RaTI: Reliability-aware Truth Inference for Multi-Model Question Answering

Anonymous authors
Paper under double-blind review

Abstract

Large language models (LLMs) have demonstrated strong reasoning capabilities on question answering (QA), while factual errors remain common due to limitations in knowledge coverage. Combining responses from multiple LLMs can improve answer reliability through their complementary knowledge. A central challenge in multi-model QA is determining how much each candidate should influence the final answer because factual reliability varies across candidates. Existing methods typically use an additional LLM for aggregation or treat agreement among generated answers as confidence. However, response agreement does not necessarily indicate factual reliability. Several models may generate the same incorrect answer, giving an unreliable candidate strong consensus and suppressing a less frequent correct answer. To address this limitation, we propose RaTI (**R**eliability-**a**ware **T**ruth **I**nference), a novel reliability-aware truth inference framework for robust multi-model QA aggregation. Rather than relying solely on response consistency, RaTI models the factual reliability of each candidate through the internal representations of LLMs. It further converts the estimated hallucination risks into credibility weights to calibrate semantic Minimum Bayes Risk inference. In this manner, RaTI enables high-reliability candidates to dominate the final aggregation while suppressing misleading false consensus, thereby reducing the reliance on response agreement signals in existing methods. Experiments on six QA benchmarks show that RaTI consistently improves over individual models and representative collaboration baselines, while further analyses confirm the effectiveness of reliability estimation and its robustness under increasing candidate error rates.

1 Introduction

Large language models (LLMs) have demonstrated strong knowledge modeling and reasoning capabilities, achieving remarkable performance on question answering (QA) tasks (Qin et al., 2026; Miao et al., 2025). However, constrained by the boundaries of their knowledge coverage and reasoning capabilities, LLMs may still produce factual errors when handling complex reasoning problems (Jiang et al., 2023). Importantly, such errors often vary by model, as differences in pre-training data and model architectures lead to distinct strengths and failure patterns across questions (Huang et al., 2024). This model-level complementarity has motivated multi-model answer aggregation, which combines responses from multiple LLMs to construct a richer candidate space and improve final prediction accuracy.

Existing multi-model collaboration QA frameworks fall into two broad categories. The first category relies on an additional LLM aggregator to integrate multiple candidate answers as shown in Figure 1(a) (Xiong et al., 2023). Typically, a more capable LLM, such as GPT-4, is employed to analyze the candidates and generate the final answer (Wang et al., 2025a). However, its effectiveness depends on the aggregator's ability to identify erroneous candidates. Alternatively, a second category adopts aggregation strategies based on self-consistency as shown in Figure 1(b) (Wang et al., 2025b; 2023). These methods repeatedly sample outputs from participating models and use the degree of agreement among the generated answers as a proxy for confidence, based on which they select the final answer (Wang et al., 2026a; Taubenfeld et al., 2025).

Although consistency provides a useful signal of collective agreement, it can become unreliable when multiple models make the same error on challenging questions. This limitation arises because

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

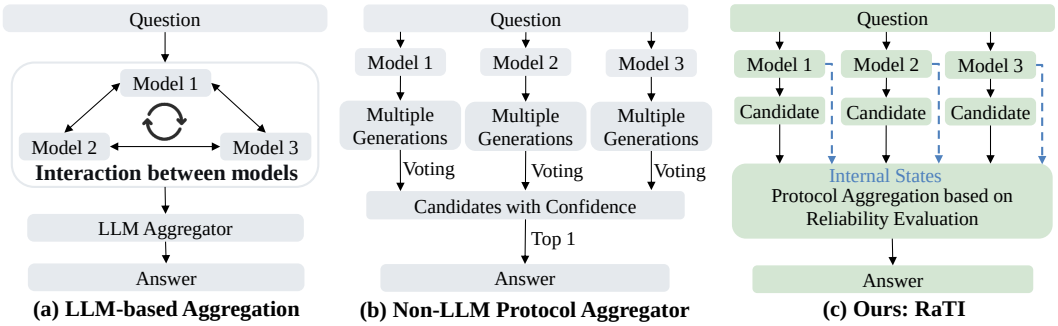


Figure 1: Architecture comparison between prior collaborative methods and our RaTI framework.

existing methods **often treat candidates uniformly without explicitly modeling their reliability**. Self-consistency therefore does not necessarily reflect the factual reliability of an individual candidate (Zhang et al., 2025a). A model may generate the same incorrect answer across multiple samples and consequently receive a high self-consistency confidence (Tan et al., 2025). For example, given the question *What is the capital of Australia?*, one model may repeatedly generate the incorrect answer *Sydney*, thereby receiving high consistency confidence, whereas another less stable model may still produce the correct answer *Canberra*. Therefore, without explicitly distinguishing the reliability of candidates, unreliable responses may disproportionately affect the final aggregation decision.

To address this limitation, an additional signal beyond consistency among candidates is needed to estimate the factual reliability of each response. Hallucination detection provides a feasible solution by identifying content that lacks contextual or factual support and estimating the likelihood of factual errors (Ji et al., 2023; Wang et al., 2026b). Recent studies show that internal representations of language models encode signals related to hallucinations and factual errors, enabling hallucination risk estimation for each candidate (Chen et al., 2024a; Zhang et al., 2025c; Han et al., 2025). This risk can serve as a reliability signal, allowing the aggregation process to assign greater influence to reliable candidates while reducing the impact of high-risk ones. In this setting, answer aggregation can be viewed as inferring the most reliable answer from multiple candidate observations with different levels of credibility. This perspective naturally connects multi-model answer aggregation with truth inference, where observations contribute differently according to their estimated reliability (Li et al., 2016; Zheng et al., 2017).

Accordingly, we propose the RaTI (**R**eliability-**a**ware **T**ruth **I**nference) framework for multi-model collaborative QA, as shown in Figure 1(c). Unlike consistency-based aggregation, which derives confidence from agreement among generated answers, RaTI estimates candidate reliability independently from hidden-state factuality signals and then uses it to regulate semantic consensus. Specifically, the framework first prompts multiple lightweight models to generate candidate answers independently, thereby constructing a diverse candidate space. Subsequently, a hallucination detector extracts the hidden-state representations, evaluating the fine-grained reliability of each candidate answer in full view of the input and response. The resulting hallucination risk scores are then transformed into credibility weights that constrain the subsequent truth inference computation. This weighting allows low-risk candidates to exert greater influence on the consensus decision, while reducing the impact of high-risk candidates even when they receive substantial agreement from other models. In this way, RaTI combines factual reliability and semantic agreement within a unified answer-selection objective.

The main contributions of this paper are summarized as follows.

- We distinguish candidate factual reliability from inter-model agreement in multi-model question answering, highlighting that response agreement alone is insufficient to determine the reliability of individual candidates.
- We propose **RaTI**, which estimates candidate reliability from internal representations and transforms hallucination risks into credibility weights for semantic Minimum Bayes Risk inference, allowing candidate reliability to directly regulate consensus formation.
- Extensive experiments on six QA benchmarks demonstrate consistent improvements over individual models and representative multi-model collaboration baselines. Further analyses validate the effectiveness of reliability estimation and reliability-aware aggregation as candidate error rates increase.

2 Related Works

Multi-Model Question Answering and Aggregation. Multi-model collaboration has been increasingly explored for question answering by leveraging the complementary capabilities of different language models. Existing approaches mainly rely on inter-model interaction, generative aggregation, or confidence-based selection. Multi-Agent Debate (Du et al., 2024), ChatEval (Chan et al., 2024), and C2R (Jang et al., 2025) refine predictions through iterative communication, while Mixture-of-Agents (MoA) (Wang et al., 2025a) and LLM-Blender (Jiang et al., 2023) aggregate multiple candidate responses using additional ranking or generative modules. Other methods avoid explicit interaction and select answers according to confidence or consistency, such as SLM-MUX (Wang et al., 2026a), while routing methods like FrugalGPT (Chen et al., 2024b) dynamically choose models based on capability and cost. However, these methods typically infer answer quality from interaction, aggregation, or response agreement, which can be unreliable when multiple models produce correlated errors. Our work instead introduces candidate-level reliability as an additional signal for multi-model answer aggregation.

Reliability Estimation for LLM Responses. Estimating the reliability of model-generated responses has been extensively studied in hallucination detection and uncertainty estimation. Output-based approaches commonly use sampling consistency or generation uncertainty. For example, SelfCheckGPT (Manakul et al., 2023) compares multiple sampled responses, while semantic entropy (Farquhar et al., 2024) measures uncertainty over semantically equivalent generations. Recent studies further show that internal model representations contain informative signals associated with factual correctness. INSIDE (Chen et al., 2024a), MIND (Su et al.), and factuality probes (Han et al., 2025) exploit hidden states for hallucination detection, while In-Context Sharpness (Chen et al., 2024c) identifies activation patterns related to factual errors. Other methods investigate richer representation dynamics, including Chain-of-Embedding (Wang et al., 2025c), LLM-Check (Sriramanan et al., 2024), ICR Probe (Zhang et al., 2025c), and layer-wise information deficiency (Kim et al., 2025), while PRISM (Zhang et al., 2025b) studies the generalization of internal-state-based detectors across domains. These studies demonstrate that response reliability can be estimated from signals beyond surface-level agreement, but they primarily focus on hallucination detection or model self-evaluation. In contrast, we use such reliability estimates to support answer selection in multi-model question answering.

3 Methods

RaTI formulates multi-model question answering as hallucination evaluation constrained truth inference. As shown in Figure 2, RaTI first generates candidate responses from multiple lightweight models (Section 3.1), then evaluates the hallucination risk of each candidate (Section 3.2), and finally incorporates the resulting reliability signals into truth inference (Section 3.3). RaTI estimates the factual reliability of each response from model internal states and maps the resulting hallucination risks to credibility weights. These weights regulate each candidate's contribution to semantic consensus, reducing the influence of repeated but unreliable responses. We use hallucination evaluation based on hidden states with reliability weighted MBR as the default configuration.

3.1 Multi-Model Collaborative Sampling

Lightweight language models may suffer from model-specific uncertainty and decoding errors, which can accumulate during generation and lead to hallucinated responses. To improve robustness, RaTI constructs a heterogeneous model set

$$\mathcal{M} = \{M_1, M_2, \ldots, M_n\},$$

where $\mathcal{M}$ denotes the participating model set, M_i is the i-th model, and n is the total number of models. Model heterogeneity provides complementary inductive biases and failure patterns under the same input.

For each $M_i \in \mathcal{M}$, RaTI generates one candidate response using deterministic decoding:

$$R_i = \text{Decode}(M_i \mid C, Q), \quad i = 1, \ldots, n, \tag{1}$$

where R_i is the response generated by M_i, C denotes the reference context, Q is the input question, and $\text{Decode}(\cdot)$ denotes the decoding procedure.

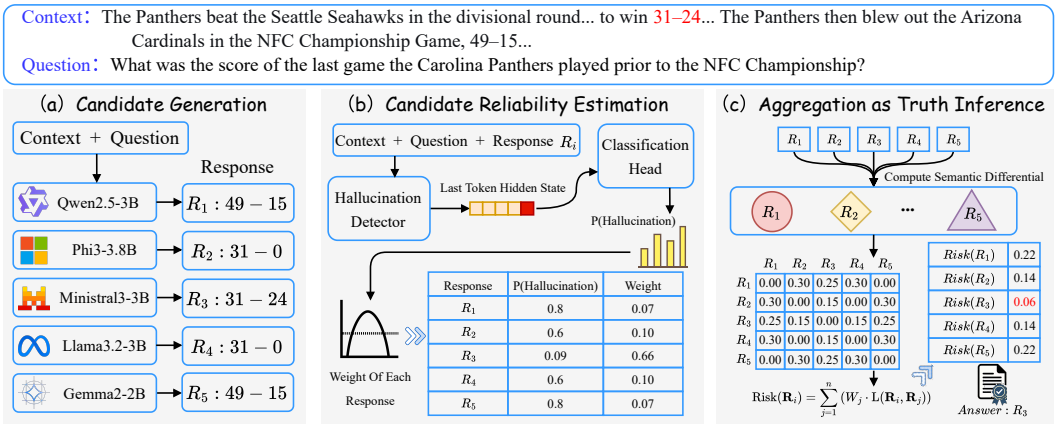


Figure 2: Overview of the proposed RaTI framework. (a) Candidate Response Generation produces diverse responses from multiple lightweight language models. (b) Candidate Reliability Estimation quantifies the hallucination risk of each response and maps it to a credibility weight. (c) Aggregation as Truth Inference selects the final answer through reliability weighted Minimum Bayes Risk inference.

The resulting candidate set

$$\mathcal{R} = \{R_1, R_2, \ldots, R_n\}$$

defines the hypothesis space for subsequent reliability estimation and consensus decision.

