA and ambiguity-focused work primarily studies answer generation or disambiguation, while the implications of valid multi-answer structure for output-based hallucination detection remain largely unexplored.

2.3 Internal Signals and LLM-based Judging

Another line of work detects hallucinations using model-internal signals, such as hidden states, probing classifiers, token-level uncertainty or representation geometry (Chen et al., 2024; Kossen et al., 2024; Du et al., 2024; Zhang et al., 2025). These studies show that factual errors can be reflected in internal model states, providing signals beyond output consistency. However, such methods typically require access to hidden representations, logits, or model-specific architectures, which limits their applicability to closed-source LLMs and makes cross-model deployment less straightforward. LLM-as-judge methods instead use LLMs to evaluate correctness, factuality, or instruction following (Zheng et al., 2023; Liu et al., 2023; Kim et al., 2024). This paradigm is flexible and can be applied when model internals are unavailable. However, the resulting judgments can be sensitive to prompt wording, demonstration selection, output format, evaluation criteria, and the judge model’s own parametric knowledge (Zheng et al., 2023; Tan et al., 2025; Zeng et al., 2024).

3 Methodology

3.1 Framework Overview

Our framework detects hallucinations by interpreting sampled answer variation under the inferred question type. Given an input question q , we first obtain an SA/MA type signal, either from an external classifier or from tag-aware generation. Then, with q as input to LLM, low-temperature decoding produces a primary answer, while high-temperature sampling provides candidate answers for feature extraction. From these generations, we compute output-based diagnostic signals, including semantic dispersion, validity, contradiction, and type stability. These signals are interpreted according to the inferred SA/MA type signal. Under SA, answer dispersion is treated as uncertainty, under MA, diversity may reflect valid alternatives, so validity and contradiction are more informative than dispersion alone. Finally, the resulting type-conditioned features serve as input to two alternative detection routes. Route A provides a compact calibrated estimator: it maps the features to a hallucination risk score using separate SA and MA risk heads. Route B instead uses an LLM judge: it verbalizes the same features as natural-language evidence for few-shot correctness assessment, allowing the judge to reason over the feature evidence.

3.2 Obtaining Question-Type Signals

We use a question-type signal to distinguish between SA and MA regimes. SA questions have a single semantically coherent answer cluster, whereas MA questions have multiple semantically distinct valid answer clusters, including set-valued and ambiguous cases. Appendix A.1 provides our full definition and examples.

A standard way to obtain this signal is to train an external question-type classifier, which is applicable to both open- and closed-source LLMs. We fine-tune a DeBERTa-v3-large classifier to predict an SA/MA label from the input question, and use this prediction for downstream feature routing and uncertainty estimation. For open-source LLMs, we further introduce a tag-aware generation strategy. Using the constructed SA/MA labels, we fine-tune the model with $\langle \text{TAG} \rangle \langle \text{ANSWER} \rangle$ targets so that the type signal and answer are produced in a single generation pass. Appendix A.2 describes the label, SFT data construction and the computation of a type-stability feature which measures the reliability of the inferred SA/MA signal. Training details are provided in Appendix B.1.

3.3 Type-conditioned Diagnostic Features

This step converts sampled generations into diagnostic features for downstream detection. For each question, low-temperature decoding produces the primary answer a_0 , while high-temperature sampling produces K additional candidate answers $\{a_i\}_{i=1}^K$ that provide uncertainty evidence.

Validity and semantic relations. We next compute answer-level validity and pairwise semantic relations among sampled answers. To estimate validity, we use the answer-generating model itself as a verifier: open-source models are fine-tuned on the training split, while closed-source models are queried with the same validity-checking prompt. This yields a plausibility score $V_i \in [0, 1]$ for each sampled answer a_i . The invalidity risk is

$$r_{\text{invalid}} = 1 - \frac{1}{K} \sum_{i=1}^K V_i.$$

We then compare sampled answers pairwise using a bidirectional NLI model. For each pair (a_i, a_j) , S_{ij} is defined as the average semantic entailment probability across the two directions, while C_{ij} is the maximum directional contradiction probability.

Route A: Type-Aware Uncertainty Estimation

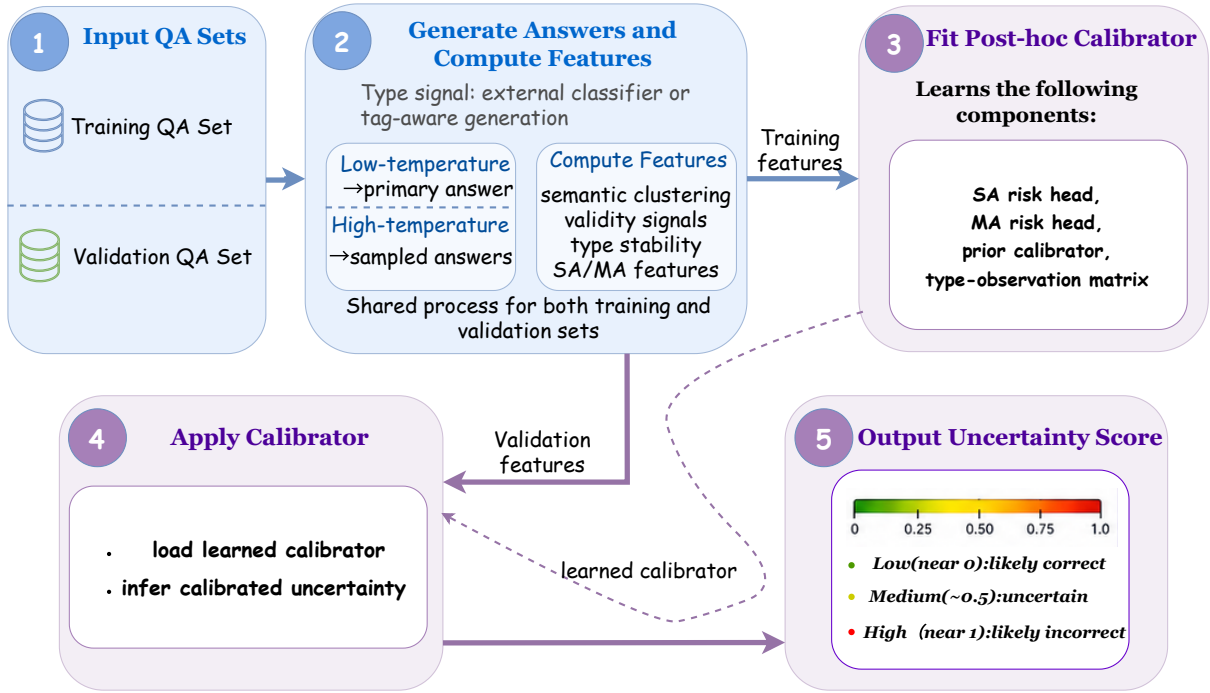


Figure 2: Route A: type-aware uncertainty estimation. Generated answers are converted into SA/MA-conditioned diagnostic features and mapped by a post-hoc calibrator to a hallucination risk score.

Finally, we partition sampled answers into semantic clusters $\mathcal{C}$ using exact-match equivalence and bidirectional entailment.

SA and MA risk features. For SA questions, answer dispersion is treated as an uncertainty signal, following prior consistency- and semantic-uncertainty-based detection methods (Manakul et al., 2023; Farquhar et al., 2024). We compute a macro-level dispersion score using the normalized entropy of validity-weighted cluster masses:

$$r_{\text{macro}} = -\frac{1}{\log |\mathcal{C}|} \sum_{c \in \mathcal{C}} p_c \log p_c,$$

$$p_c = \frac{\sum_{i \in c} V_i}{\sum_{c' \in \mathcal{C}} \sum_{j \in c'} V_j}.$$

We also compute within-cluster inconsistency:

$$r_{\text{micro}} = 1 - \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \text{mean}_{i < j, i, j \in c} S_{ij}.$$

Therefore, $x_{\text{SA}} = [r_{\text{macro}}, r_{\text{micro}}, r_{\text{invalid}}]$.

For MA questions, semantic diversity is not penalized directly. Instead, we use invalidity and NLI-based contradiction signals:

$$r_{\text{MA-invalid}} = r_{\text{invalid}},$$

$$r_{\text{MA-contradiction}} = \text{mean}_{i < j} (C_{ij} V_i V_j).$$

Therefore, $x_{\text{MA}} = [r_{\text{MA-invalid}}, r_{\text{MA-contradiction}}]$.

3.4 Route A: Type-Aware Uncertainty Estimation

Route A is a lightweight estimator that maps type-conditioned diagnostic features to a calibrated uncertainty score for hallucination detection. It assigns a hallucination risk score to a_0 using features extracted from sampled answers and the inferred question type.

To handle noisy type predictions, we treat the answer regime $z \in \{\text{SA}, \text{MA}\}$ as a latent variable. For each possible regime, we define a separate type-specific logistic risk head over the corresponding feature vector:

$$r_{\text{SA}}^{\text{base}} = \sigma(w_{\text{SA}}^\top x_{\text{SA}} + b_{\text{SA}}),$$

$$r_{\text{MA}}^{\text{base}} = \sigma(w_{\text{MA}}^\top x_{\text{MA}} + b_{\text{MA}}).$$

We estimate the empirical SA prior from sampled type signals:

$$p_{\text{SA}} = \frac{1}{|\mathcal{T}|} \sum_{t_i \in \mathcal{T}} \mathbf{1}(t_i = \text{SA}).$$

After calibrating the prior and type-stability score on the training split, we obtain the regime-specific risks r_{SA} and r_{MA} .

