CAST: An Interpretable Evidence-Based Confidence Estimation Tool for RAG with Human-in-the-Loop Verification

Xutong Liu[1], **Morris Groot Beumer**[2], **Jörg Osterrieder**[1], **Max Baak**[2]
[1]University of Twente, Enschede, the Netherlands
[2]ING Bank, Amsterdam, the Netherlands
xutong.liu@utwente.nl, morris.groot.beumer@ing.com,
joerg.osterrieder@utwente.nl, max.baak@ing.com

Abstract

Large language models (LLMs) increasingly assist financial analysis tasks over filings, reports, and news via retrieval-augmented generation (RAG), but a human reviewer typically remains accountable for the final decision. In this setting, a confidence score must be more than well-calibrated: it must be evidence-based and explainable, so reviewers can see why an answer should be trusted. We introduce CAST (Confidence via Answer Support Testing), a Bayesian-inspired confidence method that compares a generated answer with a plausible competing alternative and evaluates how retrieved passages support each hypothesis. CAST produces both an answer-level confidence score and citation-level evidence for review. We also demonstrate CAST through an interactive reviewer-facing tool that visualizes supporting and opposing evidence for human verification. On 1,286 financial-report questions from MMDocRAG, CAST achieves the highest AUROC among seven evaluated methods (0.634 vs. 0.598 for the strongest baseline).

1 Introduction

LLMs are increasingly used to assist the financial industry in question-answering tasks over filings, reports, earnings materials, and news using retrieval-augmented generation (RAG) (Li et al., 2023). In many deployments the model does not act autonomously, but an expert reviews the answer before it is used and remains accountable for the decision it informs upon (Joshi, 2025). This human-in-the-loop (HITL) verification not only requires the generated answer to the question, but also the evidence supporting the answer, in the form of citations. A well-calibrated, evidence-based LLM confidence score is particularly helpful to this review. For example, as illustrated in Fig. 1, at below-threshold confidence the reviewer needs to judge whether that evidence actually supports the claim; in general this is a time-consuming, manual process.

This makes confidence estimation for HITL verification a distinct problem from confidence estimation for open-ended generation: the confidence score must not only be well-calibrated, but evidence-based and explainable, so a reviewer can see why an answer should or should not be trusted.

Existing confidence estimation methods fall short of this requirement. Token-level logits and semantic entropy reflect the model's internal certainty but do not (directly) consult the retrieved evidence (Kuhn et al., 2024; Geng et al., 2024). Sampling-based methods (self-consistency, semantic entropy) require multiple additional generations and still produce only a bare number (Manakul et al., 2023a; Farquhar et al., 2024). Verbalized confidence is known to be poorly calibrated and overconfident (Tian et al., 2023). However, conventional confidence-estimation methods generally do not expose how individual retrieved passages contribute to the confidence score, limiting their usefulness for evidence-level human review.

This work presents CAST (Confidence via Answer Support Testing), a novel method which closes this gap by reusing evidence the RAG pipeline has already retrieved. CAST requires only one extra processing step after a regular RAG pipeline and works with any black-box language model. Its confidence score follows a Bayesian-inspired formulation that compares the relative evidence for the generated answer and a plausible competing alternative. An LLM judge evaluates how well each retrieved passage supports both hypotheses, and the resulting citation-level judgments are aggregated into an answer-level confidence score. Because scoring happens passage-by-passage, CAST produces not just a number but a quantitative, citation-level rationale showing which evidence supports the generated answer and which favors the alternative. Moreover, the strength of