3.2 Sequence-Level Hallucination Quantification and Weight Mapping

After obtaining the candidate set $\mathcal{R}$, RaTI evaluates the factual reliability of each candidate without relying on external knowledge graphs or additional retrieval procedures. Specifically, we employ a lightweight decoder-only Transformer detector with a task-trained binary sequence classification head to estimate sequence-level hallucination severity. The detector directly maps internal causal representations to a response-level factuality judgment and remains frozen throughout all experiments.

For each candidate response R_i, we construct a unified evaluation sequence

$$X_i = [\text{Prompt}_{sys}; C; Q; R_i] = (x_1, x_2, \ldots, x_T), \tag{2}$$

where X_i denotes the evaluation sequence for candidate R_i, Prompt_{sys} is the system instruction, C is the reference context, Q is the input question, and R_i is the response generated by the i-th model. The semicolon denotes ordered concatenation, x_t is the token at position t, and T is the sequence length after tokenization.

Under causal attention, the candidate response is processed after the system instruction, context, and question, allowing its hidden representations to condition on the information used for factual consistency evaluation.

After propagating X_i through the decoder, the final-layer hidden state of the last valid token, $h_T^{(L)} \in \mathbb{R}^{d_h}$, summarizes the preceding sequence:

$$h_T^{(L)} = f_{\text{attn}}(x_T, x_{T-1}, \ldots, x_1), \tag{3}$$

where L denotes the final Transformer layer, d_h is the hidden-state dimension, and $f_{\text{attn}}(\cdot)$ denotes the causal Transformer mapping. This representation provides a compact sequence-level summary without additional global pooling.

The hidden representation $h_T^{(L)}$ is then projected by the trained sequence-level classification head into a two-dimensional logit space:

$$Z_i = [z_{i,0}, z_{i,1}] = (h_T^{(L)})^T W_c, \tag{4}$$

where $Z_i \in \mathbb{R}^2$ denotes the binary logits, $W_c \in \mathbb{R}^{d_h \times 2}$ denotes the learned classification parameters, and $z_{i,0}$ and $z_{i,1}$ correspond to the non-hallucination and hallucination classes, respectively. The

original detector trains this head jointly with a token-level classification objective. RaTI directly uses the released trained parameters without additional fine-tuning; further details are provided in Appendix B.

The hallucination severity score $S_i \in [0, 1]$ is computed as

$$S_i = \frac{\exp(z_{i,1})}{\exp(z_{i,0}) + \exp(z_{i,1})}, \tag{5}$$

where S_i denotes the estimated hallucination risk of candidate R_i. A larger S_i indicates a higher risk of factual inconsistency. We use S_i as an instance-level relative hallucination risk score to derive candidate credibility within each candidate set. The hallucination scores are then transformed into normalized credibility weights:

$$W_i = \frac{(S_i + \epsilon)^{-1}}{\sum_{j=1}^{n} (S_j + \epsilon)^{-1}}, \tag{6}$$

where W_i denotes the credibility weight of candidate R_i, $\epsilon = 10^{-6}$ is a smoothing constant, and j indexes all candidates in $\mathcal{R}$. Because W_i decreases with S_i, candidates with lower hallucination risk receive greater influence in the subsequent consensus decision.

3.3 Weighted Minimum Bayes Risk Decision

Given the candidate set $\mathcal{R}$ and the corresponding credibility weights $\{W_i\}_{i=1}^{n}$, RaTI performs a weighted MBR decision to identify the candidate that best represents the reliable semantic consensus. Conventional consensus decoding methods often rely on lexical overlap measures such as BLEU or ROUGE. However, such metrics may assign large penalties to semantically equivalent responses expressed with different wording, which is undesirable in open-ended question answering.

To measure disagreement at the semantic level, RaTI introduces a **Semantic Differential Loss** based on sentence embeddings. A pre-trained SBERT encoder maps each candidate response R_i to a normalized embedding $\hat{v}_i \in \mathbb{R}^{d_s}$, where $\hat{v}_i$ denotes the semantic embedding of R_i and d_s is the dimensionality of the embedding space. For any pair of candidates R_i and R_j, the semantic differential loss is defined as

$$\mathcal{L}(R_i, R_j) = 1 - \text{Sim}(R_i, R_j) = 1 - \hat{v}_i \cdot \hat{v}_j, \tag{7}$$

where $\mathcal{L}(R_i, R_j)$ denotes the semantic disagreement between R_i and R_j, $\text{Sim}(R_i, R_j)$ denotes their cosine similarity, and $\hat{v}_i \cdot \hat{v}_j$ is the dot product between their normalized embeddings. Because the embeddings are normalized, the dot product is equivalent to cosine similarity.

Semantically similar candidates therefore yield a small loss, whereas strongly dissimilar candidates produce a larger loss. In particular, $\mathcal{L}(R_i, R_j)$ ranges from 0 for identical embedding directions to 2 for opposite directions. This allows RaTI to measure consensus in semantic space rather than relying on exact lexical overlap.

RaTI then combines semantic disagreement with the hallucination-derived credibility weights. For each candidate R_i, the weighted semantic risk is computed as

$$\text{Risk}(R_i) = \sum_{j=1}^{n} W_j \, \mathcal{L}(R_i, R_j), \tag{8}$$

where $\text{Risk}(R_i)$ denotes the expected semantic disagreement associated with selecting candidate R_i, W_j is the credibility weight of candidate R_j, and j indexes all candidates in $\mathcal{R}$. As a result, disagreement with a highly credible candidate contributes more strongly to the overall risk.

The final answer is selected by minimizing the weighted semantic risk over the discrete candidate space:

$$\hat{R} = \arg\min_{R_i \in \mathcal{R}} \text{Risk}(R_i), \tag{9}$$

where $\hat{R}$ denotes the final response selected by RaTI.

This decision rule jointly accounts for *semantic agreement* and *factual reliability*. A candidate receives a lower risk when it is semantically close to other candidates, particularly those with high credibility weights. Conversely, an isolated response receives a larger risk when it disagrees with reliable candidates. In this way, RaTI selects a consensus answer that is supported both semantically and factually. Further analysis of the reliability-weighted MBR formulation is provided in Appendix C.

4 EXPERIMENTS

We conduct extensive experiments on six question answering benchmarks to evaluate RaTI from four perspectives: (1) overall multi-model QA performance, (2) the contribution of individual components and reliability estimation mechanisms, (3) robustness and error characteristics, and (4) autoregressive generation overhead. Detailed dataset statistics are moved to Appendix A to keep the main experimental section focused on the empirical findings.

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate RaTI on six question answering and reading comprehension benchmarks: SQuAD (Rajpurkar et al., 2018), NewsQA (Trischler et al., 2017), RACE (Lai et al., 2017), Relation Extraction(RE) (Levy et al., 2017), DROP (Dua et al., 2019), and the English subset of TYDIQA (Clark et al., 2020). These benchmarks cover factual extraction, relation identification, multi-sentence comprehension, and discrete reasoning.Dataset details are provided in the appendix A.

Base Models. The candidate pool consists of five locally deployable lightweight language models: **Llama3.2-3B**, **Phi3-3.8B**, **Qwen2.5-3B**, **Gemma2-2B**, and **Ministral3-3B**. All models are deployed through Ollama and generate candidate responses independently.

Baselines and Variants. We compare RaTI with individual lightweight models and three representative multi-model collaborative inference methods: LLM-Debate (Du et al., 2024), Mixture-of-Agents (MoA) (Wang et al., 2025a), and SLM-MUX (Wang et al., 2026a). To analyze the effects of reliability estimation and consensus inference, we additionally combine two candidate scoring mechanisms, perplexity-based (**PPL-based**) scoring (Ren et al., 2023) and hidden-state-based (**HS-based**) hallucination scoring, with four truth-inference strategies: Majority Voting (**MV**), CRH (Li et al., 2014), Dawid–Skene (**DS**) (Dawid & Skene, 1979), and Minimum Bayes Risk (**MBR**). Among the evaluated configurations, HS-based hallucination evaluation combined with weighted MBR achieves the strongest overall performance and is used as the default RaTI configuration.

Evaluation Metrics. All methods are evaluated using three complementary metrics: F1, which measures token-level overlap between predictions and reference answers; Exact Match (EM), which requires an exact correspondence with the ground truth; and BLEU, which evaluates n-gram similarity between generated and reference responses.

4.2 OVERALL PERFORMANCE

Table 1 reports the overall performance across the six benchmarks. In addition to individual lightweight models and existing collaborative baselines, we include PPL- and HS-based variants with different truth-inference strategies to examine the effects of reliability estimation and final consensus decision making. All results reported in Table 1 are obtained using deterministic decoding with the temperature set to $T = 0$, except for SLM-MUX, which uses $T = 0.5$ to enable repeated sampling from each individual model.

Overall Effectiveness. RaTI achieves the best F1 performance on five of the six benchmarks and remains competitive on DROP, where the HS-based DS variant obtains the highest score. Compared with the strongest individual lightweight model on each dataset, RaTI improves F1 by 2.9–5.6 points across all six benchmarks. Similar trends are observed for EM and BLEU, indicating that the gains are consistent across different evaluation criteria. These results demonstrate that combining different candidate responses can recover useful information that is not consistently available from any single lightweight model.

Comparison with Collaborative Baselines. RaTI consistently outperforms LLM-Debate, MoA, and SLM-MUX across all six datasets in terms of F1, with gains of 1.9–4.6 points over the strongest collaborative baseline on each benchmark. Unlike debate-based methods or generative aggregation, RaTI performs collaboration directly at the output decision level without requiring iterative inter-model communication or an additional autoregressive synthesis stage. The results therefore suggest that jointly exploiting candidate reliability and semantic agreement provides a more effective collaboration mechanism for lightweight language models.

Effect of Reliability Estimation. The comparison between PPL and HS based variants further shows that candidate-level reliability estimation is important for collaborative inference. Under both

Table 1: Experimental results on six question answering benchmarks. Single underline indicates the best result among single lightweight models, and **bold** indicates the overall best result.

Model	SQuAD			Relation Extraction			TYDIQA			NewsQA			RACE			DROP		
	F1	EM	BLEU	F1	EM	BLEU	F1	EM	BLEU	F1	EM	BLEU	F1	EM	BLEU	F1	EM	BLEU
Single Lightweight Models																		
Gemma2	72.1	56.0	63.9	78.4	62.4	71.5	53.0	35.5	44.8	49.6	32.1	40.5	48.9	31.0	39.5	54.5	39.9	47.6
Llama	79.1	67.0	73.3	83.6	71.2	78.7	62.2	48.6	55.6	54.2	37.8	46.6	51.4	37.8	45.3	51.8	43.4	48.3
Phi	82.3	67.6	78.9	83.7	71.8	81.1	63.8	45.8	58.2	54.3	34.2	48.7	55.9	38.9	51.7	58.0	47.3	55.2
Ministral	79.4	67.5	74.4	84.6	74.7	80.9	58.5	44.4	51.1	53.6	36.3	45.6	54.8	39.6	48.9	58.0	49.7	55.0
Qwen	71.0	59.2	66.0	80.1	70.6	76.5	54.6	41.7	47.9	50.8	34.8	43.8	48.9	35.1	43.3	52.8	43.5	49.2
Multi-Model Collaboration Baselines																		
LLM-Debate	80.3	68.6	75.3	82.0	70.6	78.0	60.1	45.6	53.3	54.8	38.0	47.4	56.5	40.9	50.1	61.7	50.9	57.3
MoA	73.0	59.9	67.9	75.6	66.7	72.2	55.1	38.8	48.4	52.3	33.1	45.1	48.5	35.0	42.8	53.0	43.3	48.9
SLM-MUX	80.6	66.5	76.3	84.8	73.4	81.6	65.3	50.4	59.6	56.1	36.5	49.3	53.0	36.5	47.0	54.6	43.0	50.5
RaTI: PPL-based Truth Inference (Ours)																		
MV	80.2	69.0	74.5	86.8	76.7	82.8	61.6	48.3	54.7	54.6	38.8	46.9	55.4	40.8	48.3	61.1	52.8	57.6
CRH	82.6	71.8	77.4	87.6	78.2	83.9	63.6	49.9	56.7	53.4	37.3	45.3	55.7	41.4	49.0	62.1	52.5	58.1
DS	83.7	73.1	79.2	87.7	78.6	84.2	65.0	51.2	58.3	57.5	40.5	50.0	58.7	44.4	52.5	64.0	55.1	60.6
MBR	82.7	71.4	79.2	87.0	78.0	84.1	65.8	51.6	59.8	57.5	39.3	51.0	57.8	43.3	53.0	62.0	53.2	59.4
RaTI: HS-based Truth Inference (Ours)																		
MV	84.5	73.5	80.3	87.9	79.4	84.9	67.5	53.5	61.2	58.5	40.9	51.4	58.7	43.8	53.1	62.1	53.4	58.8
CRH	82.6	71.8	77.4	87.3	78.1	83.7	63.6	49.9	56.7	54.9	38.2	46.9	56.5	42.4	50.1	62.1	52.5	58.1
DS	84.0	72.9	79.1	87.7	78.5	84.2	65.0	51.1	58.1	57.6	40.6	49.9	58.6	43.8	52.3	**64.4**	**55.6**	**60.8**
MBR	**85.2**	**73.8**	**81.0**	**88.2**	**79.7**	**85.2**	**67.9**	**53.7**	**61.6**	**59.0**	**41.2**	**51.8**	**58.8**	**43.9**	**53.3**	63.6	54.5	60.3

MV and MBR, replacing PPL with the hidden-state-based hallucination score improves F1 on all six benchmarks. The improvement is particularly pronounced for MV, for example increasing F1 from 80.2 to 84.5 on SQuAD and from 61.6 to 67.5 on TYDIQA. These results indicate that the hidden-state estimator provides a more informative instance-level reliability signal than generation likelihood alone.