Route B: Feature-Verbalized LLM Judge

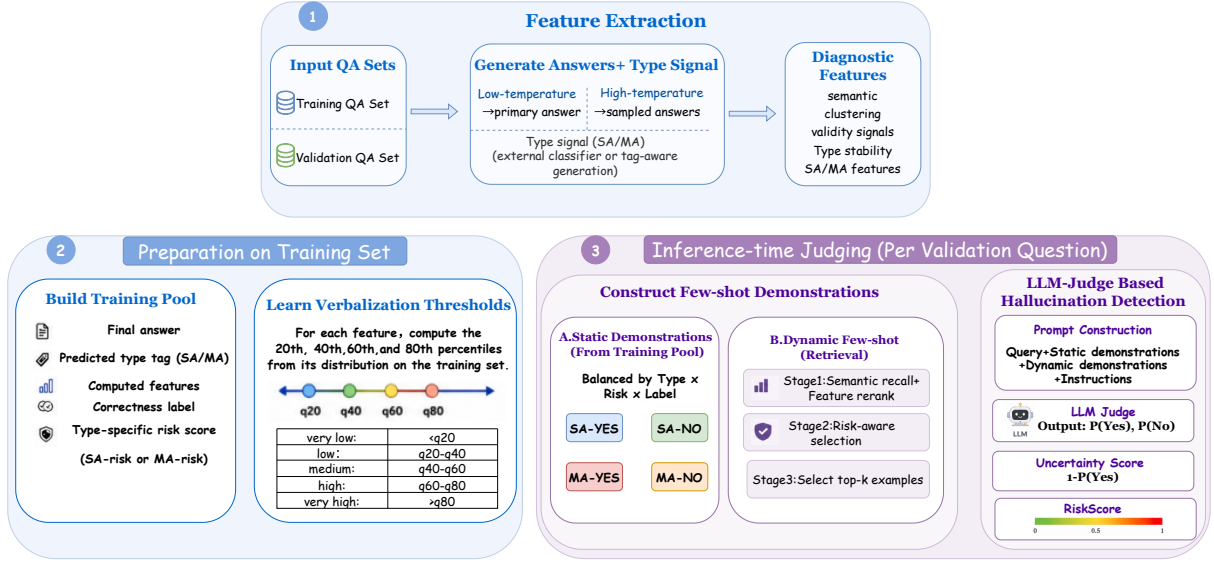


Figure 3: Route B: feature-verbalized LLM judging. Diagnostic features are verbalized with training-set thresholds and provided to the judge with static and dynamically retrieved demonstrations. The uncertainty score is $1 - P(\text{Yes})$.

The primary type signal t_0 is modeled as a noisy observation of the latent regime. Given the prior $P(z | q)$ and a learned observation model $P(t_0 | z)$, we infer the posterior regime distribution:

$$P(z | q, t_0) = \frac{P(z | q)P(t_0 | z)}{\sum_{z' \in \{SA, MA\}} P(z' | q)P(t_0 | z')}.$$

The final calibrated uncertainty score is the posterior expected risk:

$$U_{\text{cal}}(q, a_0) = \sum_{z \in \{SA, MA\}} P(z | q, t_0) r_z.$$

The calibrator is trained by marginalizing over the latent regime, with details provided in Appendix B.2.

3.5 Route B: Feature-Verbalized Few-Shot LLM-as-a-Judge

Route B presents the diagnostic features to an LLM judge in an interpretable form. Each feature is verbalized using training-set quantile thresholds and mapped to a calibrated descriptor, such as low, medium, high, or very high, while retaining its numerical value. This produces a feature summary that combines quantitative signals with natural-language evidence.

Static and dynamic few-shot demonstrations.

We construct few-shot demonstrations from the training split in two forms: static and dynamic.

Each demonstration is a structured record containing the question, low-temperature answer, inferred type signal, diagnostic features, verbalized feature summary, and correctness label. Static demonstrations are selected from four type-label buckets: SA_Yes, SA_No, MA_Yes, and MA_No, and are further stratified by calibrated risk level to form a balanced prompt. Dynamic demonstrations are retrieved separately for each validation instance. We first recall semantically similar questions using sentence embeddings, and then rerank the candidates by Mahalanobis distance in the diagnostic feature space. This process selects examples that are both semantically relevant and feature-aligned, while preserving coverage across type-label buckets and risk levels.

Judge scoring. The final prompt includes the task instruction, static and dynamically retrieved demonstrations, and the target instance with its verbalized feature summary. The LLM judge outputs the probability that the final answer is correct, and we define the uncertainty score as

$$U_{\text{judge}}(q, a_0) = 1 - P_{\text{judge}}(\text{Yes} | q, a_0, t_0, \phi),$$

where ϕ denotes the verbalized feature summary and selected demonstrations.

Models	Methods	NQ-Open			AmbigQA			WebQA		
		AUROC	AUPR	AURAC	AUROC	AUPR	AURAC	AUROC	AUPR	AURAC
	Semantic Clustering	0.7027	0.7535	0.5209	0.7375	0.7141	0.6580	0.6803	<u>0.6378</u>	0.6083
Llama	LLM Judge (Ours)	<u>0.7484</u>	<u>0.7833</u>	<u>0.6040</u>	<u>0.7782</u>	<u>0.7808</u>	<u>0.6653</u>	<u>0.6887</u>	0.6183	<u>0.6966</u>
	Type-Aware Uncertainty (Ours)	0.8006	0.8494	0.6292	0.8335	0.8517	0.6868	0.7318	0.6952	0.7119
	Semantic Clustering	0.7275	<u>0.8410</u>	0.4336	0.7156	<u>0.7920</u>	0.4999	0.6858	0.6815	0.5697
Qwen	LLM Judge (Ours)	<u>0.7549</u>	0.7788	<u>0.6102</u>	<u>0.7636</u>	0.7858	<u>0.6133</u>	<u>0.7375</u>	<u>0.7124</u>	<u>0.6984</u>
	Type-Aware Uncertainty (Ours)	0.8262	0.8650	0.6478	0.8530	0.8893	0.6584	0.7617	0.7510	0.7078
	Semantic Clustering	0.6770	<u>0.7542</u>	0.5035	0.6957	0.6937	0.6133	0.6435	0.5949	0.6116
Mistral	LLM Judge (Ours)	<u>0.6990</u>	0.7264	<u>0.5989</u>	<u>0.7461</u>	<u>0.7645</u>	<u>0.6375</u>	<u>0.7245</u>	<u>0.6680</u>	<u>0.7113</u>
	Type-Aware Uncertainty (Ours)	0.8064	0.8434	0.6509	0.8356	0.8733	0.6809	0.7480	0.7229	0.7210
	Semantic Clustering	0.7340	0.8384	0.4528	0.7671	0.7477	<u>0.6791</u>	0.7177	<u>0.7270</u>	0.5673
GLM	LLM Judge (Ours)	<u>0.7841</u>	<u>0.8425</u>	<u>0.5903</u>	<u>0.7707</u>	<u>0.7912</u>	0.6464	<u>0.7341</u>	0.7030	<u>0.6922</u>
	Type-Aware Uncertainty (Ours)	0.8238	0.8840	0.6123	0.8461	0.8727	0.6937	0.7494	0.7415	0.6930

Table 1: Main hallucination detection results on open-source LLMs. Best results are bolded, and second-best results are underlined.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate on three open-domain QA datasets covering both factoid and ambiguous multi-answer settings: **NQ-Open** (Kwiatkowski et al., 2019), **WebQA** (Berant et al., 2013), and **AmbigQA** (Min et al., 2020). NQ-Open provides Wikipedia-grounded answers to real Google queries, WebQA uses Freebase-grounded web queries, and AmbigQA extends NQ-Open with ambiguity annotations and disambiguated rewrites.

Human validation of SA/MA labels. To assess the reliability of the automatically constructed question-type labels, we manually validate the SA/MA assignments for 900 questions, with 300 samples from each dataset. The automatic labels achieve 90.44% agreement with human judgments, with Cohen’s $\kappa = 0.8089$ and Macro-F1 of 0.9041, indicating strong agreement and supporting the reliability of the reference-derived SA/MA labels. Detailed results are provided in Appendix A.3.

Models. Our experiments are conducted across four representative open-source LLMs and three closed-source LLMs. The open-source models include **Qwen2.5-7B-Instruct**, **Llama-3-8B-**

Instruct, **GLM-4-9B-Chat** and **Mistral-7B-Instruct-v0.3**, while the closed-source models include **GPT-4.1**, **Gemini-3-Flash-Preview** and **Grok-4.2**.

Baselines. Our main baseline is **Semantic Entropy** (Kossen et al., 2024). We generate one low-temperature answer and multiple high-temperature sampled answers, cluster the sampled answers by bidirectional NLI equivalence, and compute uncertainty as normalized entropy over semantic clusters. Other representative hallucination detection baselines are described in Appendix C.

Evaluation metrics. For hallucination detection, incorrect answers are treated as positive cases and the model’s uncertainty score is used as the detection score (Farquhar et al., 2024; Manakul et al., 2023). We evaluate ranking performance with **AUROC** and **AUPR** (Farquhar et al., 2024; Janiak et al., 2025), and selective prediction performance with **AURAC**, the area under the accuracy–coverage curve (Geifman and El-Yaniv, 2017; Pugnana and Ruggieri, 2023).

4.2 Implementation Details

Answer generation setup. For each question, low-temperature decoding produces the primary

Models	Methods	NQ-Open			AmbigQA			WebQA		
		AUROC	AUPR	AURAC	AUROC	AUPR	AURAC	AUROC	AUPR	AURAC
GPT	Semantic Clustering	0.6843	0.4767	0.7441	0.7181	0.3325	<u>0.8765</u>	0.6096	0.5310	<u>0.6467</u>
	LLM Judge (Ours)	<u>0.7069</u>	<u>0.5627</u>	0.7479	<u>0.8600</u>	<u>0.6856</u>	0.8838	0.6660	<u>0.5984</u>	0.6603
	Type-Aware Uncertainty (Ours)	0.7440	0.7020	<u>0.7448</u>	0.8636	0.8047	0.8709	<u>0.6480</u>	0.6309	0.6459
Gemini	Semantic Clustering	0.6487	0.4562	0.7442	0.7018	0.2726	<u>0.9187</u>	0.5911	0.5051	0.5865
	LLM Judge (Ours)	<u>0.7131</u>	<u>0.5375</u>	0.7520	<u>0.8385</u>	<u>0.5068</u>	0.9199	0.6801	0.6135	0.6847
	Type-Aware Uncertainty (Ours)	0.7218	0.6803	<u>0.7484</u>	0.8694	0.7700	0.9065	<u>0.6415</u>	<u>0.6088</u>	<u>0.6059</u>
Grok	Semantic Clustering	0.7466	0.6043	0.7614	0.7723	0.4900	0.8892	0.6293	0.5466	<u>0.6613</u>
	LLM Judge (Ours)	<u>0.7765</u>	<u>0.7168</u>	<u>0.7270</u>	<u>0.7893</u>	<u>0.5003</u>	0.8771	0.6700	<u>0.6055</u>	0.6715
	Type-Aware Uncertainty (Ours)	0.7826	0.7741	0.7208	0.8336	0.7434	<u>0.8839</u>	<u>0.6490</u>	0.6217	0.6488

Table 2: Main hallucination detection results on closed-source LLMs. Best results are bolded, and second-best results are underlined.