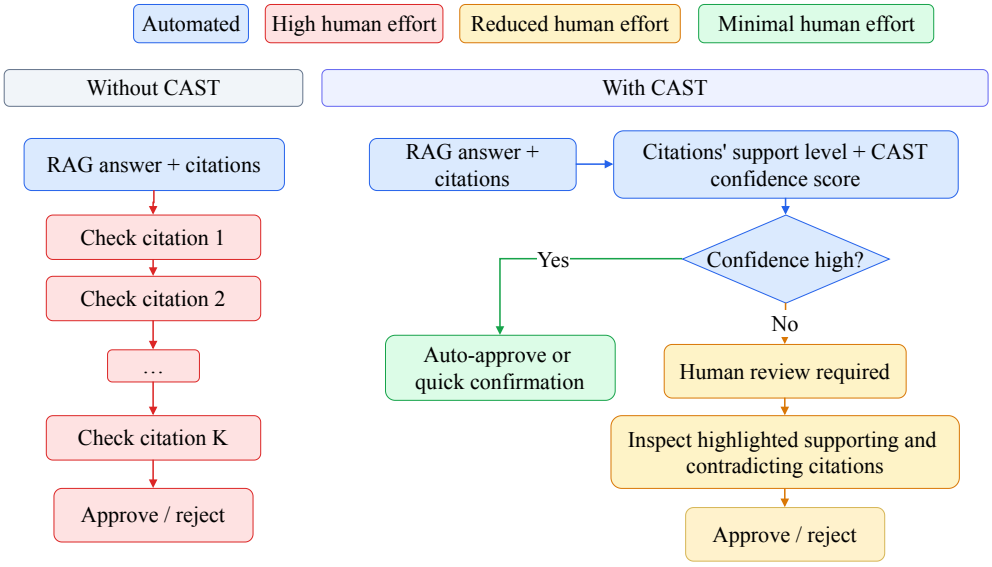

Figure 1: Integration of CAST confidence scores in a human-in-the-loop financial question-answering workflow.

each piece of evidence can be easily visualized to support human review.

Our contributions are as follows. We present CAST, a practical evidence-based confidence scoring method that compares an LLM-generated answer with a plausible competing alternative using the evidence already retrieved by a RAG system. We develop an interactive reviewer-facing tool that exposes CAST's citation-level support judgments, allowing reviewers to inspect which evidence supports the generated answer and which favors the alternative. We evaluate CAST on 1,286 financial-report questions from MMDocRAG (Dong et al., 2026) against six confidence-estimation baselines, demonstrating its potential for human-in-the-loop financial RAG. The replication kit, including scripts, data, and graphs, is publicly available (Anonymous, 2026).

2 Related Work

2.1 Confidence Estimation for LLMs

Confidence estimation methods are commonly divided into white-box approaches, which use model internals such as token probabilities or hidden states, and black-box approaches, which rely only on observable outputs (Liu et al., 2025). Representative white-box methods include the original formulations of P(True) and P(IK) (Kadavath et al., 2022) and semantic entropy (Kuhn et al., 2024), while representative black-box methods include SelfCheckGPT (Manakul et al., 2023b) and verbalized confidence (Tian et al., 2023). Since financial RAG systems are often built on closed, API-served LLMs without access to internal states, we implement Verbalized P(True), Verbalized P(IKnow), and semantic entropy as black-box methods, querying the model directly rather than using log-probabilities, and retain logit confidence as our sole white-box baseline.

2.2 Evidence-Based Confidence in RAG

Recent work has begun grounding confidence in retrieved evidence rather than model uncertainty alone. RAGTruth (Niu et al., 2024) shows that hallucinations persist even in RAG systems, motivating explicit evidence verification. Attribution methods such as RARR (Gao et al., 2023) verify whether retrieved passages support generated claims, but focus on claim verification rather than confidence estimation. The work most closely related to ours is that of (Becker and Soatto, 2024), which estimates confidence from the consistency of model-generated explanations using textual entailment. While both methods employ entailment signals, their approach evaluates the internal consistency of generated explanations, whereas CAST grounds confidence in retrieved evidence. Instead of measuring support for a single hypothesis, CAST compares the generated answer against a constructed alternative hypothesis and aggregates supporting and contradicting evidence into a Bayesian confidence score. The resulting citation-level explanations provide reviewers with a transparent rationale for the confidence estimate.

3 CAST methodology

The CAST method is motivated by Bayesian evidence aggregation. Bayes' rule provides a principled framework for expressing how retrieved evidence can update the relative plausibility of the generated answer and a competing alternative. The practical CAST score introduced below follows this log-odds intuition, while using support scores produced by an LLM judge as evidence signals rather than explicitly estimated likelihood ratios.

3.1 Bayesian design

Consider the input question, an LLM-generated answer, and the retrieved set of citations relevant to answering the question ($\mathcal{E}$). CAST tests two competing hypotheses:

- H: the LLM-generated answer to the question is correct;
- H_A: a constructed alternative answer to the question is correct.

Here, H_A represents one plausible competing hypothesis rather than the full complement of H; the two coincide only for binary questions.