4.3 Component and Reliability Analysis

We next investigate where the improvements of RaTI originate. Specifically, we examine the contribution of its major components, the complementarity of the different candidate models, and the quality of the hallucination-based reliability signal.

Component Contribution. Figure 3 compares the major configurations of RaTI by averaging their performance across the six benchmarks. The complete framework achieves the strongest overall F1, EM, and BLEU, showing that candidate reliability estimation and semantic consensus provide complementary benefits. In particular, the reliability signal reduces the influence of unreliable responses, while weighted MBR further exploits semantic agreement among candidates.

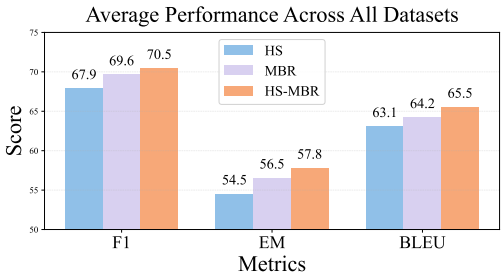


Figure 3: Average ablation results across six QA benchmarks.

The consistent improvement of the full framework indicates that RaTI's gains cannot be attributed solely to candidate aggregation. Instead, explicitly incorporating factual reliability into the consensus process is important for stable multi-model inference.

Reliability Score Quality. Figure 4 evaluates whether the reliability signal used by RaTI can effectively distinguish reliable from unreliable candidate responses. We label a candidate as unreliable if it fails EM against the ground-truth answer, and reliable otherwise. The HS estimator achieves an ROC-AUC of 0.728, compared with 0.667 for PPL. Moreover, the Precision@K and Recall@K results show that HS provides stronger ranking capability for identifying high-risk candidate responses, particularly when only a small proportion of candidates are inspected. This result is consistent with Table 1, where HS-based variants generally outperform their PPL-based counterparts under the same aggregation strategy. Together, these results indicate that hidden-state representations provide a more informative instance-level reliability signal for multi-model consensus.

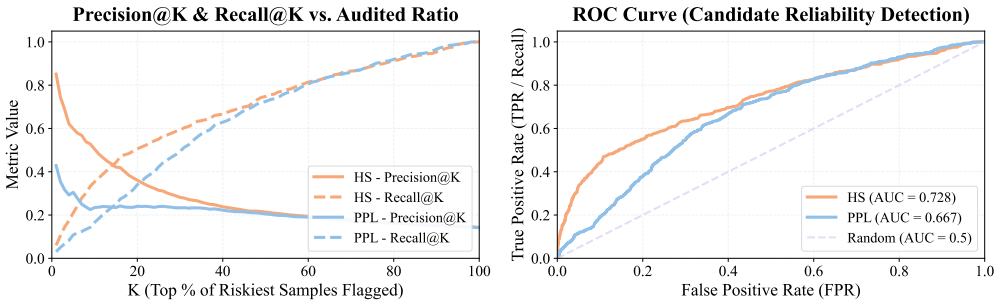


Figure 4: Reliability evaluation of candidate scoring methods. Left: Precision@K and Recall@K for ranking high-risk candidate responses under different auditing ratios; right: ROC curves for hallucination detection using HS based and PPL based reliability scores.

Contribution of Different Models. Figure 5 examines whether the performance of RaTI is dominated by a particular member of the candidate pool. Removing individual models produces substantially different effects across datasets, and no single model consistently dominates all tasks. This dataset-dependent sensitivity suggests that the participating lightweight models provide complementary rather than redundant information. In particular, the contribution of a model appears to depend on the characteristics of the target dataset, indicating that different models may capture distinct reasoning patterns or factual evidence. This also helps explain why combining different models can provide a more stable candidate space than relying on a single model family.

Interestingly, removing a model occasionally leads to a small performance improvement. This indicates that increasing candidate diversity can also introduce noisy or unreliable responses, further motivating the reliability-aware weighting mechanism used by RaTI. Rather than treating all candidates equally, RaTI can reduce the influence of less reliable responses while preserving the complementary information contributed by different models.

4.4 Robustness and Error Analysis

We further examine the behavior of RaTI when candidate quality deteriorates and analyze the major sources of its remaining errors.

Robustness to Candidate Errors. Figure 6 evaluates collaborative methods as the proportion of incorrect responses among the five candidates increases. RaTI remains relatively stable in the moderate-error regime, since its decision considers both candidate reliability and semantic agreement rather than response frequency alone, so unreliable majority opinions can exert less influence than under conventional voting-based aggregation. However, when roughly 80%–100% of the candidate pool becomes incorrect, all methods degrade substantially. This highlights a fundamental limitation of output-level collaboration, in that reliable consensus cannot be recovered when the candidate set contains almost no useful evidence.

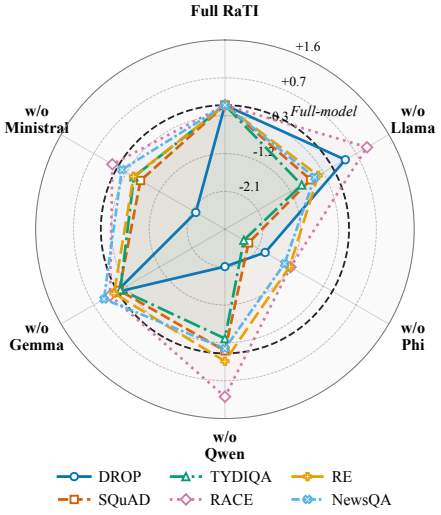


Figure 5: Model contribution sensitivity across datasets. Each axis reports the F1 change after removing one lightweight model from the full five-model RaTI candidate pool.

Error Decomposition. Figure 7 decomposes unsuccessful predictions into candidate, scoring, and consensus failures. Candidate failure is the dominant error source across all six datasets, accounting for 52.1%–81.2% of the remaining failures, whereas consensus failure accounts for only 5.8%–13.4%.These results indicate that the principal bottleneck lies upstream of the final MBR decision. Future improvements are therefore more likely to come from increasing candidate coverage and improving reliability estimation than from introducing a more complex consensus algorithm.

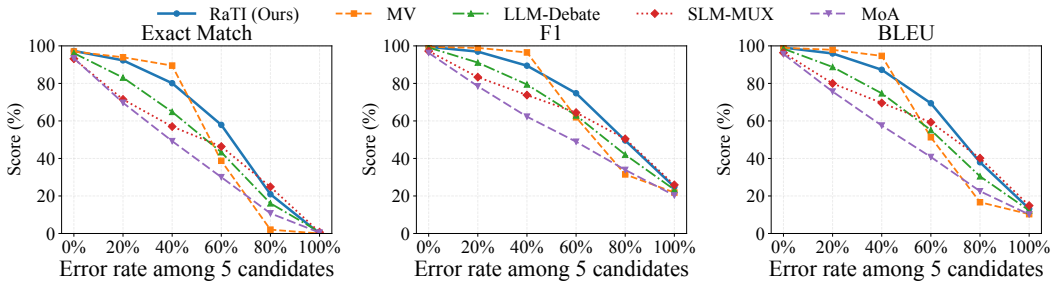


Figure 6: Robustness to increasing candidate error rates. We progressively increase the proportion of incorrect responses among the five model candidates and report Exact Match, F1, and BLEU for RaTI, MV, LLM-Debate, SLM-MUX, and MoA.

4.5 Autoregressive Token Consumption Analysis

We compare the autoregressive generation overhead of different collaborative inference methods by reporting the average number of generated output tokens per question.

As shown in Table 2, RaTI requires only one response from each candidate model and introduces no additional autoregressive generation, resulting in the same generated-token consumption as majority voting. In contrast, MoA, SLM-MUX, and LLM-Debate require additional generation for aggregation, repeated sampling, or multi-round interaction.

4.6 Discussion

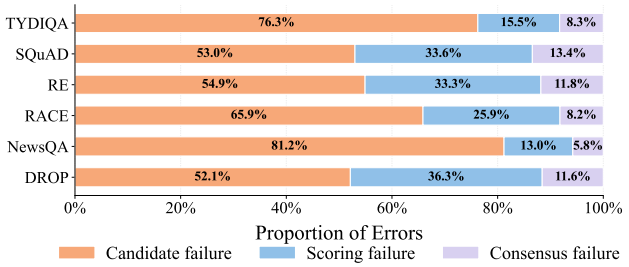


Figure 7: Error decomposition of RaTI across the six datasets. Remaining errors are categorized into candidate failure, scoring failure, and consensus failure.

Overall, the experimental results indicate that RaTI's effectiveness stems from using instance-level hallucination evaluation to constrain multi-model truth inference. Component analysis shows that combining reliability estimation with weighted MBR outperforms either component alone, and hidden-state-based scoring provides a more informative reliability signal than perplexity-based scoring. Model ablation reveals complementary, dataset-dependent contributions from individual models. Robustness analysis confirms RaTI remains effective with a moderate proportion of incorrect candidates, though performance degrades when reliable evidence is scarce, consistent with error decomposition where candidate failure dominates remaining errors and consensus failure plays a smaller role.

5 Conclusion

RaTI provides a reliability-aware approach to multi-model question answering by formulating collaborative answer selection as hallucination-evaluation-constrained truth inference. The results across six QA benchmarks suggest that this design can improve the reliability of collaborative inference over individual lightweight models and several representative aggregation strategies. A current limitation is that RaTI still depends on the quality of the candidate pool and may fail when correct evidence is absent or when correlated errors receive insufficiently low reliability scores. Future work could therefore focus on improving candidate coverage, reliability calibration, and the modeling of correlated errors among participating models.

Table 2: Average autoregressive generated-token consumption per question.

Method	Tokens	Extra	Ratio
MV	1943.4	0	1.00×
RaTI	**1943.4**	**0**	**1.00×**
MoA	2452.4	509.0	1.26×
SLM-MUX	5830.6	3887.2	3.00×
LLM-Debate	6210.1	4266.7	3.20×

AI USE STATEMENT

During manuscript preparation, generative AI tools were used to assist with language refinement, code prototyping and debugging, finding related work, and improving the paper's organization and presentation. All related work found with AI assistance was independently verified by the authors to ensure it actually exists and the bibliographic information is accurate. All experiments were executed and checked by the authors, and all numerical results, analyses, and scientific claims were manually verified before inclusion. AI-generated suggestions were treated only as draft material and not as independent evidence. The authors made all final decisions regarding the methodology, experimental design, interpretation of results, and manuscript content, and take full responsibility for the paper.

ETHICS STATEMENT

We confirm that this work complies with the ICLR Code of Ethics. It studies factual reliability in multi-model question answering using publicly available QA benchmarks and locally deployable language models. The study involves no newly collected personal data, human-subject experiments, or decisions affecting individuals. All datasets and models should be used under their respective licenses and terms. We have considered potential issues related to misuse, fairness, bias, and privacy, and to the best of our knowledge, the work meets applicable ethical standards.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide the implementation code and experimental scripts at https://anonymous.4open.science/r/crowdsourcing-QA. The main experimental setup, including datasets, candidate models, baselines, decoding settings, and evaluation metrics, is described in Section 4.1. Detailed dataset statistics, model configurations, truth inference variants, and implementations of collaborative baselines are provided in Appendix A. Details of the hallucination detector and perplexity-based reliability estimation are provided in Appendix B, while additional experimental settings and analyses are reported in Appendix D.

REFERENCES

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. ChatEval: Towards Better LLM-based Evaluators through Multi-Agent Debate. In *ICLR*, 2024.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. INSIDE: LLMs' Internal States Retain the Power of Hallucination Detection. In *ICLR*, pp. 3056–3076, 2024a.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to Use Large Language Models While Reducing Cost and Improving Performance. *TMLR*, 2024b. Featured Certification.

Shiqi Chen, Miao Xiong, Junteng Liu, Zhengxuan Wu, Teng Xiao, Siyang Gao, and Junxian He. In-Context Sharpness as Alerts: An Inner Representation Perspective for Hallucination Mitigation. In *ICML*, volume 235, pp. 7553–7567, 2024c.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. TyDi QA: A Benchmark for Information-Seeking Question Answering in Typologically Diverse Languages. *TACL*, 8:454–470, 2020.