Model	Acc.	SA-F1	MA-F1
NQ-Open ($N=3610$; SA: 66.95%; MA: 33.05%)			
Llama	0.7800	0.8533	0.5600
Qwen	0.7775	0.8468	0.5936
Mistral	0.7350	0.8008	0.6045
GLM	0.7875	0.8571	0.5854
AmbigQA ($N=2002$; SA: 42.66%; MA: 57.34%)			
Qwen	0.6575	0.6119	0.6935
Llama	0.6300	0.6186	0.6408
Mistral	0.7325	0.6348	0.7890
GLM	0.6200	0.6122	0.6275
WebQA ($N=2032$; SA: 67.18%; MA: 32.82%)			
Qwen	0.7975	0.8492	0.6920
Llama	0.8150	0.8625	0.7176
Mistral	0.8450	0.8604	0.8258
GLM	0.8050	0.8550	0.7023

Table 3: Question-type tagging performance of tag-aware open-source LLMs. We report overall accuracy and F1 scores for SA and MA labels. The SA/MA ratios are obtained by semantic clustering over annotated correct answers.

answer, and high-temperature sampling produces $K = 9$ additional answers. The model generates a single answer, for MA questions, it selects one valid answer rather than listing all answers. We use temperature $\tau = 0.8$, top- $p = 0.9$ for high-temperature sampling.

Prompt details. Appendix D provides all prompt templates used in our pipeline, covering answer

Datasets	Acc.	MA-P	MA-R	MA-F1
NQ-Open	0.7920	0.7522	0.7311	0.7415
WebQA	0.8007	0.6667	0.7586	0.7097
AmbigQA	0.7099	0.7103	0.7489	0.7291

Table 4: Question-type classification performance of the fine-tuned DeBERTa-v3-large classifier. We report accuracy and MA-class precision, recall, and F1.

generation, type prediction, validity verification, and LLM-based judging.

Computational cost. Our framework incurs additional inference cost due to sampled generations and semantic analysis. Route A is substantially cheaper than Route B, while the sampling budget provides a direct accuracy–efficiency trade-off. A detailed latency breakdown and deployment discussion are provided in Appendix H.

5 Results and Analysis

5.1 Main Results

Hallucination detection results. Tables 1 and 2 report the main results on open- and closed-source LLMs across NQ-Open, AmbigQA, and WebQA. We compare representative semantic clustering with two variants of our method: LLM Judge and Type-Aware Uncertainty Estimation.

Question-type tagging accuracy. We also evaluate the quality of the question-type signals used

Models	Methods	NQ-Open	AmbigQA	WebQA
Llama	SelfCheck-Prompt	0.6900	0.7139	0.6591
	SelfCheck-Ngram	0.6501	0.6273	<u>0.7110</u>
	EigenScore	0.6436	0.6228	0.6337
	Energy Score	0.6452	0.6725	0.6530
	Lexical Similarity	0.6528	0.6638	0.6825
	Semantic Entropy	0.6921	0.7746	0.6714
	Predictive Kernel Ent.	0.6680	0.4510	0.6737
	Spectral Uncertainty	0.7000	0.6039	0.6800
	Input Clarif. Ens.	–	0.5951	–
	LLM Judge (Ours)	<u>0.7484</u>	<u>0.7782</u>	0.6887
	Type-Aware Uncertainty (Ours)	0.8006	0.8335	0.7318
Qwen	SelfCheck-Prompt	0.7430	0.7240	0.6831
	SelfCheck-Ngram	0.6437	0.6660	0.7018
	EigenScore	0.6695	0.6334	0.6083
	Energy Score	0.5720	0.5765	0.5231
	Lexical Similarity	0.6861	0.6972	0.6794
	Semantic Entropy	0.6931	<u>0.7680</u>	0.6880
	Input Clarif. Ens.	–	0.5951	–
	LLM Judge (Ours)	<u>0.7549</u>	0.7636	<u>0.7375</u>
	Type-Aware Uncertainty (Ours)	0.8262	0.8530	0.7617

Table 5: Comparison with existing hallucination detection methods across datasets. Each value denotes AUROC. Best results are bolded, and second-best results are underlined.

in our framework. Table 3 reports the tagging performance of Fine-tuned LLMs. Table 4 reports the performance of the DeBERTa-v3-large classifier.

Comparison with existing methods. Table 5 summarizes AUROC results from other representative methods. Compared with them, our method achieves better detection performance.

Additional discussion of the main results is provided in Appendix E.1.

Robustness to auxiliary components. We further examine sensitivity to the NLI backbone and LLM-judge prompt. Replacing the NLI model yields closely aligned performance, while Route B remains stable across several semantically aligned prompt variants. These results suggest that the improvements are not tied to a specific auxiliary model or prompt formulation. Full results are provided in Appendix E.5.

5.2 Generalization Beyond Short-Form QA

We further evaluate the proposed framework beyond short-form QA, including long-form ambiguous QA on ASQA, long-form factuality evaluation on FActScore, and retrieval-augmented multi-

Datasets	Models	Tag	ROC	PR	RAC
NQ-Open	Qwen	Gold	0.7064	0.7478	0.5701
		Rand.	0.6930	0.7132	0.5693
		All-SA	0.6088	0.6936	0.5019
	Mistral	Gold	0.7142	0.7731	0.5762
		Rand.	0.6632	0.7485	0.5226
		All-SA	0.6919	0.7643	0.5530
	GLM	Gold	0.7540	0.7997	0.5827
		Rand.	0.6666	0.7315	0.5347
		All-SA	0.6138	0.7243	0.4569
WebQA	Qwen	Gold	0.6678	0.6750	0.5933
		Rand.	0.6262	0.6171	0.5903
		All-SA	0.6298	0.6263	0.5877
	Mistral	Gold	0.6751	0.6539	0.6309
		Rand.	0.6622	0.6454	0.6361
		All-SA	0.6606	0.6430	0.6405
	GLM	Gold	0.6783	0.6580	0.6159
		Rand.	0.6446	0.6203	0.5923
		All-SA	0.5829	0.5986	0.5217
AmbigQA	Qwen	Gold	0.7815	0.8053	0.6294
		Rand.	0.7624	0.7743	0.6186
		All-SA	0.6687	0.7025	0.5739
	Mistral	Gold	0.7409	0.7512	0.6368
		Rand.	0.6446	0.6716	0.5751
		All-SA	0.7376	0.7391	0.6442
	GLM	Gold	0.7368	0.7494	0.6608
		Rand.	0.6531	0.6509	0.6165
		All-SA	0.5991	0.6286	0.5432

Table 6: LLM-as-judge ablation using diagnostic features selected by the low-temperature tag. ROC, PR, and RAC denote AUROC, AUPR, and AURAC. “Gold”, “Rand.”, and “All-SA” denote gold-tag, random-tag, and all-SA settings. Best results are bolded.

answer QA on QAMPARI. For long-form responses, we apply type-aware detection to factoid- or atomic-fact-level verification units and aggregate their risks at the response level. The proposed methods generally outperform type-agnostic uncertainty baselines across these settings. Full experimental details and results are provided in Appendix G.

5.3 Matched type-agnostic control

We compare our estimator with matched single-head controls using the same SA/MA-derived signals. Their weaker performance indicates that the gains come from explicit type-conditioned risk modeling rather than simply from richer features or type information. Full results are provided in the Appendix E.2.

5.4 Ablations on Type Information

We study whether the detector benefits from meaningful SA/MA type information. Table 6 compares **Gold**, **Random**, and **All-SA** tag settings for the LLM judge when only the feature subset associated with the provided tag is used. Gold tags generally outperform random and all-SA variants, indicating that performance gains depend on meaningful type signals. The All-SA setting often degrades performance because it collapses legitimate multi-answer diversity into single-answer uncertainty. Additional ablations are reported in Appendix E.3.

5.5 Robustness to Uncertain Type Signals

We group examples by type stability to assess robustness to uncertain SA/MA predictions. Performance generally improves with more stable type signals, while our estimator still retains positive AUROC gains over Semantic Clustering in the Borderline group across all three datasets. Full results are provided in Appendix E.4.

5.6 Analysis of Type-Dependent Diversity Signals

We further examine how answer diversity relates to hallucination under different question types. Appendix F.1 and Appendix F.2 show that semantic entropy is a stronger hallucination signal for SA questions than for MA questions, indicating that diversity under MA can reflect legitimate answer multiplicity rather than unreliability. Appendix F.3 provides feature-level evidence that the proposed diagnostic features behave consistently with our type-aware design.

6 Case Study

We provide case studies in Appendix I. These examples illustrate how the proposed method interprets answer diversity conditionally on the inferred question type: dispersed responses are treated as uncertainty under SA, while diverse valid set members are not directly penalized under MA.

7 Conclusion

In this work, we introduce a type-aware hallucination detector that conditions output-based uncertainty on the input question type. Our method derives SA/MA-conditioned diagnostic features from sampled generations and applies them to two alternative detectors: calibrated type-aware uncertainty estimation and feature-verbalized few-shot

LLM judging. Extensive experiments on NQ-Open, WebQA, and AmbigQA demonstrate consistent improvements over semantic clustering and representative baselines across open- and closed-source LLMs. Ablation studies and case studies further show that accurate SA/MA signals help distinguish hallucination-induced dispersion from legitimate answer diversity.

Limitations

Our framework requires multiple high-temperature generations to compute semantic dispersion, validity, and contradiction features. This provides richer uncertainty information but increases inference cost, especially for closed-source LLMs. Although we extend the evaluation beyond short-form QA to long-form generation on ASQA and FActScore and retrieval-augmented QA on QAMPARI, these experiments are smaller in scale than our primary evaluation, and broader validation on multi-hop reasoning, dialogue, and diverse retrieval settings remains future work.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (No.62306140), the Fundamental Research Funds for the Central Universities (NO.NXD2026009), the National Natural Science Foundation of China (No.62476127), the Natural Science Foundation of Jiangsu Province (No.BK20242039), the Fundamental Research Funds for the Central Universities (NO.NXD2026006), the Research Fund (No.PO250624101698), the Scientific Research Starting Foundation of Nanjing University of Aeronautics and Astronautics (No.YQR21022), and the High Performance Computing Platform of Nanjing University of Aeronautics and Astronautics.