Using Bayes' formula and assuming all citations are conditionally independent, each citation e_i updates the prior ratio with its own likelihood ratio:

$$\begin{aligned}\frac{P(H|\mathcal{E})}{P(H_A|\mathcal{E})} &= \frac{P(\mathcal{E}|H)}{P(\mathcal{E}|H_A)}\frac{P(H)}{P(H_A)} \\ &= \left\{\prod_i \frac{P(e_i|H)}{P(e_i|H_A)}\right\}\frac{P(H)}{P(H_A)}. \quad (1)\end{aligned}$$

We take equal prior odds, $P(H)/P(H_A) = 1$. Since the likelihood terms are not directly available, CAST uses LLM-assigned support scores as evidence signals in the Bayesian-inspired aggregation described in Section 3.3.

CAST constructs the alternative according to the detected answer type: binary answers are reversed, numeric answers are perturbed by 10–40%, and qualitative answers are rewritten using a different entity or conclusion at a comparable level of detail. Answer types are identified through pattern-matching rules. Gemini 2.5 Flash is then used only to instantiate the corresponding alternative-answer template (Appendix A.3).

Each retrieved citation is then evaluated against H and H_A to obtain relative support scores (Section 3.2), which are aggregated into the final CAST confidence score (Section 3.3).

3.2 Citation-Level Evidence Testing

For each citation e_i, an LLM judge assigns a support score to the generated-answer hypothesis and the constructed alternative hypothesis:

$$s_i^H = s(e_i, H), \qquad s_i^{H_A} = s(e_i, H_A). \quad (2)$$

The judge uses the rubric in Appendix B (Table 2); it performs textual classification into 13 numerical support levels, with scores $[0, \ldots, 12]$, indicating how strongly the citation supports the claims made by each hypothesis. The two scores are assigned independently, allowing a citation to support both hypotheses to different degrees.

CAST summarizes their relative support using the normalized evidence gap:

$$\Delta_i = \frac{s_i^H - s_i^{H_A}}{12}, \quad (3)$$

where $\Delta_i > 0$ favors the generated answer, $\Delta_i < 0$ favors the constructed alternative, and values near zero indicate balanced or inconclusive evidence.

3.3 Confidence Aggregation

The implementation additionally labels citations as directly or weakly relevant. Directly relevant citations are used to compute the principal evidence gap. Let $\mathcal{E}_d$ denote the subset of citations marked as directly relevant. If no citation is marked as directly relevant then $\mathcal{E}_d = \mathcal{E}$.

CAST first calculates the mean normalized evidence gap:

$$\bar{\Delta}_d = \frac{1}{|\mathcal{E}_d|}\sum_{e_i \in \mathcal{E}_d} \frac{s_i^H - s_i^{H_A}}{12}. \quad (4)$$

A positive value indicates that the relevant evidence favors the generated answer, while a negative value indicates that it favors the constructed alternative.

Because an average may conceal individual passages that strongly favor the constructed alternative over the generated answer, CAST also applies an explicit contradiction penalty. A citation contributes to this penalty when its support for H_A exceeds a threshold τ_{H_A}:

$$f_{\text{contra}} = \frac{1}{|\mathcal{E}|}\sum_{e_i \in \mathcal{E}} \mathbb{I}\left[s_i^{H_A} \geq \tau_{H_A}\right], \quad (5)$$

where $\mathbb{I}[\cdot]$ denotes the indicator function.

The final CAST confidence score is:

$$C_{\text{CAST}} = \sigma\left(K\left(\bar{\Delta}_d - f_{\text{contra}}\right)\right), \quad (6)$$

where $\sigma(x) = 1/(1 + e^{-x})$, K controls the sensitivity of the score to the evidence, and τ_{H_A} controls the minimum support for the constructed alternative required to trigger the contradiction penalty. By default, CAST uses $K = 4.0$ and $\tau_{H_A} = 8$. If no citations are available, CAST returns the neutral score 0.5 rather than inferring confidence without evidence.

The mean evidence gap and contradiction penalty serve complementary purposes. The former represents the overall direction of the relevant evidence, while the latter reduces confidence when one or more citations strongly favor the constructed alternative. Since both components are derived from visible citation-level scores, the final confidence remains traceable to evidence that the reviewer can inspect directly.

4 Demonstration

4.1 Interactive Reviewer Interface

The CAST demo presents each question-answer pair as a self-contained evidence report. The confidence score and its qualitative category are displayed prominently at the top of the page, see Figure 2. The interface presents the original question and compares the LLM-generated answer with the ground-truth answer available in the example dataset.