Alexander Philip Dawid and Allan M. Skene. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28, 1979.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving Factuality and Reasoning in Language Models through Multiagent Debate. In *ICML*, pp. 11733–11763, 2024.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. DROP: A Reading Comprehension Benchmark Requiring Discrete Reasoning Over Paragraphs. In *ACL*, volume 1, pp. 2368–2378, 2019.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting Hallucinations in Large Language Models Using Semantic Entropy. *Nature*, 630(8017):625–630, 2024.

Jiatong Han, Neil Band, Muhammed Razzak, Jannik Kossen, Tim GJ Rudner, and Yarin Gal. Simple Factuality Probes Detect Hallucinations in Long-Form Natural Language Generation. In *Findings of ACL*, pp. 16209–16226, 2025.

Yichong Huang, Xiaocheng Feng, Baohang Li, Yang Xiang, Hui Wang, Ting Liu, and Bing Qin. Ensemble Learning for Heterogeneous Large Language Models with Deep Parallel Collaboration. In *NeurIPS*, volume 37, pp. 119838–119860, 2024.

Youwon Jang, Woo Suk Choi, Minjoon Jung, Minsu Lee, and Byoung-Tak Zhang. Confidence-Guided Refinement Reasoning for Zero-Shot Question Answering. In *EMNLP*, pp. 6933–6950, 2025.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.*, 55(12):248:1–248:38, 2023.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. Llm-Blender: Ensembling Large Language Models with Pairwise Ranking and Generative Fusion. In *ACL*, volume 1, pp. 14165–14178, 2023.

Hazel Kim, Tom A. Lamb, Adel Bibi, Philip Torr, and Yarin Gal. Detecting LLM Hallucination Through Layer-wise Information Deficiency: Analysis of Ambiguous Prompts and Unanswerable Questions. In *EMNLP*, pp. 32310–32322, 2025.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. RACE: Large-scale ReAding Comprehension Dataset From Examinations. In *EMNLP*, pp. 785–794, 2017.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-Shot Relation Extraction via Reading Comprehension. In *CoNLL*, pp. 333–342, 2017.

Qi Li, Yaliang Li, Jing Gao, Bo Zhao, Wei Fan, and Jiawei Han. Resolving Conflicts in Heterogeneous Data by Truth Discovery and Source Reliability Estimation. In *ACM SIGMOD*, pp. 1187–1198, 2014.

Yaliang Li, Jing Gao, Chuishi Meng, Qi Li, Lu Su, Bo Zhao, Wei Fan, and Jiawei Han. A Survey on Truth Discovery. *ACM SIGKDD Explorations Newsletter*, 17(2):1–16, 2016.

Potsawee Manakul, Adian Liusie, and Mark Gales. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. In *EMNLP*, pp. 9004–9017, 2023.

Xupeng Miao, Gabriele Oliaro, Zhihao Zhang, Xinhao Cheng, Hongyi Jin, Tianqi Chen, and Zhihao Jia. Towards Efficient Generative Large Language Model Serving: A Survey from Algorithms to systems. *ACM Comput. Surv.*, 58(1), 2025.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models. In *ACL*, volume 1, pp. 10862–10878, 2024.

Bibek Paudel, Alexander Lyzhov, Preetam Joshi, and Puneet Anand. HalluciNot: Hallucination Detection through Context and Common Knowledge Verification. *arXiv preprint arXiv:2504.07069*, 2025.

Libo Qin, Qiguang Chen, Xiachong Feng, Yang Wu, Yongheng Zhang, Yinghui Li, Min Li, Wanxiang Che, and Philip S Yu. Large language models meet NLP: A survey. *Frontiers of Computer Science*, 20(11):2011361, 2026.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know What You Don't Know: Unanswerable Questions for SQuAD. In *ACL*, volume 2, pp. 784–789, 2018.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J Liu. Out-of-Distribution Detection and Selective Generation for Conditional Language Models. In *ICLR*, 2023.

Gaurang Sriramanan, Siddhant Bharti, Vinu Sankar Sadasivan, Shoumik Saha, Priyatham Kattakinda, and Soheil Feizi. LLM-check: Investigating detection of hallucinations in large language models. In *NeurIPS*, 2024.

Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. Unsupervised Real-Time Hallucination Detection based on the Internal States of Large Language Models. In *Findings of ACL*, pp. 14379–14391.

Hexiang Tan, Fei Sun, Sha Liu, Du Su, Qi Cao, Xin Chen, Jingang Wang, Xunliang Cai, Yuanzhuo Wang, Huawei Shen, and Xueqi Cheng. Too Consistent to Detect: A Study of Self-Consistent Errors in LLMs. In *EMNLP*, pp. 4755–4765, 2025.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. Confidence Improves Self-Consistency in LLMs. In *Findings of ACL*, pp. 20090–20111, 2025.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. NewsQA: A Machine Comprehension Dataset. In *RepL4NLP (ACL Workshop)*, pp. 191–200, 2017.

Chenyu Wang, Zishen Wan, Hao Kang, Emma Chen, Zhiqiang Xie, Tushar Krishna, Vijay Janapa Reddi, and Yilun Du. SLM-MUX: Orchestrating Small Language Models for Reasoning. In *ICLR*, 2026a.

Cunxiang Wang, Xiaoze Liu, Yuanhao Yue, Qipeng Guo, Xiangkun Hu, Xiangru Tang, Tianhang Zhang, Cheng Jiayang, Yunzhi Yao, Xuming Hu, Zehan Qi, Wenyang Gao, Yidong Wang, Linyi Yang, Jindong Wang, Xing Xie, Zheng Zhang, and Yue Zhang. Survey on Factuality in Large Language Models. *ACM Comput. Surv.*, 58(1):13:1–13:37, 2026b.

Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Y Zou. Mixture-of-Agents Enhances Large Language Model Capabilities. In *ICLR*, pp. 33944–33963, 2025a.

Weiqin Wang, Yile Wang, and Hui Huang. Ranked Voting based Self-Consistency of Large Language Models. In *Findings of ACL*, pp. 14410–14426, 2025b.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *ICLR*, 2023.

Yiming Wang, Pei Zhang, Baosong Yang, Derek Wong, and Rui Wang. Latent Space Chain-of-Embedding Enables Output-free LLM Self-Evaluation. In *ICLR*, pp. 70938–70970, 2025c.

Kai Xiong, Xiao Ding, Yixin Cao, Ting Liu, and Bing Qin. Examining Inter-Consistency of Large Language Models Collaboration: An In-depth Analysis via Debate. In *Findings of ACL*, pp. 7572–7590, 2023.

Dongyu Zhang, Qingqing Hong, Bingxuan Hou, Jiayi Lin, Chenyang Zhang, Jialin Li, and Junli Wang. Bold Claims or Self-Doubt? Factuality Hallucination Type Detection via Belief State. In *Findings of ACL*, pp. 9946–9959, 2025a.

Fujie Zhang, Peiqi Yu, Biao Yi, Baolei Zhang, Tong Li, and Zheli Liu. Prompt-Guided Internal States for Hallucination Detection of Large Language Models. In *ACL*, volume 1, pp. 21806–21818, 2025b.

Zhenliang Zhang, Xinyu Hu, Huixuan Zhang, Junzhe Zhang, and Xiaojun Wan. Icr Probe: Tracking Hidden State Dynamics for Reliable Hallucination Detection in LLMs. In *ACL*, volume 1, pp. 17986–18002, 2025c.

Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. Truth Inference in Crowdsourcing: Is the Problem Solved? *PVLDB*, 10(5):541–552, 2017.

A Datasets, Models and Implementation Details

We evaluate RaTI on six question answering benchmarks covering different types of reading comprehension and reasoning capabilities. These datasets are selected to provide a diverse evaluation environment, including extractive QA, long-context comprehension, multi-hop reasoning, relation extraction, and discrete numerical reasoning. The detailed statistics are summarized in Table 3.

Table 3: Statistics of the experimental datasets.

Dataset	Domain	Task Characteristics	Samples
SQuAD (Rajpurkar et al., 2018)	Wikipedia articles	Extractive factual QA	10,570
NewsQA (Trischler et al., 2017)	News documents	Long-context comprehension	4,212
RACE (Lai et al., 2017)	English examinations	Multi-sentence reasoning	674
Relation Extraction (Levy et al., 2017)	Structured text	Relation identification	2,948
DROP (Dua et al., 2019)	Wikipedia paragraphs	Discrete numerical reasoning	1,503
TYDIQA-en (Clark et al., 2020)	Diverse information sources	Cross-domain QA	4,136
Total	–	–	**24,043**

A.1 Dataset Description

SQuAD. SQuAD is an extractive question answering benchmark constructed from Wikipedia articles. Each example contains a passage, a question, and an answer span extracted from the passage. It evaluates whether models can accurately identify factual information from given contexts.

NewsQA. NewsQA consists of questions generated from news articles. Compared with SQuAD, it requires stronger contextual understanding because answers are often distributed across multiple sentences rather than directly appearing as single spans.

RACE. RACE is collected from English examinations and requires models to select correct answers from multiple candidates. The dataset emphasizes multi-sentence reasoning and comprehension ability.

Relation Extraction. Following the reading-comprehension formulation of relation extraction, each instance requires identifying the relation between entities from textual contexts. This dataset evaluates whether models can correctly infer structured information.

DROP. DROP focuses on discrete reasoning over paragraphs. Unlike conventional span extraction tasks, solving DROP often requires arithmetic operations and reasoning over multiple pieces of information.

TYDIQA-en. TYDIQA is designed to evaluate question answering across diverse linguistic and cultural backgrounds. We use the English subset to evaluate robustness under varied question distributions.

A.2 Model Description

RaTI employs five different lightweight language models for candidate generation, including Llama3.2-3B[1], Phi3-3.8B[2], Qwen2.5-3B[3], Gemma2-2B[4], and Ministral3-3B[5]. These models are selected from different model families to construct a diverse candidate answer space while maintaining relatively low inference costs. All candidate models are deployed locally through Ollama and independently generate one response for each context–question pair. In addition to the candidate generators, RaTI uses a Qwen2.5-3B-based hallucination detector for candidate-level reliability

[1] https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct
[2] https://huggingface.co/microsoft/Phi-3-mini-4k-instruct-gguf
[3] https://huggingface.co/Qwen/Qwen2.5-3B
[4] https://huggingface.co/google/gemma-2-2b
[5] https://huggingface.co/mistralai/Ministral-3-3B-Instruct-2512

Table 4: Models used in the RaTI framework and their corresponding roles.

Model	Scale	Role	Module
Llama3.2-3B	3B	Candidate generation	Collaborative sampling
Phi3-3.8B	3.8B	Candidate generation	Collaborative sampling
Qwen2.5-3B	3B	Candidate generation	Collaborative sampling
Gemma2-2B	2B	Candidate generation	Collaborative sampling
Ministral3-3B	3B	Candidate generation	Collaborative sampling
Qwen2.5-3B	3B	Hallucination detection	Reliability estimation
all-MiniLM-L6-v2	–	Sentence embedding	Weighted MBR

estimation and all-MiniLM-L6-v2 as the semantic encoder for weighted MBR decoding. The models and their roles are summarized in Table 4.

Candidate Generation Models. The candidate pool consists of Llama3.2-3B, Phi3-3.8B, Qwen2.5-3B, Gemma2-2B, and Ministral3-3B. Each model independently receives the same context and question and produces one candidate response without observing the outputs of the other models. The different model pool is designed to capture complementary generation behaviors and failure patterns across different lightweight model families. RaTI does not assign fixed reliability according to model identity; instead, the reliability of each response is estimated separately for the current input.

Hallucination Detector. For candidate-level factual reliability estimation, we use a sequence-level hallucination detector built upon Qwen2.5-3B as its backbone model. For each candidate response R_i, the detector receives the concatenated system prompt, context, question, and response, $X_i = [\mathrm{Prompt}_{sys}; C; Q; R_i]$. Following the formulation described in Section 3.2, the hidden representation of the last valid token from the final decoder layer is passed to a binary classification head to estimate the hallucination probability $P(H_1 \mid X_i)$. The resulting score is subsequently converted into a credibility weight used by the consensus module. The hallucination detector is used only for sequence-level reliability estimation and does not generate or modify candidate answers.

Semantic Encoder. For semantic consensus estimation, we use `all-MiniLM-L6-v2` as the sentence embedding model. Each candidate response is independently encoded into a normalized sentence-level representation. Pairwise semantic differential loss is then computed using cosine distance between candidate embeddings, as described in Section 3.3. These semantic distances are combined with the hallucination-derived credibility weights to compute the weighted minimum Bayes risk of each candidate. The semantic encoder is used only for representation extraction and introduces no additional autoregressive generation.

Implementation Protocol. All five candidate language models generate their responses independently under the same input formulation. RaTI performs no inter-model communication during candidate generation and does not require an additional generative coordinator. After the five responses are obtained, the Qwen2.5-3B-based hallucination detector estimates their instance-level factual reliability, while `all-MiniLM-L6-v2` measures pairwise semantic agreement. The final answer is selected from the original candidate set using the reliability-weighted MBR objective.

A.3 Implementation of Truth-Inference Variants

We additionally implement three representative truth-inference strategies, namely Majority Voting (MV), CRH, and Dawid–Skene (DS), for comparison with the proposed weighted MBR decoder.