References

- Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

- Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. Inside: LLMs’ internal states retain the power of hallucination detection. *arXiv preprint arXiv:2402.03744*.
- Xuefeng Du, Chaowei Xiao, and Yixuan Li. 2024. Haloscope: Harnessing unlabeled LLM generations for hallucination detection. *Advances in Neural Information Processing Systems*, 37:102948–102972.
- Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.
- Yonatan Geifman and Ran El-Yaniv. 2017. Selective classification for deep neural networks. *Advances in neural information processing systems*, 30.
- Raavi Gupta, Pranav Hari Panicker, Sumit Bhatia, and Ganesh Ramakrishnan. 2025. Consistency is the key: Detecting hallucinations in LLM generated text by checking inconsistencies about key facts. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 2053–2068.
- Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.
- Denis Janiak, Jakub Binkowski, Albert Sawczyn, Bogdan Gabrys, Ravid Shwartz-Ziv, and Tomasz Jan Kajdanowicz. 2025. The illusion of progress: Re-evaluating hallucination detection in LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 34716–34733.
- Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38.
- Seungone Kim, Jay Shin, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Ryan Shin, Sungdong Kim, James Thorne, Minjoon Seo, et al. 2024. Prometheus: Inducing fine-grained evaluation capability in language models. In *International Conference on Learning Representations*, volume 2024, pages 29927–29962.
- Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. 2024. Semantic entropy probes: Robust and cheap hallucination detection in LLMs. *arXiv preprint arXiv:2406.15927*.
- Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.
- Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 6086–6096.
- Yang Liu, Dan Iter, Yichong Xu, Shuhang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLP evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 2511–2522.
- Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. 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*, pages 9004–9017.
- Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. Ambigqa: Answering ambiguous open-domain questions. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 5783–5797.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Andrea Pugnana and Salvatore Ruggieri. 2023. AUC-based selective classification. In *International conference on artificial intelligence and statistics*, pages 2494–2514. PMLR.
- Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. Asqa: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288.
- Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Tang, Alejandro Cuadron, Chenguang Wang, Raluca Popa, and Ion Stoica. 2025. Judgebench: A benchmark for evaluating LLM-based judges. In *International Conference on Learning Representations*, volume 2025, pages 63277–63303.
- Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. 2024. Evaluating large language models at evaluating instruction following. In *International Conference on Learning Representations*, volume 2024, pages 40193–40219.

Fujie Zhang, Peiqi Yu, Biao Yi, Baolei Zhang, Tong Li, and Zheli Liu. 2025. Prompt-guided internal states for hallucination detection of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21806–21818.

Jiaxin Zhang, Zhuohang Li, Kamalika Das, Bradley Malin, and Sricharan Kumar. 2023. Sac3: reliable hallucination detection in black-box language models via semantic-aware cross-check consistency. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15445–15458.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A Question-Type Definition and Label Construction

A.1 Definition of Multi-Answer Questions

We define the question type according to the semantic structure of the valid answer set. A question is labeled as single-answer (SA) if all valid answers refer to the same underlying semantic answer target. Surface-level variants such as aliases, abbreviations, spelling variants, full names, or paraphrases are therefore treated as SA. A question is labeled as multi-answer (MA) if its valid answers form multiple semantically distinct answer clusters. We consider two major forms of MA questions.

Set-valued or list-style multi-answer questions.

Some questions ask for members of a valid answer set. This type appears in datasets such as NQ-Open and WebQA. For example, the WebQA item “what books did beverly cleary right?” has multiple valid book answers, including Ramona’s World, Ramona Quimby, Age 8, Ralph S. Mouse, and Dear Mr. Henshaw. In such cases, different sampled generations may produce different correct members of the answer set. These answers are semantically distinct but still valid. Therefore, diversity across these answer clusters should not be treated as hallucination by itself.

Ambiguous or multi-interpretation questions.

AmbigQA contains questions that admit multiple plausible interpretations. Each interpretation may correspond to a different valid answer or answer set. For example, the question “Where was the film High Plains Drifter filmed?” can be interpreted as asking where it was principally filmed,

where it was filmed in Nevada, or where it was filmed in California, with valid answers including Mono Lake, Reno, and Inyo National Forest. We treat such questions as MA when answers from different disambiguated rewrites form multiple semantic clusters. In this setting, different generations may correspond to different valid interpretations of the same ambiguous question.

Correctness judgment. For SA questions, a correct answer should match the single correct answer. For MA questions, a generated answer is considered valid if it matches one of the valid answers. Thus, an MA answer is not required to enumerate all possible answers. This definition matches our detection objective: the model should distinguish unsupported or contradictory answers from legitimate variation.

A.2 SA/MA Label and SFT Data Construction

We derive SA/MA labels from the valid reference answers of each question. We group answers into semantic clusters using sentence-embedding similarity. Questions with one semantic answer cluster are labeled SA, while questions with multiple distinct clusters are labeled MA. For ambiguous datasets, answers from all disambiguated QA pairs are collected before clustering. We optionally use an LLM verifier to refine the preliminary cluster-based label.

For tag-aware SFT, we construct examples in the format <TAG> <ANSWER>. Each SA question contributes one instance with a representative answer, while each MA question contributes one instance for each valid answer:

$$\text{output} = \text{SA } a_j \quad \text{or} \quad \text{output} = \text{MA } a_j, \quad a_j \in \mathcal{A}.$$

This trains the model to generate one valid answer rather than enumerate the entire answer set. We finally shuffle and balance the SA and MA pools to form the SFT training data.

Type stability. Type stability measures the reliability of the inferred SA/MA type signal. A high stability score indicates a reliable SA/MA type signal, while a low score suggests that type-conditioned feature routing may be less reliable. For tag-aware SFT setting, type stability is computed from the generated type tags across sampled answers. Let $\mathcal{T} = \{t_i : t_i \in \{\text{SA}, \text{MA}\}\}$ denote the valid sampled type tags, and let t_0 be the primary type tag.

Dataset	N	Acc.	κ	Macro-F1	SA F1	MA F1
NQ-Open	300	0.8567	0.7133	0.8539	0.8739	0.8340
WebQA	300	0.9600	0.9200	0.9600	0.9605	0.9595
AmbigQA	300	0.8967	0.7933	0.8966	0.8984	0.8949
Overall	900	0.9044	0.8089	0.9041	0.9095	0.8988

Table 7: Human validation of automatically constructed labels. We report agreement between reference-derived labels and manual annotations on 300 questions per dataset. Automatic labels are hidden during annotation.

We define

$$p_{\text{understand}} = \frac{1}{|\mathcal{T}|} \sum_{t_i \in \mathcal{T}} \mathbf{1}[t_i = t_0].$$

A larger value indicates that the model assigns a more stable question-type across sampled generations. In the external-classifier setting, the classifier produces a single SA/MA prediction, which is used for all sampled answers. We therefore treat it as deterministic by setting $p_{\text{understand}} = 1$.

A.3 Human Validation of SA/MA Labels

We manually validate the automatically constructed SA/MA labels on 900 questions, with 300 sampled from each dataset. Samples are stratified by the automatic labels, which are hidden during annotation to reduce confirmation bias. All samples receive a human SA or MA decision based on the question and its reference answers.

Table 7 shows that the automatic labels exhibit strong agreement with human judgments across all three datasets. These results support the reliability of the reference-derived SA/MA labels used in our experiments.

B Experimental Details

B.1 Training Details.

Tag-aware SFT for open-source LLMs. Open-source LLMs are fine-tuned to generate outputs in the tag-first format <TAG> <ANSWER>. We use LoRA-based SFT on attention and MLP projection layers, with rank 16, $\alpha = 32$, and dropout 0.05. Training uses bfloat16 precision, cutoff length 256, AdamW with cosine scheduling, learning rate 5×10^{-5} , 60 warmup steps, per-device batch size 2, gradient accumulation 8, and 4 epochs.

To improve type-signal reliability, we upweight the front SA/MA tag tokens and add an auxiliary tag discrimination loss:

$$\mathcal{L} = \mathcal{L}_{\text{wLM}} + \lambda_{\text{tag}} \mathcal{L}_{\text{tag}},$$

where $\mathcal{L}_{\text{wLM}}$ is the weighted language modeling loss and $\mathcal{L}_{\text{tag}}$ encourages correct front-tag prediction.

DeBERTa type classifier for closed-source LLMs. For black-box LLMs, we train a DeBERTa-v3-large classifier to provide the question-type signal. The classifier takes the question as input and predicts a binary SA/MA label, with SA=0 and MA=1. We fine-tune the model with maximum length 128, learning rate 1×10^{-5} , weight decay 0.01, per-device training batch size 8, per-device evaluation batch size 16, gradient accumulation 2, and 5 epochs. The best checkpoint is selected by validation macro-F1.

B.2 Calibration Details for Route A

Section 3.4 defines the two-regime risk estimator and the posterior expected uncertainty score. Here we provide the calibration details used to train the estimator on the training split. In addition to the regime-specific risk heads, the calibrator learns prior calibration parameters, an understanding-confidence gate, and a noisy observation model for the primary type signal.

Prior and understanding calibration. The empirical SA prior estimated from sampled type signals is calibrated in logit space:

$$\tilde{p}_{\text{SA}} = \sigma(\alpha_s \logit(p_{\text{SA}}) + \beta_s).$$

Similarly, the question-understanding confidence p_{u} is calibrated as

$$\tilde{p}_{\text{u}} = \sigma(\alpha_u \logit(p_{\text{u}}) + \beta_u).$$

The calibrated understanding confidence is used as a global risk gate:

$$\begin{aligned} r_{\text{SA}} &= 1 - \tilde{p}_{\text{u}} \left(1 - r_{\text{SA}}^{\text{base}} \right), \\ r_{\text{MA}} &= 1 - \tilde{p}_{\text{u}} \left(1 - r_{\text{MA}}^{\text{base}} \right). \end{aligned}$$

Thus, lower question-understanding confidence increases the calibrated risk in both regimes.

Noisy type-signal observation model. The primary type signal t_0 is treated as a noisy observation of the latent answer regime. We learn a state-conditional observation matrix

$$M_{z,t} = P(t_0 = t \mid z), \quad z, t \in \{\text{SA}, \text{MA}\}.$$

The calibrated prior over latent regimes is

$$\begin{aligned} P(z = \text{SA} \mid q) &= \tilde{p}_{\text{SA}}, \\ P(z = \text{MA} \mid q) &= 1 - \tilde{p}_{\text{SA}}. \end{aligned}$$

Together with $M_{z,t}$, this prior gives the posterior regime distribution used in Section 3.4.

Calibration objective. Let $y_i \in \{0, 1\}$ denote the hallucination label for instance i , where $y_i = 1$ indicates an incorrect answer. For each latent regime z , the likelihood of the observed label is defined by the calibrated regime-specific risk:

$$P(y_i \mid z, q_i) = r_{i,z}^{y_i} (1 - r_{i,z})^{1-y_i}.$$

We train the calibrator by minimizing the marginal negative log-likelihood over the latent regime:

$$\mathcal{L}_{\text{NLL}} = - \sum_i \log m_i,$$

where

$$m_i = \sum_{z \in \{\text{SA}, \text{MA}\}} P(z \mid q_i) P(t_{0,i} \mid z) P(y_i \mid z, q_i).$$

To encourage the observation model to preserve meaningful type information, we regularize the tag observation matrix toward a diagonal structure:

$$\mathcal{L}_{\text{tag}} = -\frac{1}{2} \sum_{z \in \{\text{SA}, \text{MA}\}} \log M_{z,z}.$$

The final calibration objective is

$$\mathcal{L}_{\text{cal}} = \mathcal{L}_{\text{NLL}} + \lambda_{\text{tag}} \mathcal{L}_{\text{tag}}.$$

In our implementation, we optimize the calibrator with Adam for 2000 epochs, using learning rate 0.03, weight decay 10^{-5} , and $\lambda_{\text{tag}} = 10^{-3}$.