The central component is an interactive citation-level evidence visualization. For every retrieved citation, the interface displays a passage preview and its support scores for H and H_A. Horizontal bars visualize the relative support around a neutral baseline: evidence favoring H_A appears on the contradiction side, while evidence favoring H appears on the support side. Hovering over a citation reveals its full text, relevance label, support levels, and the judge's rationale.

The interface also reports aggregate evidence statistics, including the number of supporting, neutral, and contradicting citations and the mean evidence gap. A final summary explains why the confidence is high or low and shows the confidence computation used for the case.

4.2 Financial-Report QA Example

Figure 2 also shows a financial-report question asking how, for one document, total short-term borrowings and long-term debt changed between 2019 and 2020. The LLM answer states that the amount increased by \$6,300 million, whereas the ground-truth answer implies an increase of \$2,746 million.

CAST assigns the answer a confidence of 0.005. All five retrieved citations favor the alternative answer, resulting in five contradicting citations, no neutral citations, no supporting citations, and a mean evidence gap of -0.38. The force-plot-style visualization exposes the separate support scores for each citation, while the summary explains how the conflicting financial values lead to the very low confidence.

This example illustrates how CAST can help an analyst identify both that an answer is unreliable and which retrieved passages challenge it. Instead of manually searching through all sources without guidance, the reviewer can prioritize the citations with the strongest contradictory evidence.

Beyond financial-report question answering, CAST is designed for potential use in evidence-intensive workflows such as risk analysis, KYC, compliance review, and regulatory reporting. It is designed to assist rather than replace human judgment by allowing high-confidence cases to receive minimal review while directing analysts to the most relevant supporting or contradicting evidence in low-confidence cases.

5 Experimental Setup

5.1 Research Questions

We evaluate CAST through the following research questions:

- **RQ1:** Does evidence-based confidence estimation improve confidence quality compared with existing confidence estimation methods?
- **RQ2:** How does CAST perform across different question types?

5.2 Dataset

We evaluate on 1,286 questions over financial report filings in the financial domain, drawn from MMDocRAG's Financial Report split (Dong et al., 2026). We focus on this domain specifically because it is representative of the human-in-the-loop financial review setting motivating this work (§1): long, structured, multi-page filings where an employee needs to verify an LLM-generated answer against source evidence before relying on it, rather than a short-answer or open-domain QA setting where that verification step is less central to the task.

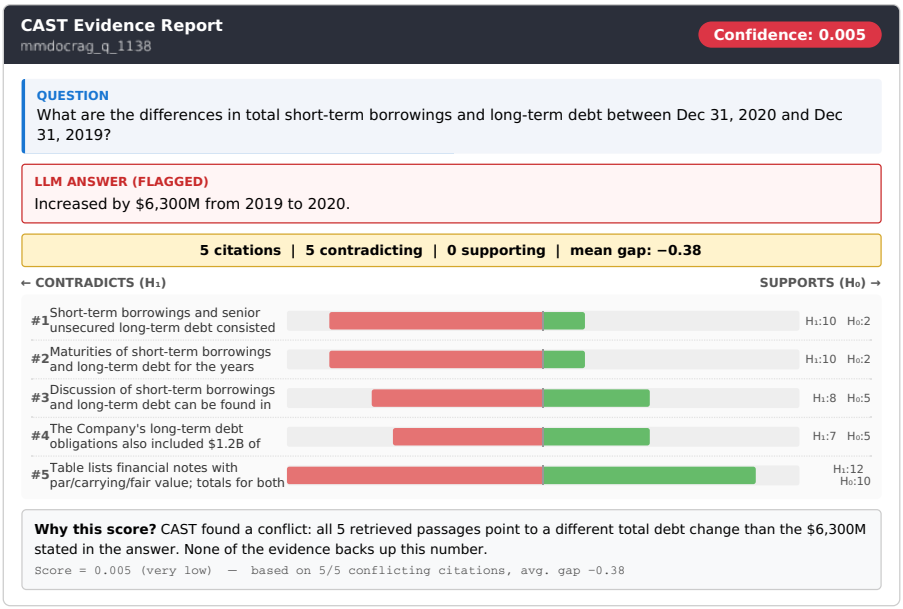

Figure 2: The CAST interactive evidence report demo.

Aggregate results alone cannot show *which* questions a confidence method handles well, so we manually identified five question-type categories within this domain: calculation (758), factual lookup (387), comparative (98), interpretive (40), and aggregative (3), reflecting different reasoning demands from direct fact retrieval to multi-passage synthesis.