Majority Voting (MV). MV selects the answer supported by the largest number of candidate responses. Candidates with identical answer strings are grouped together, and the answer with the highest vote count is selected as the final prediction. When multiple answers receive the same number of votes, the candidate with the highest credibility is selected as the tie-breaking result. The credibility of candidate R_i is derived from its hallucination score S_i using the normalized inverse mapping.

CRH. We adapt Conflict Resolution on Heterogeneous Data (CRH) to multi-model QA by treating each language model as a source. Source weights are initialized from the average hallucination scores, with lower-risk models receiving larger initial weights. At each iteration, the disagreement between each source and the current truth estimates is computed, and the source weights are updated according to the normalized average disagreement:

$$w_m = -\log(\bar{e}_m + \epsilon), \tag{10}$$

where $\bar{e}_m$ denotes the normalized average error of source m. The truth estimates are then updated using weighted voting for textual answers and a weighted median for numerical answers. The procedure terminates upon convergence or after at most 15 iterations.

Dawid–Skene (DS). We use an instance-conditioned one-coin Dawid–Skene variant adapted to multi-model QA aggregation. For each question, each model is associated with an accuracy parameter θ_m, initialized from the corresponding candidate-level hallucination score. An EM procedure then alternates between estimating the posterior distribution over candidate answers and updating the model accuracy parameters with Beta-prior smoothing. The candidate with the highest posterior probability is selected as the final answer. We use a convergence tolerance of 10^{-5} and at most 30 EM iterations.

A.4 Implementation Details of Collaborative Baselines.

LLM-Debate. We implement LLM-Debate using the same five lightweight models as participating agents. The independently generated candidate responses are used as the initial-round answers. We conduct two debate rounds in total, where each agent is provided with the latest responses from the other agents together with the original context and question, and is asked to refine its prediction. The final answer is determined by majority voting over the agents' responses after the last round.

Mixture-of-Agents (MoA). We implement MoA using a two-layer aggregation procedure. In the first layer, the five lightweight models provide independent candidate responses. In the second layer, these responses, together with the original context and question, are provided to a `mistral-7B`[6] aggregator, which synthesizes them into a single final answer. The same answer-format constraints are applied to the proposer and aggregation stages.

SLM-MUX. For SLM-MUX, each lightweight model produces three independently sampled responses. Within each model, majority voting over the three responses determines its candidate answer, and the agreement ratio is used as the model confidence. The final answer is selected from the model with the highest confidence; ties are resolved using the average hallucination-based credibility score.

B Details of the Hallucination Detector

B.1 Hidden-State-Based Hallucination Detection

RaTI employs the pretrained HDM-2 detector Paudel et al. (2025) as an external factual-risk estimator. HDM-2 uses Qwen2.5-3B-Instruct as its backbone and incorporates both token-level and sequence-level binary classification heads for context-based hallucination detection. The detector is pretrained independently of RaTI, and we directly use the released parameters without any additional fine-tuning on our evaluation datasets.

The context-based detector is trained using a multi-task objective that jointly optimizes token-level and sequence-level classification. Let

$$H = [h_1, h_2, \ldots, h_T] \tag{11}$$

denote the hidden representations produced by the backbone. The two prediction tasks can be generally expressed as

$$Z_t^{\text{tok}} = W_{\text{tok}} h_t + b_{\text{tok}}, \tag{12}$$

[6] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3

and

$$Z^{\mathrm{seq}} = W_{\mathrm{seq}} h_{\mathrm{seq}}, \tag{13}$$

where W_{tok} and W_{seq} denote the learned token-level and sequence-level classification parameters, respectively.

Accordingly, the training objective can be written in a general form as

$$\mathcal{L} = \lambda_{\mathrm{tok}} \mathcal{L}_{\mathrm{tok}} + \lambda_{\mathrm{seq}} \mathcal{L}_{\mathrm{seq}}. \tag{14}$$

The original study specifies the joint token- and sequence-classification objective but does not report all optimization details, such as the exact loss weights, learning rate, batch size, or number of training epochs. Therefore, RaTI does not attempt to reconstruct or retrain the detector.

According to the released configuration, the detector also adopts parameter-efficient adaptation with rank $r = 8$, scaling factor $\alpha = 16$, and dropout 0.1. These parameters are part of the pretrained detector and remain fixed in all RaTI experiments.

Training Data and Annotation. The context-based hallucination detector is trained on RagTruth Niu et al. (2024) and HDMBENCH[7], a diverse benchmark for context-grounded hallucination detection. Crucially, to prevent data leakage, the detector's training set does not overlap with the evaluation data in RaTI, and none of the test samples from the six QA benchmarks are included during training or fine-tuning.

The generated responses are annotated at both sentence and phrase levels using three categories:

- *supported by context*,
- *supported by general knowledge but not explicitly stated in the context*, and
- *hallucination*.

The original HDM-2 study uses DeepSeek-R1 to produce categorical judgments, supporting rationales, and span-level annotations, followed by filtering and quality control. A subset of the resulting annotations is additionally subjected to human validation. When official training, validation, and test splits are available, the original splits are retained; otherwise, the dataset is divided into training, validation, and test subsets.

Sequence-Level Hallucination Score. In RaTI, only the sequence-level output of the pretrained detector is used. For a candidate response R_i, the detector obtains the hidden representation associated with the last valid token,

$$h_{i,T}^{(L)} \in \mathbb{R}^d, \tag{15}$$

and projects it into a two-dimensional logit space:

$$Z_i = (h_{i,T}^{(L)})^T W_c = [z_{i,0}, z_{i,1}], \tag{16}$$

where W_c denotes the learned sequence-level binary classification head.

The hallucination risk score is then computed as

$$S_i = \frac{\exp(z_{i,1})}{\exp(z_{i,0}) + \exp(z_{i,1})}. \tag{17}$$

A larger S_i indicates a higher estimated hallucination risk. Since the released inference procedure applies Softmax directly to the classification logits without an additional probability-calibration stage, we treat S_i as a relative hallucination risk score rather than an absolutely calibrated probability of factual incorrectness.

Reproducibility and Scope. All candidate responses in RaTI are evaluated using the same frozen HDM-2 detector. Thus, the hallucination scores are generated under a fixed evaluation protocol and are not affected by task-specific retraining.

The original HDM-2 work publicly reports its backbone model, training datasets, annotation procedure, multi-task formulation, and released model weights and inference implementation. However, several low-level training hyperparameters are not fully specified. Consequently, our experiments focus on reproducible inference with the released detector rather than reproducing its original training process from scratch.

[7] https://huggingface.co/datasets/dataframer/HDM-Bench

B.2 Perplexity-Based Hallucination Detection

For the PPL-based variants, we use an autoregressive language model to estimate the generation likelihood of each candidate response. To handle long contexts, the reference context is divided into overlapping token chunks using a sliding window. For each chunk C_k, we construct the input

$$X_{i,k} = [C_k; Q; R_i],$$

and compute the average negative log-likelihood only over the tokens of candidate response R_i:

$$\mathrm{NLL}_{i,k} = -\frac{1}{|R_i|} \sum_{t=1}^{|R_i|} \log p(r_{i,t} \mid C_k, Q, r_{i,<t}). \tag{18}$$

The corresponding perplexity is

$$\mathrm{PPL}_{i,k} = \exp(\mathrm{NLL}_{i,k}). \tag{19}$$

The final PPL score of candidate R_i is defined as the minimum perplexity across all context chunks:

$$S_i^{\mathrm{PPL}} = \min_k \mathrm{PPL}_{i,k}\,. \tag{20}$$

This minimum-over-chunks strategy allows a candidate to be evaluated against the most relevant local context while reducing the effect of irrelevant portions of long inputs. Lower PPL indicates that the candidate is more strongly supported by the evaluator model under at least one context chunk. In our implementation, adjacent chunks overlap to reduce information loss near chunk boundaries.

C Theoretical Analysis of Reliability-Weighted MBR

We provide a theoretical interpretation of how hallucination-based reliability estimation affects the MBR decision.

Let

$$\mathcal{R} = \{R_1, \ldots, R_n\}$$

denote the candidate response set for an input X. For each candidate R_j, let

$$\rho_j = P(Z_j = 1 \mid X, R_j)$$

denote its latent factual reliability, where $Z_j = 1$ indicates that R_j is factually reliable. Based on these unknown reliabilities, we define the oracle reliability distribution

$$p_j^* = \frac{\rho_j}{\sum_{k=1}^n \rho_k}, \qquad \sum_{j=1}^n p_j^* = 1. \tag{21}$$

Let

$$W = (W_1, \ldots, W_n), \qquad \sum_{j=1}^n W_j = 1,$$

denote the reliability distribution estimated from hallucination scores. Thus, W serves as an approximation to the unknown oracle distribution p^*.

For a semantic loss $\mathcal{L}(R_i, R_j)$ satisfying

$$0 \le \mathcal{L}(R_i, R_j) \le B, \tag{22}$$

define the expected semantic risk under the oracle reliability distribution as

$$\mathcal{R}_{p^*}(R_i) = \sum_{j=1}^n p_j^* \mathcal{L}(R_i, R_j), \tag{23}$$

and the corresponding risk under the estimated reliability distribution as

$$\mathcal{R}_W(R_i) = \sum_{j=1}^{n} W_j \mathcal{L}(R_i, R_j). \tag{24}$$

Let

$$R^* = \arg\min_{R_i \in \mathcal{R}} \mathcal{R}_{p^*}(R_i) \tag{25}$$

denote the response selected using the oracle reliability distribution, and let

$$\widehat{R} = \arg\min_{R_i \in \mathcal{R}} \mathcal{R}_W(R_i) \tag{26}$$

denote the response selected using the hallucination-derived reliability distribution.

Theorem 1. The excess oracle risk of $\widehat{R}$ is bounded by

$$\mathcal{R}_{p^*}(\widehat{R}) - \mathcal{R}_{p^*}(R^*) \le 2B\,\mathrm{TV}(p^*, W), \tag{27}$$

where

$$\mathrm{TV}(p^*, W) = \frac{1}{2}\sum_{j=1}^{n} |p_j^* - W_j| \tag{28}$$

denotes the total variation distance between the oracle and estimated reliability distributions.

Proof. For any candidate R_i, because the semantic loss is bounded by Eq. equation 22, the difference between its expected risks under p^* and W satisfies

$$|\mathcal{R}_{p^*}(R_i) - \mathcal{R}_W(R_i)| \le B\,\mathrm{TV}(p^*, W). \tag{29}$$

Applying Eq. equation 29 to $\widehat{R}$ gives

$$\mathcal{R}_{p^*}(\widehat{R}) \le \mathcal{R}_W(\widehat{R}) + B\,\mathrm{TV}(p^*, W). \tag{30}$$

Since $\widehat{R}$ minimizes $\mathcal{R}_W$,

$$\mathcal{R}_W(\widehat{R}) \le \mathcal{R}_W(R^*). \tag{31}$$

Applying Eq. equation 29 again to R^* yields

$$\mathcal{R}_W(R^*) \le \mathcal{R}_{p^*}(R^*) + B\,\mathrm{TV}(p^*, W). \tag{32}$$

Combining these inequalities, we obtain

$$\mathcal{R}_{p^*}(\widehat{R}) - \mathcal{R}_{p^*}(R^*) \le 2B\,\mathrm{TV}(p^*, W), \tag{33}$$

which proves Eq. equation 27. □

The bound can be further expressed through the Kullback–Leibler divergence. By Pinsker's inequality,

$$\mathrm{TV}(p^*, W) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p^* \| W)}, \tag{34}$$

where

$$D_{\mathrm{KL}}(p^* \| W) = \sum_{j=1}^{n} p_j^* \log \frac{p_j^*}{W_j}. \tag{35}$$

Substituting Eq. equation 34 into Eq. equation 27 gives

$$\mathcal{R}_{p^*}(\widehat{R}) - \mathcal{R}_{p^*}(R^*) \le B\sqrt{2 D_{\mathrm{KL}}(p^* \| W)}. \tag{36}$$

Therefore, the excess semantic risk of the final decision is controlled by the discrepancy between the hallucination-derived reliability distribution and the latent oracle reliability distribution. A more accurate reliability estimate leads to a tighter upper bound on the resulting decision risk.

Table 5: Comparison of uniform, model-level, dataset-level, and instance-level reliability weighting under the same MBR framework. GlobalAcc-MBR uses fixed model weights estimated from overall validation performance, while DatasetAcc-MBR uses fixed dataset-specific model weights. Instance-HS-MBR corresponds to RaTI and estimates candidate reliability separately for each input. Bold indicates the best result for each dataset and metric.