C Baseline Methods

We compare our method with the representative hallucination detection methods reported in Table 5. These baselines cover consistency-based, lexical, semantic, spectral, and ambiguity-oriented uncertainty signals.

SelfCheck-Prompt. SelfCheck-Prompt detects hallucinations by comparing the primary answer with multiple sampled responses using an LLM-based consistency checker. If the sampled responses do not support the primary answer, the hallucination score increases.

SelfCheck-Ngram. SelfCheck-Ngram measures consistency among generated answers using n-gram overlap. Lower n-gram agreement indicates stronger cross-sample inconsistency and higher hallucination risk.

EigenScore. EigenScore estimates uncertainty from the spectral structure of model representations. It assumes that hallucinated and factual responses exhibit different representation geometry.

Energy Score. Energy Score uses likelihood-derived energy as a confidence signal. Higher energy indicates lower model confidence and a greater likelihood of hallucination.

Lexical Similarity. Lexical Similarity measures surface-level agreement among sampled answers. Low lexical similarity suggests unstable generations and is treated as evidence of higher uncertainty.

Semantic Entropy. Semantic Entropy clusters sampled answers into semantic equivalence classes and computes entropy over the resulting cluster distribution. Higher entropy indicates larger semantic dispersion and higher hallucination risk.

Predictive Kernel Entropy. Predictive Kernel Entropy estimates uncertainty from a kernel-induced predictive distribution over generated answers. It captures semantic diversity through pairwise similarities among samples.

Input Clarification Ensemble. Input Clarification Ensemble constructs clarified variants of the original question and aggregates the model’s responses across these variants. Large variation across clarified inputs indicates higher uncertainty.

Spectral Uncertainty. Spectral Uncertainty measures uncertainty from the spectral properties of a semantic similarity matrix. It captures structural dispersion among generated answers and assigns higher uncertainty to more diverse semantic configurations.

D Prompt Templates

For reproducibility, we provide the prompt templates used throughout our experimental pipeline. Tables 8 and 9 summarize the prompts for answer generation, validity and NLI assessment, and LLM-based hallucination judging. We additionally provide the alternative LLM-judge prompts used in the prompt-robustness analysis in Appendix E.5.

Prompt	Template
Answer generation (open-source LLMs)	You are a question answering assistant for open-domain QA. First decide the question type, then answer with one concise answer. Use SA if the question has one underlying semantic answer target. Aliases, full names, abbreviations, spelling variants, or paraphrases of the same entity still count as SA. Use MA if the question has multiple semantically distinct valid answers, including set-valued questions or genuinely ambiguous questions. Output exactly one tag followed by exactly one answer. If the tag is MA, choose one valid answer from the answer set. Do not list multiple answers. Do not explain. Format: <TAG> <ANSWER>.
Answer generation (closed-source LLMs)	You are a question answering assistant for open-domain QA. Answer the question with one concise answer. Do not explain. Do not list multiple answers. Return only the answer text.
Answer-validity check	You are a knowledgeable and rational evaluator assessing open-domain questions. The question may be ambiguous and have multiple correct answers depending on reasonable interpretations. Determine whether the "Candidate Item" answers the question correctly. If the candidate is plausible or defensible under any reasonable interpretation, respond strictly with Yes. If it is incorrect or impossible under all interpretations, respond strictly with No. Return only Yes or No. Question: Name a fruit. Candidate Item: Apple. → Yes; Question: Name a fruit. Candidate Item: Chair. → No.
Pairwise NLI	Premise Context: {question}. Fact: The answer is a_i . Hypothesis: Based on the context, the answer can be described as a_j . Premise Context: {question}. Fact: The answer is a_j . Hypothesis: Based on the context, the answer can be described as a_i .

Table 8: Prompt templates for tag-aware answer generation, closed-source answer generation, answer-validity check, and pairwise NLI.

D.1 LLM-Judge Prompt Variants

For the prompt-robustness analysis in Appendix E.5, we construct three variants of the original LLM-judge prompt while preserving the same prediction objective and diagnostic information. The variants differ mainly in conciseness, factuality emphasis, and feature emphasis.

Concise. This variant shortens the original prompt while preserving the same decision principle.

You are a judge for open-domain question answering hallucination detection. Decide whether the final answer correctly answers the question.

You are given the question, the final low-temperature answer, the predicted question-type tag, and a diagnostic feature summary from sampled answers.

Question type tags:
- SA: the question has a single expected answer.
- MA: the question may have multiple valid answers or interpretations.

Use both the final answer itself and the diagnostic features.
For SA, high semantic disagreement among sampled answers indicates higher

risk.

For MA, answer diversity can be valid, so focus more on invalidity and contradiction.

Return strictly one word: Yes or No.

Strict. This variant places greater emphasis on factual validity.

You are a strict factuality judge for open-domain question answering. Your task is to decide whether the final answer is correct for the question.

Mark Yes only if the final answer is semantically compatible with the question and is valid under at least one reasonable interpretation.
Mark No if the answer is unsupported, irrelevant, contradictory, overly vague, or invalid under the question context.

You are also given a diagnostic feature summary from multiple sampled answers. Use these features as auxiliary evidence:

- For SA questions, high global disagreement, local inconsistency, or invalidity suggests that the final answer is unreliable.
- For MA questions, high diversity alone is not a reason to reject the answer; instead, focus on invalidity and

Prompt	Template
Feature-verbalized LLM judge	You are a strict judge for open-domain question answering hallucination detection. Determine whether the final answer is correct for the question. You are given: (1) the question, (2) the final low-temperature answer, (3) the low-temperature question-type tag, and (4) a diagnostic feature summary extracted from multiple sampled answers. Judge correctness using both semantic compatibility and the diagnostic feature summary. SA denotes a question with a single expected answer, while MA denotes a question with multiple valid answers or interpretations. Global semantic consensus risk measures disagreement among sampled answers; local semantic inconsistency measures weak cohesion within semantic groups; invalidity risk measures whether sampled answers are unlikely to be valid; question-understanding confidence measures type stability; and multi-answer invalidity and contradiction risks measure invalidity and conflict under the MA regime. First judge whether the final answer itself is valid. For SA, high global disagreement, local inconsistency, or invalidity indicates higher hallucination risk. For MA, answer diversity is not itself a negative signal; instead focus on multi-answer invalidity and contradiction. If the answer is semantically incompatible with the question, output No. If tag-relevant risks are high, prefer No; if the answer is valid and the relevant risks are mild, output Yes. Under MA, the answer must still be independently valid.
Output format	Open-source: Return strictly one word: Yes or No. Closed-source: Return only JSON in the following format: {"label": "Yes" or "No", "confidence": a number between 0 and 1}. Here, Yes means that the final answer is correct and No means that it is hallucinated or incorrect. The confidence indicates how confident you are in the predicted label.
Target instance format	Question: {question} Final Answer: {low-temperature answer} Low-temperature Tag: {SA/MA} Diagnostic Feature Summary: - Global semantic consensus risk: {level} ($r_{\text{macro}} = \{\text{value}\}$) - Local semantic inconsistency: {level} ($r_{\text{micro}} = \{\text{value}\}$) - Invalidity risk: {level} ($r_{\text{invalid}} = \{\text{value}\}$) - Question-understanding confidence: {level} ($p_{\text{understand}} = \{\text{value}\}$) - Multi-answer invalidity risk: {level} ($r_{\text{ma-invalid}} = \{\text{value}\}$) - Multi-answer contradiction risk: {level} ($r_{\text{ma-contradiction}} = \{\text{value}\}$)

Table 9: Prompt template for the feature-verbalized LLM judge. Open- and closed-source models use the same judging instructions and target-instance format, differing only in the required output format. Static and dynamically retrieved few-shot demonstrations follow the same target-instance format.

contradiction among plausible answers.	- Local semantic inconsistency measures weak cohesion within semantic groups.
Do not assume correctness merely because the tag is MA. The final answer must still be independently valid.	- Invalidity risk measures whether sampled answers are unlikely to be valid.
Return strictly one word: Yes or No.	- Question-understanding confidence measures stability of the predicted SA/MA tag.
Feature-focused. This variant places greater emphasis on the diagnostic features while still requiring the final answer to be valid.	- Multi-answer invalidity risk measures invalidity under the MA regime.
You are a hallucination detection judge. Your decision should be based on the final answer and the diagnostic feature summary.	- Multi-answer contradiction risk measures conflict among plausible MA answers.
The diagnostic features summarize sampled-answer behavior:	Interpret features according to the predicted tag:
- Global semantic consensus risk measures disagreement among sampled answers.	- SA: semantic disagreement is strong evidence of uncertainty or hallucination.
	- MA: semantic diversity may be legitimate, so do not penalize diversity alone.
	Focus on invalidity and contradiction.

If the final answer is incompatible with the question, output No.
 If the final answer is plausible and tag-relevant risks are low, output Yes.
 If tag-relevant risks are high, prefer No.

Return strictly one word: Yes or No.

E Additional Experimental Results

E.1 Additional Analysis of Main Results

We provide a more detailed analysis of the main results reported in Section 5.1. Tables 1 and 2 show that type-aware uncertainty estimation consistently improves over semantic clustering across datasets and model families. The gains are especially clear on NQ-Open and AmbigQA, where the distinction between single-answer and multi-answer regimes is important for interpreting sampled answer variation. These results suggest that type-conditioned feature routing helps avoid treating all semantic dispersion as hallucination evidence.

On open-source models, Type-Aware Uncertainty Estimation achieves the strongest overall performance across AUROC, AUPR, and AURAC in most settings. Compared with the LLM Judge route, the calibrated estimator often gives stronger ranking performance, indicating that the compact type-conditioned risk heads can effectively convert diagnostic features into a hallucination score. The LLM Judge route also improves over semantic clustering in many cases, showing that verbalized diagnostic features provide useful evidence for correctness assessment.

The closed-source results further demonstrate that the framework is applicable in black-box settings. Since closed-source models do not expose internal states, our method relies only on generated answers and the external DeBERTa-based question-type classifier. The improvements over semantic clustering show that the proposed type-aware formulation does not depend on access to model internals or tag-aware fine-tuning.

We also evaluate the quality of the SA/MA type signals. Table 3 reports the tagging performance of tag-aware open-source LLMs, and Table 4 reports the performance of the DeBERTa-v3-large classifier. Although the predicted tags are not perfect, they provide meaningful type information for downstream uncertainty estimation. This is consistent with the ablation results in Section 5.4, where

gold tags generally outperform random and all-SA variants.