5.3 RAG system

We use a standard RAG configuration (Martineau et al., 2023). For each document, the content of each PDF page is extracted, and the pages are encoded into vector representations. Given a question, the system retrieves the most relevant pages by comparing the question with the indexed page representations. The retrieved pages are provided to an LLM, which generates an answer and accompanying reasoning. As part of its output, the LLM identifies relevant citations, i.e., continuous pieces of text, used to support the answer. A final set of citations is selected by removing duplicates and merging overlapping ones. The generated answer and resulting set of citations form the input to CAST.

5.4 Evaluation Pipeline

For each question, CAST evaluates the answer generated by the RAG system and its associated evidence passages (text spans extracted from a page). Each passage is evaluated separately against the generated answer hypothesis and the constructed alternative hypothesis. CAST then aggregates the resulting passage-level support scores into an answer-level confidence score, as described in Section 3.

LLM-generated answer correctness is determined independently using an LLM judge (Zheng et al., 2023). The judge compares each generated answer against the ground truth answer and assigns one of three labels: YES (accurate and complete), PARTIAL (accurate but incomplete), or NO (contradicts the reference). For confidence evaluation, YES answers are treated as correct, while PARTIAL and NO are grouped as incorrect, reflecting the selective-review setting in which both incomplete and contradictory answers require human verification. The full judge prompt is given in Appendix A.1.

In the whole experiment, Gemini-2.5-Flash serves as the LLM to instantiates the alternative answer, judges citation-level support, and evaluates correctness.

5.5 Baseline Confidence Scores

Following the black-box constraint motivated in §2 that most financial RAG deployments query LLMs through APIs with limited access to internal states or full log-probabilities, five of our six baselines are implemented as black-box methods, with logit confidence retained as the sole white-box reference point.

Agreement. Agreement estimates confidence from the proportion of independently sampled answers that are semantically consistent with the original answer (Manakul et al., 2023a). We use five generations per question.

Semantic Entropy. Semantic Entropy groups sampled responses by semantic equivalence and measures uncertainty over the resulting answer clusters (Farquhar et al., 2024). We use five generations per question.

Verbalized Confidence. The model reports its confidence as part of the original answer-generation prompt (Tian et al., 2023).

Verbalized P(True). Following prior self-evaluation approaches (Kadavath et al., 2022), the model is asked whether its generated answer is true, and the resulting probability is used as confidence.

Verbalized P(IKnow). The model estimates whether it knows enough to answer the question correctly (Kadavath et al., 2022). This provides a self-reported knowledge-confidence signal using one additional LLM call.

Logit Confidence.: Confidence is derived from token-level generation probabilities and therefore requires access to model logits but no additional LLM call (Geng et al., 2024).

5.6 Evaluation Metrics

Using the binary correctness labels defined in Section 5.4, we evaluate each confidence score along three complementary dimensions:

Discrimination: Does the score separate correct from incorrect answers? Measured by AUROC (the probability that a randomly chosen correct answer receives a higher confidence score than a randomly chosen incorrect one).

Calibration: Does the score's numeric value track the actual probability of correctness? We assess calibration using reliability diagrams, which compare predicted confidence with observed accuracy across confidence bins.

Rank correlation: Does higher confidence go with higher correctness overall? Measured by Spearman's ρ between the confidence score and the binary correctness label.

6 Results

6.1 Confidence Quality Across Methods (RQ1)

CAST is the strongest method on AUROC, outperforming every baseline (Figure 3): 0.634 versus Agreement's 0.598, the next-best method, while using fewer LLM calls per question than Agreement's sampling-based approach. Meanwhile, many of the baseline methods cannot distinguish between correct and incorrect answers according to their AUROC performance, except for CAST and Agreement.

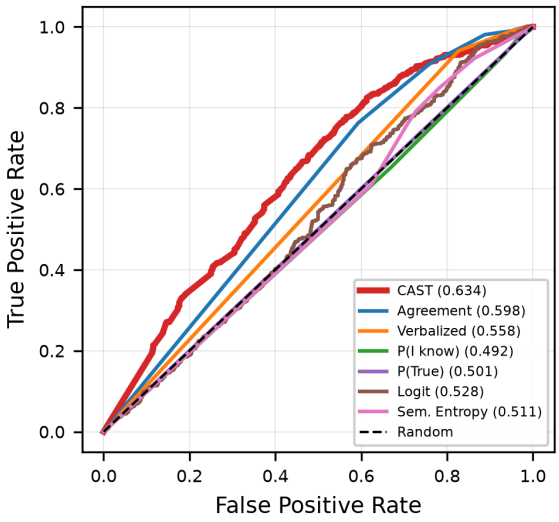

Figure 3: ROC curves for CAST and all baseline confidence methods (AUROC in legend).