	SQuAD			TYDIQA			NewsQA			Avg. F1
Method	F1	EM	BLEU	F1	EM	BLEU	F1	EM	BLEU	
Uniform-MBR	84.1	72.1	79.3	64.7	50.3	58.0	57.3	39.9	49.7	68.7
GlobalAcc-MBR	84.1	72.0	79.3	64.9	50.4	58.1	57.3	39.7	49.6	68.8
DatasetAcc-MBR	84.1	72.0	79.3	65.1	50.7	58.4	57.4	39.8	49.8	68.9
Instance-HS-MBR (RaTI)	**85.2**	**73.8**	**81.0**	**67.9**	**53.7**	**61.6**	**59.0**	**41.2**	**51.8**	**70.7**

D Additional Experiments and Analysis

D.1 Instance-Level vs. Model-Level Reliability

A central motivation of RaTI is that the overall capability of a model does not necessarily reflect the reliability of its response to a particular question. To examine whether instance-level reliability estimation is necessary, we compare four weighting strategies within the same MBR decoding framework. Specifically, **Uniform-MBR** assigns equal credibility to all candidates, i.e., $W_i = 1/N$. **GlobalAcc-MBR** assigns each model a fixed reliability weight according to its overall validation performance across datasets, while **DatasetAcc-MBR** uses a dataset-specific fixed weight estimated from the corresponding validation set. Finally, **Instance-HS-MBR**, corresponding to RaTI, estimates the reliability of each candidate separately using the hidden-state-based hallucination score for the current context–question pair. The candidate set, semantic encoder, and MBR decision rule are kept unchanged across all variants.

Table 5 reports the results on SQuAD, TYDIQA, and NewsQA. Fixed model-level reliability provides only limited improvements over uniform weighting. In particular, GlobalAcc-MBR and DatasetAcc-MBR achieve very similar performance to Uniform-MBR, suggesting that assigning a model a globally fixed credibility cannot adequately characterize its behavior at the individual-example level. Dataset-specific weighting provides a small advantage over global weighting on TYDIQA and NewsQA, but the improvement remains modest.

In contrast, Instance-HS-MBR consistently achieves the best F1, EM, and BLEU scores on all three datasets. Averaged across the three benchmarks, RaTI obtains 70.7 F1, 56.2 EM, and 64.8 BLEU, compared with 68.9, 54.2, and 62.5 for DatasetAcc-MBR, respectively. This corresponds to improvements of approximately 1.8 F1, 2.0 EM, and 2.3 BLEU points over the strongest fixed-reliability variant.

The difference between GlobalAcc-MBR and DatasetAcc-MBR is relatively small. For example, their F1 scores differ by only 0.03 points on SQuAD, 0.22 points on TYDIQA, and 0.16 points on NewsQA. This indicates that even adapting a fixed model-level weight to the target dataset is insufficient to capture the substantial variation in reliability across individual questions. A model that is generally strong on a dataset may still produce an unreliable response for a particular instance, while a weaker model may provide the correct candidate for that same question.

The advantage of instance-level weighting is more evident on the more challenging datasets. On TYDIQA, RaTI improves F1 from 65.1 for DatasetAcc-MBR to 67.9, together with gains from 50.7 to 53.7 in EM and from 58.4 to 61.6 in BLEU. Similarly, on NewsQA, instance-level reliability increases F1 from 57.4 to 59.0 and BLEU from 49.8 to 51.8. These results suggest that question-dependent reliability becomes particularly important when candidate quality varies substantially across instances.

Overall, the experiment supports the use of candidate-level rather than model-level reliability in RaTI. Global or dataset-specific accuracy can characterize the average capability of a model, but such static estimates cannot determine whether a particular response is trustworthy for the current input. By estimating hallucination risk separately for every candidate, RaTI dynamically redistributes

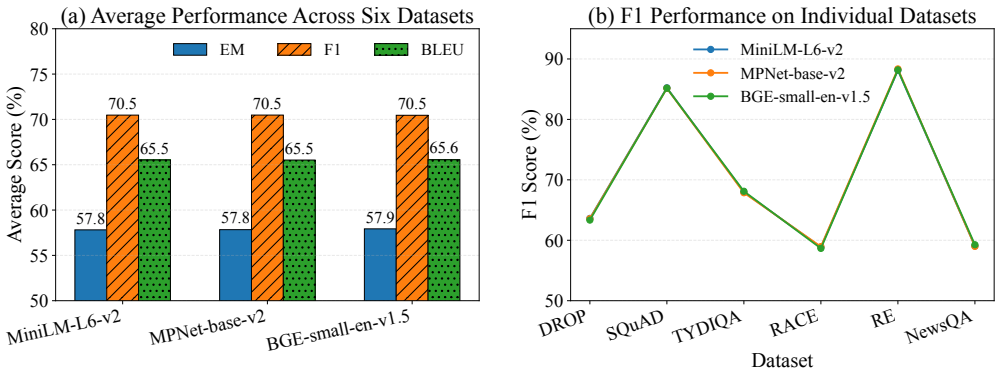


Figure 8: Sensitivity of RaTI to different semantic encoders. (a) Average EM, F1, and BLEU scores across the six QA datasets. (b) Dataset-level F1 performance using MiniLM, MPNet, and BGE-small as the semantic encoder.

credibility within each candidate set and therefore provides a more informative reliability signal for semantic consensus decoding.

D.2 Semantic Encoder Sensitivity

RaTI relies on semantic representations to measure the discrepancy between candidate responses in the weighted MBR objective. To examine whether its performance depends on a particular sentence encoder, we replace the default `all-MiniLM-L6-v2` encoder with two alternative embedding models, `all-mpnet-base-v2` and `bge-small-en-v1.5`, while keeping all other components unchanged.

As shown in Figure 8, RaTI is remarkably insensitive to the choice of semantic encoder. When averaged across the six datasets, all three encoders achieve an F1 score of 70.5. The corresponding EM scores are 57.8, 57.8, and 57.9 for MiniLM, MPNet, and BGE-small, respectively, while the BLEU scores are 65.5, 65.5, and 65.6. Thus, replacing the default MiniLM encoder with substantially different sentence embedding models results in only negligible changes in overall performance.

The dataset-level results in Figure 8(b) exhibit the same pattern. The F1 curves of the three encoders almost completely overlap across DROP, SQuAD, TyDiQA, RACE, Relation Extraction, and NewsQA, indicating that the small aggregate differences are not caused by compensating gains and losses across datasets. Instead, the behavior of RaTI remains highly consistent under different semantic representation models.

These results suggest that the effectiveness of RaTI does not originate from a particular pretrained sentence encoder. Rather, different encoders provide sufficiently consistent semantic relationships among candidate responses for the reliability-aware weighted MBR procedure. This also indicates that RaTI can be combined with different sentence representation models without requiring encoder-specific tuning. We therefore use `all-MiniLM-L6-v2` as the default encoder in the main experiments.

D.3 Weight Mapping Sensitivity

We examine the sensitivity of RaTI to the mapping function that transforms the estimated reliability into aggregation weights. Specifically, we compare four weighting strategies: Uniform, Linear, Exponential, and the Inverse mapping adopted by RaTI. Figure 9 reports both the average performance across the six datasets and the dataset-level F1 scores.

As shown in Figure 9(a), RaTI is relatively robust to the specific choice of the weight mapping function. The differences among Uniform, Linear, and Exponential mappings are generally small, with average F1 scores of 69.6, 70.0, and 69.9, respectively. This indicates that the effectiveness of the proposed aggregation framework does not depend on a narrowly tuned functional form.

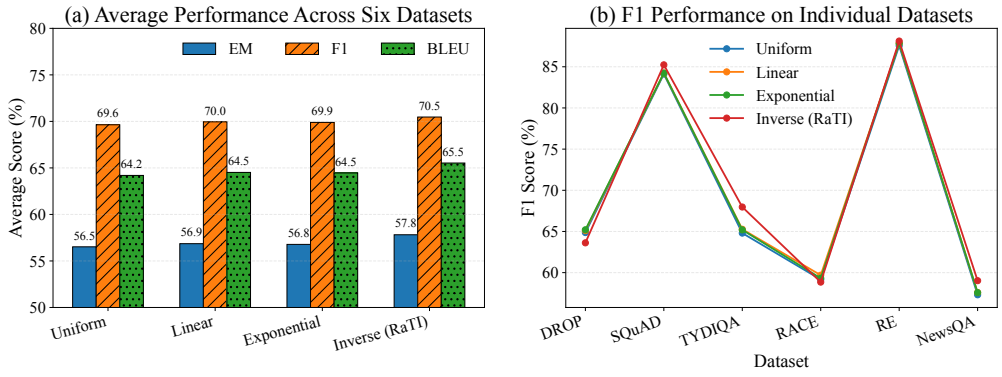


Figure 9: Sensitivity of RaTI to different reliability-to-weight mapping functions. (a) Average EM, F1, and BLEU scores across the six QA datasets. (b) Dataset-level F1 performance under different mapping strategies.

Nevertheless, the Inverse mapping used in RaTI provides the best overall performance across all three evaluation metrics. It achieves an average EM of 57.8, F1 of 70.5, and BLEU of 65.5, compared with 56.9, 70.0, and 64.5 for the strongest alternative configuration in terms of average performance. The improvements are therefore consistent across exact matching, token-level overlap, and BLEU-based evaluation, supporting the use of the Inverse mapping as the default configuration in our main experiments.

Figure 9(b) further shows that the relative behavior of different mapping functions varies across datasets. The Inverse mapping does not strictly dominate every alternative on every individual dataset; for example, other mappings achieve comparable or slightly better performance on some datasets such as DROP and RACE. However, it provides stronger performance on datasets such as TyDiQA and NewsQA while remaining competitive on SQuAD and Relation Extraction. Consequently, the Inverse mapping achieves the best aggregate performance without relying on improvements from a single dataset.

Overall, these results suggest that RaTI is not overly sensitive to the exact form of the reliability-to-weight transformation. At the same time, explicitly incorporating reliability through the Inverse mapping yields the most favorable overall trade-off, motivating its use as the default weighting strategy throughout the main experiments.

D.4 Effect of Ensemble Size

We further investigate how the number of candidate models affects the performance and robustness of RaTI. Given the five candidate models used in our main experiments, we construct ensembles of different sizes, ranging from two to five models. For each ensemble size, we evaluate all possible model combinations and report the best, mean, and worst F1 scores across these combinations. This analysis allows us to examine not only the achievable performance with a favorable model subset, but also the sensitivity of RaTI to the specific composition of the ensemble.

As shown in Figure 10, increasing the number of candidate models consistently improves the overall robustness of RaTI. The best-case performance (left) improves rapidly when increasing the ensemble from two to three or four models, but subsequently tends to saturate. For example, the best F1 on DROP increases from 62.3 with two models to 64.0 with four models, while using all five models yields a comparable score of 63.6. A similar pattern can be observed on Relation Extraction and SQuAD, where carefully selected smaller subsets can already achieve performance close to, or slightly above, that of the full ensemble. This suggests that simply increasing the number of models does not necessarily improve the best attainable performance, since a favorable subset may already contain sufficient complementary evidence.

More importantly, the mean performance (middle) exhibits a much more consistent upward trend as the ensemble size increases. Across all datasets, larger ensembles achieve higher average F1 scores over different model combinations. For instance, the mean F1 increases from 59.0 to 63.6 on DROP,

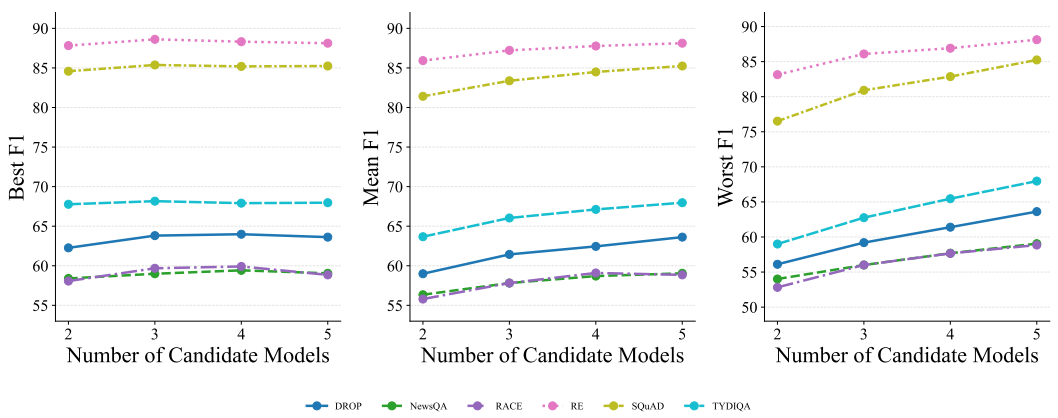


Figure 10: Effect of the number of candidate models on F1 performance across six QA datasets. From left to right, the three panels report the best, mean, and worst F1 scores among all model combinations of a given ensemble size, respectively.

from 81.4 to 85.2 on SQuAD, and from 63.7 to 68.0 on TyDiQA when the number of candidate models increases from two to five. These results indicate that incorporating additional models reduces RaTI's dependence on selecting a particular favorable subset and makes its performance more stable across different ensemble compositions.