Finally, Table 5 compares our method with representative hallucination detection baselines. The results show that Type-Aware Uncertainty Estimation achieves strong and consistent AUROC performance across datasets. This supports our main claim that output diversity should be interpreted conditionally on the answer regime, rather than used as a type-agnostic uncertainty signal.

E.2 Matched Type-Agnostic Controls

We construct two matched type-agnostic controls. The first receives the same diagnostic information as our method, including semantic-consistency, validity, contradiction, and the raw SA- and MA-specific risk scores, but treats them as ordinary numerical features in a single logistic head,

$$r_{\text{agn}} = \sigma(\mathbf{w}^\top \mathbf{x} + b).$$

It does not explicitly combine the SA- and MA-specific risks according to the inferred question type. The second control additionally includes the predicted type probabilities p_{SA} and p_{MA} as ordinary input features, while retaining the same single-head architecture.

As shown in Table 11, both controls underperform the full type-aware estimator. These results indicate that the gain is not solely due to richer feature access or type information, but to explicitly interpreting and combining the diagnostic signals according to the inferred SA/MA regime.

E.3 Additional Ablation Analysis

We provide additional ablations for both detector routes.

Table 12 compares two feature-selection strategies. Low-tag uses only the diagnostic features associated with the low-temperature predicted type signal, whereas All provides the full feature set.

Table 13 reports hallucination detection performance after splitting the evaluation set by inferred question type. We compare Type-Aware Uncertainty Estimation with the tag-free Semantic Clustering baseline on the full set, the SA subset, and the MA subset. Type-Aware Uncertainty improves AUROC on both SA and MA subsets in most settings. The gains on SA questions show that the proposed method preserves strong single-answer hallucination detection, while the gains on MA questions indicate that it better handles legitimate

answer diversity rather than treating all semantic dispersion as hallucination evidence.

Table 14 compares LLM-judge variants with gold, no-label, all-SA, and random type signals, under both few-shot and zero-shot settings.

Table 15 compares the type-aware uncertainty estimator with random-tag and all-SA variants.

E.4 Robustness to Uncertain Question-Type Signals

To examine the effect of uncertain SA/MA predictions, we group examples according to type stability $p_{\text{understand}}$ into High ($p_{\text{understand}} \geq 0.90$), Medium ($0.70 \leq p_{\text{understand}} < 0.90$), and Borderline ($p_{\text{understand}} < 0.70$) groups. Table 16 reports the full results for Qwen2.5-7B-Instruct.

Lower type stability is generally associated with higher type entropy and answer error rates, indicating that these groups capture genuinely difficult boundary cases. Despite the increased uncertainty, our type-aware estimator retains positive AUROC gains over Semantic Clustering in the Borderline group across all three datasets. The gains are generally larger for more stable type predictions, while the soft combination of SA- and MA-specific risks allows the estimator to degrade gracefully when the question type is uncertain.

E.5 Robustness to Auxiliary Components

We further examine whether the proposed framework is sensitive to specific auxiliary components used for semantic analysis and LLM-based judging.

NLI-backbone sensitivity. Our diagnostic features rely on NLI predictions for semantic entailment and contradiction. We therefore replace DeBERTa-large-MNLI with RoBERTa-large-MNLI while keeping the generated answers and all other components fixed. As shown in Table 10, performance remains closely aligned across the two backbones on both AmbigQA and WebQA, including the SA- and MA-specific subsets. This indicates that the proposed type-conditioned formulation is not sensitive to a particular NLI backbone.

LLM-judge prompt robustness. We evaluate Route B using four semantically aligned judge prompts: the original prompt, a concise variant, a stricter factuality-oriented variant, and a feature-focused variant. As shown in Table 17, performance varies only moderately across prompt formulations, with average AUROC ranging from 0.7284

Dataset	Evaluation NLI	AUROC	AUPR	SA AUROC	MA AUROC
AmbigQA	DeBERTa-large-MNLI	0.8549	0.8721	0.8452	0.8681
	RoBERTa-large-MNLI	0.8521	0.8675	0.8370	0.8711
WebQA	DeBERTa-large-MNLI	0.7278	0.7608	0.7475	0.6733
	RoBERTa-large-MNLI	0.7315	0.7604	0.7518	0.6722

Table 10: Sensitivity to the NLI backbone. We replace DeBERTa-large-MNLI with RoBERTa-large-MNLI while keeping the generated answers and the remaining pipeline fixed.

Method	Type Prob.	Type-Cond.	AUROC	AUPR	AURAC
Type-Aware	✓	✓	0.7736	0.7355	0.6889
Single-head	—	—	0.7406	0.6951	0.6569
Single-head + Type Prob.	✓	—	0.7422	0.7008	0.6542

Table 11: Matched type-agnostic controls. All variants use the same SA/MA-derived diagnostic features. Type Prob. indicates whether predicted SA/MA probabilities are provided as input, and Type-Cond. indicates explicit regime-specific risk modeling. Single-head controls do not recover the performance of the full type-aware estimator.

to 0.7605. The feature-focused variant slightly outperforms the original prompt, suggesting that Route B does not rely on a narrowly tuned prompt. Full prompt templates are provided in Appendix D.

F Additional Analysis

F.1 Diversity–Error Relationship

We measure how answer diversity relates to hallucination across question types by comparing error rates between high- and low-entropy examples:

$$\Delta_{\text{err}} = \text{ErrRate}_{\text{high entropy}} - \text{ErrRate}_{\text{low entropy}}.$$

Low- and high-entropy examples are defined by the bottom and top entropy quantile bins within each dataset–model–type group. As shown in Figure 4, Δ_{err} is consistently larger for SA than MA questions, indicating that entropy is a stronger hallucination signal under SA. For MA questions, the weaker error increase suggests that diversity may reflect legitimate answer multiplicity.

F.2 Entropy–Error Trend Analysis

Beyond the high–low entropy quantile gap reported in the main text, we also examine the full entropy–error trend. For each dataset–model–question-type group, we fit a linear relationship between semantic entropy and empirical incorrect rate across entropy quantile bins. The resulting slope captures how rapidly the incorrect rate changes as entropy increases within each group, providing a comple-

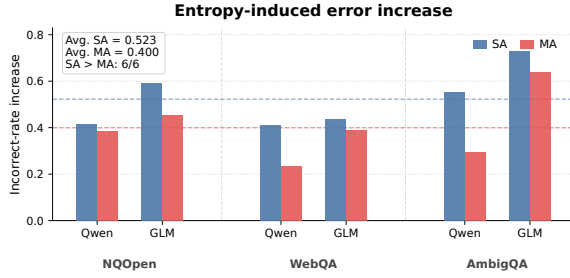


Figure 4: Semantic entropy is more predictive of hallucination for SA than MA questions, suggesting that answer diversity is a stronger error signal in single-answer regimes.

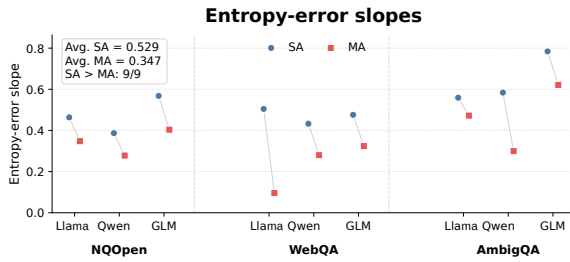


Figure 5: Entropy–error slope analysis. We fit a linear trend between semantic entropy and empirical incorrect rate across entropy quantile bins for each dataset–model–type group. Larger slopes for SA questions indicate that semantic entropy is a stronger hallucination signal in the single-answer questions than in the multi-answer questions.

mentary measure to the bottom-vs.-top quantile comparison.

As shown in Figure 5, the entropy–error slope is larger for SA questions than for MA questions in all 9 examined pairs. The average slope is 0.529 for SA and 0.347 for MA. Although semantic entropy can still provide useful evidence for some MA cases, its weaker association with errors shows that answer diversity should be interpreted conditionally on question type.

F.3 Diagnostic Feature Contrast Analysis

To better understand the behavior of the proposed diagnostic features, we analyze how each feature differs between correct and incorrect answers under the SA and MA regimes. For each feature f , we compute the contrast

$$\Delta_f = E[f \mid \text{wrong}] - E[f \mid \text{correct}],$$

separately for SA and MA questions within each dataset–model pair.

As shown in Figure 6, most feature contrasts are positive across datasets and models, indicating

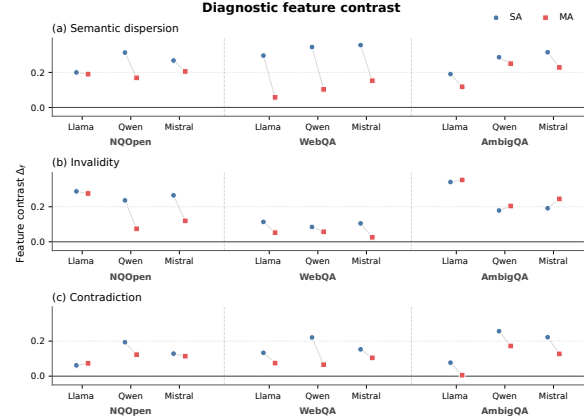


Figure 6: Diagnostic feature contrast across datasets and models. Positive values indicate larger values for incorrect answers. Semantic dispersion is more discriminative for SA questions, while invalidity and contradiction better support MA detection.

that the proposed diagnostic features are generally aligned with answer correctness. These results provide additional support for our type-conditioned feature design: SA risk estimation benefits from semantic dispersion and invalidity, whereas MA risk estimation should emphasize invalidity and contradiction rather than directly penalizing diverse but valid answers.

G Generalization Beyond Short-Form QA

G.1 Long-Form Question Answering on ASQA

We evaluate 500 ASQA examples using the annotated disambiguated QA pairs as factoid-level verification units. Unit coverage is determined by normalized lexical matching and sentence-level NLI against the acceptable answers. We compute type-aware uncertainty for each unit and aggregate the resulting risks into an answer-level score.

Table 18 shows that the calibrated type-aware estimator consistently improves AUROC over Semantic Clustering across all evaluated models, supporting the extension of our framework to long-form QA.

G.2 Atomic-Fact Evaluation on FActScore

We further evaluate the framework on FActScore at the atomic-fact level. For each factual statement, we reconstruct a claim-conditioned question, use the original claim answer as the anchor response, and generate additional high-temperature answers. We then compute the same semantic-consistency,

validity, contradiction, and SA/MA-aware signals used in the short-form setting.

As shown in Table 19, the calibrated type-aware estimator consistently outperforms Semantic Entropy across all evaluated settings and metrics, providing further evidence that the proposed formulation generalizes to long-form factuality evaluation.