The improvement is even more pronounced in the worst-case setting (right). The worst F1 increases from 56.1 to 63.6 on DROP, from 76.5 to 85.2 on SQuAD, and from 59.0 to 68.0 on TyDiQA. Similar improvements are observed on the remaining datasets. Consequently, the gap between the best and worst model combinations becomes substantially smaller as more candidate models are included. For example, on TyDiQA, the best–worst gap is 8.8 F1 points with two models, whereas it is reduced to approximately 2.4 points with four models. This demonstrates that larger different ensembles substantially mitigate the risk of selecting an unfavorable combination of models.

Overall, these results show that the primary benefit of increasing ensemble size is not necessarily a higher best-case score, but improved *stability* and *robustness* across different model compositions. With only a small number of candidate models, performance can vary considerably depending on which models are included. As the ensemble becomes larger, RaTI has access to more diverse evidence and becomes less sensitive to individual model selection, leading to steadily improved mean and worst-case performance.

D.5 Effect of Ensemble Heterogeneity

We further investigate whether the effectiveness of RaTI primarily arises from aggregating multiple samples or from combining different model families. Unlike the main experiments, which use deterministic decoding with temperature $T = 0$, this analysis adopts stochastic decoding with temperature $T = 0.5$ for all candidate generators. This setting is necessary to construct meaningful homogeneous ensembles containing multiple independently sampled responses from the same model. We compare a heterogeneous candidate pool containing one sampled response from each of the five lightweight models against homogeneous ensembles constructed by independently sampling five responses from the same model. All candidate sets are processed using the same reliability-aware weighted MBR decision procedure.

As shown in Table 6, the different ensemble consistently outperforms all homogeneous multi-sampling variants across the three answer-quality metrics. Different-5 achieves an F1 score of 69.94, compared with 65.62 for the strongest homogeneous configuration, Phi $\times 5$, yielding an improvement of 4.32 F1 points. Similar gains are observed for EM and BLEU. The best homogeneous EM score is obtained by Llama $\times 5$ at 50.76, whereas Different-5 reaches 56.06, corresponding to a 5.30-point improvement. For BLEU, Different-5 improves over the strongest homogeneous configuration from 61.34 to 65.08, a gain of 3.74 points.

Table 6: Effect of ensemble heterogeneity. Homogeneous variants contain five independently sampled responses from the same model, whereas Different-5 contains one response from each of the five different lightweight models. Diversity denotes the average pairwise semantic distance among candidate responses. Higher values are better for all metrics.

Method	F1	EM	BLEU	Diversity
Gemma $\times 5$	59.61	43.14	51.60	0.0182
Llama $\times 5$	63.47	50.76	57.72	0.0196
Phi $\times 5$	65.62	47.64	61.34	0.0317
Ministral $\times 5$	62.09	48.83	56.71	0.0500
Qwen $\times 5$	59.62	47.55	54.43	0.0122
Different-5	**69.94**	**56.06**	**65.08**	**0.2552**

The difference is particularly pronounced in candidate diversity. Repeated sampling from the same model produces relatively concentrated candidate sets, with semantic diversity ranging only from 0.0122 to 0.0500. In contrast, Different-5 achieves a diversity score of 0.2552, which is approximately $5.1\times$ higher than even the most diverse homogeneous ensemble and $9.7\times$ higher than the average diversity of the homogeneous configurations. This result indicates that repeated sampling from a single lightweight model tends to explore a relatively narrow semantic region, whereas combining different model families produces substantially more diverse hypotheses.

Importantly, the increased diversity does not come at the expense of answer quality. Instead, the different ensemble simultaneously achieves the highest diversity and the strongest F1, EM, and BLEU scores. Averaged over the five homogeneous configurations, different sampling improves F1 by 7.86 points, EM by 8.47 points, and BLEU by 8.72 points. These results suggest that the benefit of RaTI cannot be attributed merely to obtaining multiple candidate responses. Rather, different models provide complementary hypotheses and distinct error patterns that enlarge the useful candidate space available to the subsequent reliability-aware consensus stage.

This observation is consistent with the design motivation of Multi-Model Collaborative Sampling: responses generated by models from different architectural and pre-training families are less likely to exhibit identical failure patterns than repeated samples from a single model. The reliability-weighted MBR module can then exploit this different candidate space by suppressing unreliable alternatives while preserving semantically complementary evidence. Consequently, ensemble heterogeneity provides a substantially stronger basis for collaborative truth inference than homogeneous self-sampling alone.

D.6 Statistical Significance and Confidence Intervals

To assess whether the improvements of RaTI are robust to sampling variation rather than being artifacts of a particular test set, we conduct paired bootstrap significance analysis over the six QA benchmarks. For each dataset, we resample the aligned question–answer instances with replacement for 10,000 bootstrap iterations. Since RaTI and each baseline are evaluated on exactly the same examples, the same resampled indices are used for both methods in each iteration. This paired procedure preserves the instance-level correspondence between predictions and provides a direct estimate of the uncertainty of the performance difference.

For a baseline method B, we define the bootstrap performance difference as

$$\Delta^{(b)} = M^{(b)}_{\mathrm{RaTI}} - M^{(b)}_{B}, \tag{37}$$

where $M^{(b)}$ denotes the evaluation metric computed on the b-th bootstrap sample. We report the observed performance difference together with the 95% percentile bootstrap confidence interval. A difference is regarded as statistically significant when its confidence interval does not include zero.

Table 7 reports the paired bootstrap results for F1, which is the primary metric used in our overall comparison. Each entry represents the F1 improvement of RaTI over the corresponding collaborative baseline, with the 95% confidence interval shown in brackets. RaTI obtains a positive F1 difference

Table 7: Paired bootstrap analysis of F1 improvements over collaborative baselines. Each entry reports ΔF1 = $\mathrm{F1}_{\mathrm{RaTI}} - \mathrm{F1}_{\mathrm{baseline}}$, followed by the 95% bootstrap confidence interval in brackets. Bold values indicate that the confidence interval excludes zero. Bootstrap confidence intervals are estimated from 10,000 paired resamples.

Dataset	RaTI vs. LLM-Debate	RaTI vs. MoA	RaTI vs. SLM-MUX
SQuAD	$\mathbf{+4.96\ [4.39,\ 5.51]}$	$\mathbf{+12.20\ [11.91,\ 12.50]}$	$\mathbf{+4.65\ [4.06,\ 5.24]}$
Relation Extraction	$\mathbf{+6.24\ [5.34,\ 7.16]}$	$\mathbf{+12.61\ [11.37,\ 13.90]}$	$\mathbf{+3.45\ [2.51,\ 4.43]}$
TYDIQA	$\mathbf{+7.83\ [6.89,\ 8.76]}$	$\mathbf{+12.84\ [11.77,\ 13.95]}$	$\mathbf{+2.67\ [1.71,\ 3.63]}$
NewsQA	$\mathbf{+4.27\ [3.39,\ 5.13]}$	$\mathbf{+6.71\ [5.74,\ 7.69]}$	$\mathbf{+2.96\ [1.97,\ 3.90]}$
RACE	$+2.32\ [-0.34,\ 5.05]$	$\mathbf{+10.36\ [7.33,\ 13.51]}$	$\mathbf{+5.90\ [3.08,\ 8.74]}$
DROP	$+1.89\ [-0.11,\ 3.87]$	$\mathbf{+10.56\ [8.41,\ 12.74]}$	$\mathbf{+9.02\ [6.89,\ 11.22]}$

Table 8: Summary of paired bootstrap significance results across the six datasets and three collaborative baselines.

Metric	Significant Comparisons	Total Comparisons
F1	16	18
EM	17	18
BLEU	18	18
Overall	51	54

in all 18 dataset–baseline comparisons. Among them, 16 comparisons have confidence intervals entirely above zero, demonstrating that the observed improvements are statistically robust in most evaluation settings.

The significance pattern is particularly consistent for MoA and SLM-MUX. RaTI significantly outperforms both methods on all six datasets, with F1 improvements ranging from $+6.71$ to $+12.84$ points over MoA and from $+2.67$ to $+9.02$ points over SLM-MUX. These results indicate that the advantages observed in the main experiments are robust across evaluation samples rather than being driven by a small subset of instances.

Compared with LLM-Debate, RaTI also achieves higher mean F1 on every dataset. The improvements are statistically significant on SQuAD, Relation Extraction, TYDIQA, and NewsQA. On RACE and DROP, RaTI improves F1 by $+2.32$ and $+1.89$ points, respectively, but the corresponding 95% confidence intervals slightly cross zero, namely $[-0.34, 5.05]$ on RACE and $[-0.11, 3.87]$ on DROP. Therefore, although RaTI achieves higher point estimates on these two benchmarks, the F1 differences relative to LLM-Debate are not statistically significant at the 95% confidence level. The wider confidence interval on RACE is also consistent with its smaller evaluation set.

The conclusions remain consistent across the complementary evaluation metrics. As summarized in Table 8, 17 of the 18 Exact Match comparisons and all 18 BLEU comparisons yield 95% confidence intervals excluding zero. The only additional non-significant comparison is Exact Match between RaTI and LLM-Debate on RACE. Overall, 51 of the 54 dataset–baseline–metric comparisons are statistically significant under the paired bootstrap criterion.

Taken together, the bootstrap analysis provides additional evidence that RaTI's performance gains are generally stable across evaluation samples. In particular, the consistent significance over MoA and SLM-MUX across all six datasets suggests that reliability-aware semantic consensus provides a robust advantage over these collaborative inference strategies. At the same time, the non-significant F1 differences against LLM-Debate on RACE and DROP indicate that the magnitude of RaTI's advantage can depend on the dataset and evaluation sample size, providing a more conservative characterization of its empirical improvements.

D.7 Generalization to Mathematical Reasoning

The main experiments primarily evaluate RaTI on general question answering benchmarks. To further examine whether the proposed reliability-aware aggregation framework generalizes beyond

Table 9: Accuracy (%) on GSM8K. The best result is shown in bold and the second-best result is underlined.

Method	Accuracy (%)
LLM-Debate	70.49
MoA	68.69
SLM-MUX	<u>70.60</u>
RaTI	**71.27**

conventional QA tasks, we evaluate RaTI on GSM8K, a benchmark requiring multi-step mathematical reasoning. Unlike extractive or knowledge-intensive QA, GSM8K requires models to perform a sequence of intermediate reasoning steps before producing the final numeric answer, providing a substantially different setting for evaluating answer aggregation.

We compare RaTI-MBR with three representative multi-model inference approaches: LLM-Debate, Mixture-of-Agents (MoA), and SLM-MUX. All methods are evaluated using answer accuracy on the same GSM8K evaluation set. The results are reported in Table 9.

As shown in Table 9, RaTI-MBR achieves the highest accuracy of 71.27% on GSM8K, outperforming all competing methods. In particular, it improves over SLM-MUX, the strongest baseline, by approximately 0.67 percentage points, while exceeding LLM-Debate and MoA by 0.78 and 2.58 percentage points, respectively.

Although the margin over the strongest baseline is moderate, the result is notable because GSM8K differs substantially from the QA benchmarks used in our main experiments. Mathematical reasoning often produces highly diverse candidate responses: different models may follow different reasoning paths, make errors at different intermediate steps, or arrive at the same final answer through distinct derivations. Under such conditions, relying solely on response frequency or uniform aggregation may fail to distinguish reliable evidence from erroneous reasoning trajectories. RaTI-MBR instead incorporates instance-level reliability into the aggregation process, allowing the contribution of each candidate response to be adjusted according to its estimated reliability.

The consistent advantage of RaTI-MBR over LLM-Debate, MoA, and SLM-MUX suggests that the proposed reliability-aware inference mechanism is not restricted to conventional QA tasks. Rather, it remains effective in a multi-step mathematical reasoning setting, providing additional evidence for the task-level generalizability of RaTI.

D.8 Comparison with a Larger Single Model

To examine whether the gains obtained by RaTI could be reproduced simply by increasing the capacity of a single model, we additionally compare RaTI with Granite3.3:8B, which serves as a larger single-model reference. In contrast to the 8B model, RaTI constructs its candidate pool entirely from lightweight models ranging from 2B to 3.8B parameters and performs final prediction through reliability-aware consensus rather than relying on a larger autoregressive model.

As shown in Table 10, RaTI achieves performance that is competitive with Granite3.3:8B across the six QA benchmarks. Averaged over all datasets, RaTI obtains an EM score of 57.82, compared with 56.61 for Granite3.3:8B. In terms of F1 and BLEU, Granite3.3:8B achieves averages of 71.99 and 67.59, respectively, while RaTI reaches 70.48 and 65.53, corresponding to relatively small gaps of 1.51 F1 points and 2.06 BLEU points.

The comparison also reveals substantial dataset-dependent differences. RaTI achieves higher EM than Granite3.3:8B on four of the six benchmarks, including TYDIQA, RACE, Relation Extraction, and NewsQA. The advantage is particularly evident on Relation Extraction, where RaTI improves over the 8B model by 5.46 EM points, 1.12 F1 points, and 1.81 BLEU points. On SQuAD and DROP, the larger model performs better, although the EM differences remain within 1.0 point. These results suggest that increasing single-model capacity does not uniformly dominate different collaboration across different QA tasks.