G.3 Retrieval-Augmented Multi-Answer QA on QAMPARI

We evaluate the framework on 500 QAMPARI examples under retrieval augmentation. For each question, the generator receives retrieved passages, and candidate validity is assessed against the retrieved evidence, introducing challenges such as incomplete or irrelevant retrieval and multiple supported answers.

As shown in Table 20, the proposed methods generally improve AUROC and AUPR over the baseline across all evaluated models, supporting the effectiveness of type-aware uncertainty modeling under retrieval augmentation.

H Computational Cost

Our framework introduces additional computation because uncertainty estimation requires multiple sampled generations and pairwise semantic analysis. Table 21 reports end-to-end latency on NQ-Open with Qwen2.5-7B for different numbers of high-temperature samples K . We separately report primary-answer generation, sampled-answer generation, hallucination-detection overhead, and total latency.

Route A introduces moderate additional overhead relative to Semantic Clustering, whereas Route B is substantially more expensive because of the additional LLM-judge call. Increasing K also yields the expected trade-off between sampling budget and computational cost.

I Case Studies

We present qualitative examples to illustrate why answer diversity should not be interpreted as hallucination evidence without considering the answer regime. Tables 22 and 23 show representative cases from AmbigQA and WebQA. In MA cases, sampled answers may differ substantially but still correspond to valid answer-set members or reasonable interpretations, treating such diversity as uncertainty would therefore over-penalize correct outputs. In contrast, dispersed responses under

SA are more likely to indicate unreliable generation. These examples show that our method distinguishes legitimate multi-answer variation from hallucination-induced dispersion by combining the inferred SA/MA with validity and contradiction signals.

Dataset	Model	Method	Feat.	AUROC	AUPR	AURAC
NQ-Open	Llama	LLM Judge	Low-tag	0.7172	0.7672	0.5649
		LLM Judge	All	0.7448	0.7823	0.5874
		Type-Aware Uncertainty	Low-tag	0.7958	0.8720	0.5941
		Type-Aware Uncertainty	All	0.7951	0.8708	0.5935
	Qwen	LLM Judge	Low-tag	0.7256	0.7400	0.6065
		LLM Judge	All	0.7305	0.7451	0.6077
		Type-Aware Uncertainty	Low-tag	0.7852	0.8329	0.6274
		Type-Aware Uncertainty	All	0.7862	0.8343	0.6276
	Mistral	LLM Judge	Low-tag	0.7412	0.8131	0.5739
		LLM Judge	All	0.7369	0.8000	0.5708
		Type-Aware Uncertainty	Low-tag	0.8111	0.8584	0.6335
		Type-Aware Uncertainty	All	0.8111	0.8586	0.6335
	GLM	LLM Judge	Low-tag	0.7462	0.7981	0.5760
		LLM Judge	All	0.7690	0.8336	0.5825
		Type-Aware Uncertainty	Low-tag	0.8266	0.8874	0.6073
		Type-Aware Uncertainty	All	0.8260	0.8868	0.6071
WebQA	Llama	LLM Judge	Low-tag	0.6516	0.6289	0.6187
		LLM Judge	All	0.6388	0.6231	0.6250
		Type-Aware Uncertainty	Low-tag	0.7009	0.6830	0.6589
		Type-Aware Uncertainty	All	0.7041	0.6868	0.6602
	Qwen	LLM Judge	Low-tag	0.6780	0.6673	0.6289
		LLM Judge	All	0.6951	0.7010	0.6333
		Type-Aware Uncertainty	Low-tag	0.7301	0.7656	0.6452
		Type-Aware Uncertainty	All	0.7302	0.7649	0.6453
	Mistral	LLM Judge	Low-tag	0.6882	0.6783	0.6546
		LLM Judge	All	0.6869	0.6711	0.6580
		Type-Aware Uncertainty	Low-tag	0.6826	0.6833	0.6585
		Type-Aware Uncertainty	All	0.6840	0.6865	0.6588
	GLM	LLM Judge	Low-tag	0.7017	0.7174	0.6230
		LLM Judge	All	0.7200	0.7295	0.6293
		Type-Aware Uncertainty	Low-tag	0.7615	0.7897	0.6607
		Type-Aware Uncertainty	All	0.7607	0.7888	0.6603
AmbigQA	Llama	LLM Judge	Low-tag	0.7986	0.8062	0.6717
		LLM Judge	All	0.8111	0.8049	0.6839
		Type-Aware Uncertainty	Low-tag	0.8715	0.8866	0.7146
		Type-Aware Uncertainty	All	0.8732	0.8873	0.7151
	Qwen	LLM Judge	Low-tag	0.7255	0.7217	0.5957
		LLM Judge	All	0.7579	0.7551	0.6192
		Type-Aware Uncertainty	Low-tag	0.8434	0.8518	0.6852
		Type-Aware Uncertainty	All	0.8501	0.8672	0.6877
	Mistral	LLM Judge	Low-tag	0.7550	0.7594	0.6630
		LLM Judge	All	0.7349	0.7593	0.6497
		Type-Aware Uncertainty	Low-tag	0.8293	0.8681	0.6879
		Type-Aware Uncertainty	All	0.8321	0.8728	0.6889
	GLM	LLM Judge	Low-tag	0.7501	0.7387	0.6756
		LLM Judge	All	0.7604	0.7548	0.6787
		Type-Aware Uncertainty	Low-tag	0.8364	0.8505	0.7133
		Type-Aware Uncertainty	All	0.8360	0.8521	0.7128

Table 12: Comparison of feature-selection strategies for the feature-verbalized LLM judge and calibrated type-aware uncertainty estimation. *Low-tag* uses only the diagnostic features corresponding to the low-temperature predicted tag, while *All* uses the full diagnostic feature set. We report AUROC, AUPR, and AURAC for hallucination detection.

Model	Split	NQ-Open			WebQA			AmbigQA		
		SC	Type-Aware	Gain	SC	Type-Aware	Gain	SC	Type-Aware	Gain
Qwen	All	0.7275	0.8262	+0.0987	0.6858	0.7617	+0.0760	0.7156	0.8530	+0.1374
	SA	0.7392	0.8426	+0.1034	0.7065	0.7961	+0.0896	0.7521	0.9036	+0.1514
	MA	0.7135	0.8043	+0.0908	0.6723	0.7003	+0.0280	0.7062	0.8374	+0.1312
Llama	All	0.7027	0.8006	+0.0979	0.6803	0.7318	+0.0515	0.7375	0.8335	+0.0961
	SA	0.7297	0.8094	+0.0797	0.7188	0.7563	+0.0375	0.7466	0.8377	+0.0911
	MA	0.6617	0.7489	+0.0871	0.6264	0.6903	+0.0638	0.7354	0.8319	+0.0965
Mistral	All	0.7340	0.8064	+0.0724	0.7177	0.7480	+0.0303	0.7671	0.8356	+0.0686
	SA	0.7623	0.8286	+0.0663	0.7400	0.7977	+0.0577	0.7870	0.8591	+0.0721
	MA	0.6845	0.7703	+0.0858	0.6749	0.6715	-0.0034	0.7607	0.8291	+0.0684

Table 13: AUROC comparison under inferred question-type splits. **SC** denotes Semantic Clustering. Type-Aware improves over SC on both SA and MA subsets in most settings. Dataset sizes are: NQ-Open All/SA/MA = 3610/2420/1190, WebQA = 2032/1378/654, and AmbigQA = 2002/492/1510.

Datasets	Models	Tag	Few-shot			Zero-shot		
			ROC	PR	RAC	ROC	PR	RAC
NQ-Open	Llama	Gold	0.7391	0.7730	0.5803	0.7375	0.7968	0.5647
		No	0.7256	0.7610	0.5689	0.7108	0.7744	0.5508
		All-SA	0.7315	0.7666	0.5723	–	–	–
		Rand.	0.7073	0.7323	0.5831	–	–	–
	Qwen	Gold	0.7383	0.7751	0.6007	0.7079	0.7450	0.5766
		No	0.6988	0.7529	0.5870	0.7011	0.7360	0.5831
		All-SA	0.6986	0.7117	0.5908	–	–	–
		Rand.	0.7068	0.7239	0.5978	–	–	–
WebQA	Llama	Gold	0.6858	0.6471	0.6078	0.6651	0.6530	0.6251
		No	0.6605	0.6289	0.6308	0.6619	0.6429	0.6332
		All-SA	0.6557	0.6333	0.6353	–	–	–
		Rand.	0.6490	0.6346	0.6199	–	–	–
	Qwen	Gold	0.7039	0.7285	0.6296	0.7025	0.7112	0.6347
		No	0.6601	0.6432	0.6063	0.6769	0.6546	0.6210
		All-SA	0.6819	0.6905	0.6348	–	–	–
		Rand.	0.6858	0.6647	0.6585	–	–	–
AmbigQA	Llama	Gold	0.8025	0.8060	0.6811	0.7808	0.8069	0.6625
		No	0.8044	0.8140	0.6790	0.7607	0.8008	0.6480
		All-SA	0.7948	0.7810	0.6807	–	–	–
		Rand.	0.8046	0.8143	0.6791	–	–	–
	Qwen	Gold	0.7894	0.7968	0.6494	0.7364	0.7595	0.6168
		No	0.7611	0.7765	0.6376	0.7264	0.7472	0.6282
		All-SA	0.7386	0.7785	0.6188	–	–	–
		Rand.	0.7574	0.7773	0.6184	–	–	–

Table 14: Comparison of LLM-as-judge settings with and without few-shot demonstrations. We report AUROC, AUPR, and AURAC, abbreviated as ROC, PR, and RAC for compactness. “Gold”, “No”, “All-SA”, and “Rand.” denote gold-tag, no-tag, all-SA-tag, and random-tag settings, respectively. Best results are bolded.

Model	Setting	NQ-Open			WebQA			AmbigQA		
		ROC	PR	RAC	ROC	PR	RAC	ROC	PR	RAC
Llama	Type-Aware	0.8006	0.8494	0.6292	0.7318	0.6952	0.7119	0.8335	0.8517	0.6868
	Random	0.7860	0.8318	0.6218	0.6569	0.6000	0.6686	0.8340	0.8610	0.6872
	All-SA	0.7934	0.8378	0.6255	0.7110	0.6297	0.7037	0.8335	0.8582	0.6849
Mistral	Type-Aware	0.8064	0.8434	0.6509	0.7480	0.7229	0.7210	0.8356	0.8733	0.6809
	Random	0.7399	0.7842	0.6093	0.6692	0.6407	0.6635	0.8187	0.8544	0.6641
	All-SA	0.7810	0.7862	0.6412	0.7294	0.6764	0.7117	0.8059	0.8479	0.6611

Table 15: Ablation results for calibrated type-aware uncertainty estimation. We compare the full type-aware estimator with random-tag and all-SA variants. We report AUROC, AUPR, and AURAC, abbreviated as ROC, PR, and RAC for compactness. Best results are bolded.

Dataset	Stability Group	Samples	Error Rate	Mean Unc.	Type Stability	Type Entropy	Route A	Semantic Clust.	Gain
WebQA	All	2032	0.4828	0.4851	0.9085	0.2633	0.7617	0.7125	+0.0493
	High	1358	0.4543	0.4573	1.0000	0.0000	0.7897	0.7393	+0.0505
	Medium	350	0.5400	0.5381	0.8286	0.6449	0.6580	0.6122	+0.0459
	Borderline	324	0.5401	0.5448	0.6111	0.9547	0.7072	0.6643	+0.0429
AmbigQA	All	2002	0.6054	0.5918	0.8419	0.4522	0.8530	0.7718	+0.0811
	High	861	0.5610	0.5563	1.0000	0.0000	0.8555	0.7652	+0.0903
	Medium	575	0.6191	0.6203	0.8332	0.6340	0.8438	0.7569	+0.0869
	Borderline	566	0.6590	0.6170	0.6101	0.9553	0.8509	0.7913	+0.0596
NQ-Open	All	3610	0.6028	0.5953	0.9061	0.2704	0.8262	0.7484	+0.0777
	High	2377	0.5822	0.5866	1.0000	0.0000	0.8586	0.7472	+0.1114
	Medium	635	0.6283	0.6082	0.8310	0.6393	0.7511	0.7585	-0.0074
	Borderline	598	0.6572	0.6163	0.6128	0.9536	0.7442	0.7226	+0.0217

Table 16: Robustness to uncertain question-type signals on Qwen2.5-7B-Instruct. Examples are grouped by type stability into High ($p_{\text{understand}} \geq 0.90$), Medium ($0.70 \leq p_{\text{understand}} < 0.90$), and Borderline ($p_{\text{understand}} < 0.70$). Route A and Semantic Clust. report hallucination-detection AUROC for the proposed calibrated type-aware estimator and Semantic Clustering. Gain denotes the AUROC difference between Route A and Semantic Clustering.

Dataset	Model	Prompt Variant	AUROC	AUPR	AURAC
AmbigQA	Llama-3-8B	Concise	0.7474	0.8019	0.5904
		Original	0.7667	0.8138	0.6032
		Strict	0.7612	0.8121	0.6047
		Feature-focused	0.8273	0.8741	0.6340
	Qwen2.5-7B	Concise	0.7403	0.7814	0.5878
		Original	0.7759	0.7990	0.6001
		Strict	0.7590	0.7914	0.5969
		Feature-focused	0.7575	0.7983	0.5902
NQ-Open	Llama-3-8B	Concise	0.7527	0.8059	0.6034
		Original	0.7680	0.8152	0.6108
		Strict	0.7649	0.8102	0.6137
		Feature-focused	0.7798	0.8180	0.6179
	Qwen2.5-7B	Concise	0.7354	0.7131	0.6561
		Original	0.7613	0.7334	0.6725
		Strict	0.7598	0.7302	0.6730
		Feature-focused	0.7476	0.7171	0.6585
WebQA	Llama-3-8B	Concise	0.6840	0.5705	0.7425
		Original	0.7012	0.5869	0.7501
		Strict	0.6915	0.5672	0.7481
		Feature-focused	0.7079	0.6134	0.7467
	Qwen2.5-7B	Concise	0.7109	0.6255	0.7234
		Original	0.7238	0.6487	0.7294
		Strict	0.7319	0.6506	0.7360
		Feature-focused	0.7431	0.6812	0.7394
Average	–	Concise	0.7284	0.7164	0.6506
		Original	0.7495	0.7328	0.6610
		Strict	0.7447	0.7270	0.6621
		Feature-focused	0.7605	0.7504	0.6644

Table 17: Robustness of Route B to LLM-judge prompt formulations. We compare four semantically aligned prompt variants across six dataset–model settings. The bottom rows report averages over all six settings. Bold values denote the best result within each dataset–model setting and for the overall average.

Model	Method	AUROC	AUPR	AURAC
Llama-3-8B-Instruct	Semantic Clustering	0.6568	0.6908	0.5363
	LLM Judge (Ours)	0.6642	0.7306	0.5391
	Type-Aware Uncertainty (Ours)	0.7215	0.7949	0.5677
Qwen2.5-7B-Instruct	Semantic Clustering	0.6753	0.7864	0.4555
	LLM Judge (Ours)	0.6611	0.7927	0.4290
	Type-Aware Uncertainty (Ours)	0.7536	0.8513	0.5032
Mistral-7B-Instruct	Semantic Clustering	0.6578	0.7424	0.5056
	LLM Judge (Ours)	0.6775	0.7711	0.5128
	Type-Aware Uncertainty (Ours)	0.7163	0.8128	0.5208
GPT-5.4	Semantic Clustering	0.5613	0.3358	0.7624
	LLM Judge (Ours)	0.7259	0.6488	0.7472
	Type-Aware Uncertainty (Ours)	0.7911	0.7704	0.7588

Table 18: Long-form hallucination detection results on 500 ASQA examples. Annotated disambiguated QA pairs are used as factoid-level verification units, and unit-level type-aware risks are aggregated into an answer-level score. Bold values denote the best result for each model and metric.

Evaluation Setting	Method	AUROC	AUPR	Macro-F1	Hallucination-F1
InstructGPT-100, Qwen2.5-NQ-Open, $K = 9$	Semantic Entropy	0.5837	0.4699	0.5503	0.5503
	Ours (Calibrated)	0.6607	0.5151	0.6187	0.6182
InstructGPT-100, Llama-NQ-Open, $K = 9$	Semantic Entropy	0.6091	0.4905	0.5841	0.5394
	Ours (Calibrated)	0.6781	0.5347	0.6438	0.6316
InstructGPT-183, Mistral-WebQA, $K = 9$	Semantic Entropy	0.6346	0.6151	0.5980	0.5609
	Ours (Calibrated)	0.7126	0.6637	0.6547	0.6475
ChatGPT-183, Mistral-WebQA, $K = 9$	Semantic Entropy	0.6092	0.4116	0.5861	0.4385
	Ours (Calibrated)	0.6935	0.4862	0.6277	0.4919
ChatGPT-183, Llama-NQ-Open, $K = 9$	Semantic Entropy	0.5798	0.3729	0.5536	0.3474
	Ours (Calibrated)	0.6639	0.4205	0.6200	0.4855

Table 19: Atomic-fact hallucination detection results on FActScore under the expanded $K = 9$ evaluation. The evaluation setting specifies the biography source and evaluation size, followed by the sampling model and the dataset on which it was trained. Bold values denote the better result within each evaluation setting.

Model	Method	AUROC	AUPR	AURAC
Llama-3-8B	Baseline	0.6140	0.3995	0.7320
	Type-Aware Uncertainty	0.6200	0.4339	0.7028
	LLM Judge	0.6306	0.4922	0.7302
Qwen2.5-7B	Baseline	0.5539	0.3538	0.7213
	Type-Aware Uncertainty	0.6080	0.4696	0.7270
	LLM Judge	0.6064	0.4423	0.7398
Mistral-7B	Baseline	0.6027	0.4005	0.7343
	Type-Aware Uncertainty	0.6269	0.5253	0.6882
	LLM Judge	0.6771	0.6922	0.7365

Table 20: Hallucination detection results on 500 QAMPARI examples under retrieval augmentation, where answer validity is assessed against the retrieved evidence. Bold values denote the best result for each model and metric.

Method	K	Gen. Calls	Pairs	Primary	Sampled	Detection	End-to-End
Direct Response	0	1	0	0.413	0.000	0.000	0.413
Semantic Clustering	3	4	3	0.413	0.553	0.035	1.001
	5	6	10	0.413	0.651	0.040	1.104
	9	10	36	0.413	0.837	0.133	1.383
Route A	3	4	3	0.413	0.553	0.725	1.690
	5	6	10	0.413	0.651	1.127	2.190
	9	10	36	0.413	0.837	1.903	3.152
Route B	3	4	3	0.413	0.553	4.265	5.231
	5	6	10	0.413	0.651	4.565	5.623
	9	10	36	0.413	0.837	5.183	6.433

Table 21: End-to-end latency breakdown on NQ-Open with Qwen2.5-7B. Latency is reported in seconds per example. K denotes the number of high-temperature sampled answers, and Pairs denotes the number of pairwise semantic comparisons.

MA Correct	SA Hallucinated
Question: Who conducted the little albert study on conditioned fear? Low-temp Response: John B. Watson Predicted Type: MA Sampled Answers: John B. Watson and Rosalie Rayner; John B. Watson; John B. Watson; John B. Watson and Rosalie Rayner; John B. Watson; John B. Watson; John B. Watson and Rosalie Rayner; John B. Watson and Rosalie Rayner; John B. Watson and Rosalie Rayner Gold Answers: John B. Watson; Watson; Rosalie Rayner; Rayner Detection Label: Correct Interpretation: The sampled answers correspond to compatible valid variants in the AmbigQA answer set. Diversity reflects valid ambiguity rather than hallucination.	Question: The ratio of the wave height to its wave length is called? Low-temp Response: Wavenumber Predicted Type: SA Sampled Answers: wave number; amplitude; amplitude; amplitude; wave number; Wavenumber; wave number; Wavenumber; Wavenumber Gold Answers: wave steepness; wave steepnes; Airy wave theory Detection Label: Hallucination Interpretation: The sampled answers repeatedly produce plausible but incorrect concepts. Since neither concept matches the gold target, the detector assigns high risk even though the answer distribution is not broadly diverse.

Table 22: Case studies from AmbigQA.

SA Hallucination	MA Correct
Question: Who played Alf on the TV show? Low-temp Response: William F. Nolan Predicted Type: SA Sampled Answers: William F. Claxton; Paul Fusco; Paul Fusco; William F. Nolan; William F. Nolan; William F. Nolan; William F. Nolan; Paul Fusco; Paul Fusco Gold Answers: Paul Fusco Detection Label: Hallucination Interpretation: The low-temperature response is incorrect and sampled answers are split across different people. Under SA, entity-level dispersion is treated as unreliability evidence.	Question: What language is most commonly spoken in Belgium? Low-temp Response: Dutch Language Predicted Type: MA Sampled Answers: Dutch Language; French Language; French Language; Dutch Language; Dutch Language; Dutch Language; Dutch Language; Dutch Language; French Language Gold Answers: French Language; German Language; Dutch Language Detection Label: Correct Interpretation: The sampled answers cover valid members of the answer set. The MA tag allows the detector to treat this diversity as legitimate multi-answer variation.

Table 23: Case studies from WebQA.