Table 10: Comparison between RaTI and a larger single-model reference, Granite3.3:8B, across six QA benchmarks. Bold numbers indicate the better result between the two methods for each metric. Although RaTI relies only on lightweight candidate generators of 2–3.8B parameters, it achieves performance comparable to the substantially larger 8B model.

Dataset	**RaTI**			**Granite3.3:8B**		
	EM	F1	BLEU	EM	F1	BLEU
SQuAD	73.79	85.24	81.01	**74.58**	**87.15**	**84.09**
TYDIQA	**53.77**	67.97	61.61	52.59	**69.84**	**63.92**
RACE	**43.92**	58.85	53.27	43.18	**59.91**	**55.55**
Relation Extraction	**79.61**	**88.12**	**85.13**	74.15	87.00	83.32
NewsQA	**41.24**	59.05	51.85	39.60	**61.16**	**54.88**
DROP	54.56	63.62	60.29	**55.56**	**66.87**	**63.76**
Average	**57.82**	70.48	65.53	56.61	**71.99**	**67.59**

Table 11: Candidate coverage and oracle analysis across six QA benchmarks. Candidate Oracle EM measures whether at least one of the five candidate responses exactly matches the ground-truth answer. The oracle therefore reflects the coverage of the fixed candidate pool rather than an executable inference method. Selection Success denotes the proportion of oracle-solvable instances that are correctly selected by RaTI.

Dataset	Best Single EM	RaTI EM	Candidate Oracle EM	Gap ↓	Selection Success (%) ↑
DROP	49.70	54.56	76.31	21.76	71.49
SQuAD	67.64	73.79	86.11	12.32	85.70
TYDIQA	48.16	53.77	64.75	10.98	83.05
RACE	39.47	43.92	63.06	19.14	69.65
Relation Extraction	74.66	79.61	88.81	9.19	89.65
NewsQA	37.80	41.24	52.28	11.04	78.88
Average	52.91	**57.82**	71.89	14.07	79.74

Overall, the results indicate that reliability-aware collaboration among different lightweight models can recover a substantial portion of the performance associated with increasing single-model capacity. Importantly, this comparison should not be interpreted as implying that RaTI necessarily requires less total computation than an 8B model, since RaTI evaluates multiple candidate generators and an additional reliability estimator. Rather, the result demonstrates that strong QA performance can be obtained without requiring an 8B candidate generator or a large autoregressive coordinator, which is particularly relevant to deployment settings where individual model size and peak memory footprint are constrained.

D.9 Candidate Coverage and Selection Headroom

The performance of a candidate-selection framework is jointly determined by two factors: whether the candidate pool contains a correct answer and whether the inference mechanism can successfully identify that answer. To disentangle these two sources of error, we conduct a candidate coverage analysis based on an oracle selection criterion.

For each question, the Candidate Oracle is considered correct if at least one of the five candidate responses exactly matches the ground-truth answer. Formally, for a question q with candidate set $\mathcal{R}_q = \{R_1, \ldots, R_n\}$ and reference answer y_q, we define

$$\mathrm{OracleEM}(q) = \max_{R_i \in \mathcal{R}_q} \mathrm{EM}(R_i, y_q). \tag{38}$$

The resulting Candidate Oracle EM therefore measures the coverage of the fixed candidate pool rather than the performance of an executable inference method. We further define the conditional

selection success rate as

$$\text{SelectionSuccess} = \frac{\text{EM}_{\text{RaTI}}}{\text{EM}_{\text{Oracle}}}, \tag{39}$$

which measures the proportion of oracle-solvable instances that are successfully resolved by RaTI.

As reported in Table 11, the five-model candidate pool achieves an average Candidate Oracle EM of 71.89, substantially higher than the 52.91 average EM of the strongest individual lightweight models and the 57.82 EM obtained by RaTI. This confirms that different candidate generation provides considerable complementary coverage: in many cases where an individual lightweight model fails, at least one of the remaining models still produces an exact-match answer.

At the same time, the 14.07-point difference between RaTI and the Candidate Oracle reveals remaining headroom in the final selection stage. Conditional on the presence of an exact-match candidate, RaTI successfully selects the correct answer in 79.74% of cases on average across the six datasets. The highest selection success is observed on Relation Extraction (89.65%) and SQuAD (85.70%), whereas RACE (69.65%) and DROP (71.49%) remain more challenging. These results indicate that the current reliability-weighted consensus mechanism captures most, but not all, of the useful information available in the candidate pool.

The candidate coverage also varies substantially across datasets. Candidate Oracle EM reaches 88.81 on Relation Extraction and 86.11 on SQuAD, showing that the different model pool provides strong coverage on these tasks. In contrast, the oracle score is only 52.28 on NewsQA and 64.75 on TYDIQA, indicating that a considerable fraction of questions cannot be solved by any of the five candidate models. In such cases, improving the downstream consensus mechanism alone cannot recover the correct answer because no exact-match candidate is available.

These observations are consistent with the error decomposition in the main text, where candidate failure constitutes the largest category of remaining RaTI errors across all six datasets. At the same time, the non-negligible oracle gap shows that candidate selection also remains an important source of improvement, particularly on DROP, RACE, and SQuAD. Therefore, further improvements to RaTI may arise from two complementary directions: increasing the coverage and diversity of the candidate pool, and improving the calibration and discrimination of instance-level reliability estimation.

D.10 Qualitative Case Analysis

To better understand how RaTI behaves under different forms of model disagreement, we conduct a qualitative analysis of representative multi-model QA instances. Rather than treating surface-form differences as distinct decisions, we manually interpret semantically equivalent responses as belonging to the same answer cluster. The selected examples therefore focus on genuine factual conflicts, where competing candidates refer to different entities, quantities, or conclusions. The cases illustrate three desirable behaviors of RaTI—resistance to an incorrect semantic majority, reliability-aware resolution of factual conflicts, and candidate-level risk calibration—as well as a representative failure mode.

Case 1: Resisting an Incorrect Semantic Majority. Table 12 contains two competing semantic clusters. Llama and Ministral predict "M. Vick," while Qwen predicts "Vick"; these responses refer to the same entity and therefore form an incorrect three-model cluster. In contrast, only Phi and Gemma support the correct answer, "D. Brees." Despite this 3:2 semantic majority, RaTI selects "D. Brees." This demonstrates that its decision is not determined purely by response frequency: candidate-level reliability changes the effective contribution of each prediction before semantic consensus is computed. The contrast with RaTI-CRH, which follows the incorrect Vick cluster, further suggests that the way reliability is incorporated into aggregation is important for recovering the correct minority answer.

Case 2: Recovering the Correct Answer from an Incorrect Majority. Table 13 provides another case where inter-model agreement does not imply factual correctness. Clayton records $92/6 \approx 15.3$ yards per reception, while Nathan records $83/10 = 8.3$, making *Clayton* the unambiguous answer. Nevertheless, Llama, Qwen, and Ministral predict *Nathan*, producing an incorrect 3:2 majority and causing Majority Vote to fail. RaTI instead selects *Clayton*. Because the disagreement concerns two different players rather than lexical variants, the example shows that raw response frequency alone is

Table 12: RaTI overrides an incorrect semantic majority.

Context: After a tough loss at home to the Lions, . . .

Question: Which quarterback had 2 touchdown passes?

Ground Truth: D. Brees

Model / Method	Prediction	Correct
Llama3.2-3B	M. Vick	×
Phi3-3.8B	D. Brees	✓
Ministral3-3B	M. Vick	×
Gemma2-2B	D. Brees	✓
Qwen2.5-3B	Vick	×
RaTI	**D. Brees**	✓
RaTI-CRH	M. Vick	×
RaTI-MV	D. Brees	✓
RaTI-DS	D. Brees	✓

Table 13: RaTI corrects an incorrect majority using instance-level reliability.

Context: Marino finished with 318 passing yards. Clayton recorded 6 receptions for 92 yards, while Nathan caught 10 passes for 83 yards. . . .

Question: Who had a higher average of yards per caught pass, Clayton or Nathan?

Ground Truth: Clayton $(92/6 \approx 15.3 > 83/10 = 8.3)$

Model / Method	Prediction	Correct
Llama3.2-3B	Nathan	×
Qwen2.5-3B	Nathan	×
Gemma2-2B	Clayton	✓
Ministral3-3B	Nathan	×
Phi3-3.8B	Clayton	✓
Majority Vote	Nathan	×
RaTI	**Clayton**	✓

insufficient. Instance-level reliability allows RaTI to reduce the influence of repeated but unreliable predictions and recover the evidence-supported minority answer.

Case 3: Selecting a Reliable Singleton against Two Incorrect Consensus Clusters. Table 14 illustrates a more difficult setting in which the correct answer is supported by only one model. The context contains three salient scores: 31–0 at halftime, 31–24 as the final score of the divisional-round game, and 49–15 in the subsequent NFC Championship. Since the question asks for the game played prior to the NFC Championship, the correct answer is 31–24. Qwen and Gemma incorrectly select 49–15, while Phi and Llama select the halftime score 31–0. Only Ministral predicts 31–24. Thus, the correct candidate has the lowest raw frequency. RaTI overcomes this disadvantage through reliability weighting. The correct response has a hallucination probability of only 0.09 and consequently receives a weight of 0.66, substantially larger than the weights assigned to the four incorrect candidates. It also achieves the lowest weighted MBR risk (0.06). Therefore, RaTI selects 31–24 despite its singleton status. This case directly illustrates the interaction between RaTI's two components: reliability estimation suppresses unreliable repeated answers, while weighted semantic MBR determines the final candidate according to the resulting credibility distribution.

Case 4: Failure under Correlated Semantic Errors. Table 15 presents a representative failure mode. The context explicitly states that males have a median income of $27,346 compared with $21,145 for females, so the correct answer is *males*. However, Gemma, Llama, and Qwen all predict *Females/females*, forming a correlated incorrect semantic majority. The candidate set still contains useful evidence: Ministral directly predicts *males*, while Phi outputs $27,346, the correct supporting value, although it does not match the required entity answer under exact-match evaluation. RaTI

Table 14: RaTI selects a reliable singleton candidate against two incorrect semantic clusters.

Context: The Panthers defeated the Seattle Seahawks in the divisional round after taking a 31–0 halftime lead and holding off a second-half comeback to win 31–24. . . . They then defeated the Arizona Cardinals 49–15 in the NFC Championship Game.

Question: What was the score of the last game the Carolina Panthers played prior to the NFC Championship?

Ground Truth: 31–24

Candidate	Prediction	$P(\text{Hall.})$	Weight	MBR Risk	Correct
Qwen2.5-3B	49–15	0.80	0.07	0.22	×
Phi3-3.8B	31–0	0.60	0.10	0.14	×
Ministral3-3B	**31–24**	**0.09**	**0.66**	**0.06**	✓
Llama3.2-3B	31–0	0.60	0.10	0.14	×
Gemma2-2B	49–15	0.80	0.07	0.22	×
RaTI	**31–24**	**Selected candidate with minimum weighted risk**			✓

Table 15: Case 4: Failure of RaTI under a correlated incorrect semantic majority.

Context: The median income for a household in the county was $32,073, and the median income for a family was $36,999. Males had a median income of $27,346 versus $21,145 for females. . . .

Question: Whose median income was higher, males or females?

Ground Truth: males ($27,346 > $21,145)

Model / Method	Prediction	Correct
Gemma2-2B	Females	×
Llama3.2-3B	females	×
Phi3-3.8B	$27,346	△
Ministral3-3B	males	✓
Qwen2.5-3B	Females	×
RaTI	**Females**	×
LLM-Debate	Females	×
MoA	females	×
SLM-MUX	males	✓

Note: △ indicates a response that extracts the correct supporting value ($27,346 is the male median income) but does not directly answer the entity-selection question and is therefore not counted as an exact correct answer.

nevertheless selects *Females*. This example shows that reliability-weighted consensus can still fail when several models produce the same factual error and the reliability estimator does not suppress their shared cluster sufficiently. LLM-Debate and MoA make the same error, whereas SLM-MUX selects the correct answer, indicating that this is an instance-specific aggregation failure rather than a universal failure of all collaborative methods. Importantly, this is a consensus/scoring failure rather than a candidate-generation failure: the correct answer is already present in the candidate set, but receives insufficient influence. Such cases motivate future extensions that explicitly model correlations among model errors or perform cluster-level reliability calibration.

Overall Discussion. The four cases demonstrate that multi-model QA cannot be reduced to counting repeated answers. Cases 1 and 2 show that an incorrect semantic majority can be overridden when minority candidates are more reliable, while Case 3 demonstrates that reliability weighting can even recover a correct singleton against two repeated error clusters. Case 4 highlights the remaining limitation: correlated errors can still dominate when their reliability is overestimated. Overall, RaTI should therefore be viewed as a reliability-aware semantic decision mechanism that jointly considers candidate credibility and inter-candidate agreement.