Calibration, however, is a weakness for CAST, and not unique to it (Figure 4). CAST shows a V-shaped miscalibration, overconfident above roughly 0.3 confidence. Verbalized and Logit are near-degenerate, concentrating almost all mass in a single high-confidence bin; Verbalized P(IKnow) and Verbalized P(True) are low-statistics and non-monotonic across small bins; Semantic Entropy is flatter but consistently under-tracks accuracy. No method's raw score should be read as a literal probability without post-hoc correction (§7). However CAST does allow for calibration across the predicted confidence range $[0, 1]$, which it covers fully.

6.2 Question-Type Analysis (RQ2)

Aggregate results (§6) do not show *which questions* CAST handles well. Using the five question-type categories defined in §5.2, we compare CAST against Agreement, the next-strongest method, in Table 1; aggregative questions ($n = 3$) are excluded as too few for a reliable estimate.

Table 1: AUROC and Spearman ρ by question type, with the overall result for reference. Best value per row in bold.

	CAST		Agreement	
Type (n)	AUROC	ρ	AUROC	ρ
Calculation (758)	0.626	+0.218	0.560	+0.136
Fact. Lookup (387)	0.629	+0.224	**0.687**	**+0.356**
Comparative (98)	0.699	+0.342	0.643	+0.320
Interpretive (40)	**0.688**	**+0.325**	0.545	+0.103
Overall (1,286)	0.634	+0.233	0.598	+0.205

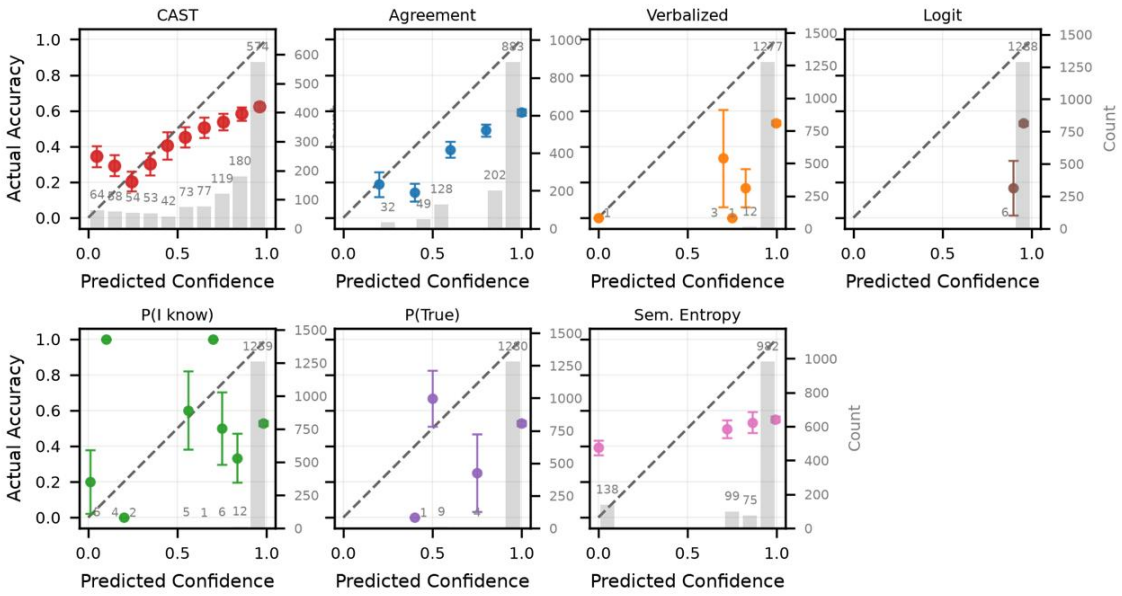

Figure 4: Calibration reliability diagrams for all seven methods: predicted confidence (binned) vs. actual accuracy. Numbers above each point show bin sample size; dashed line indicates perfect calibration.

CAST leads on three of four types (calculation, comparative, and interpretive) with its largest observed advantages on comparative and interpretive questions (+0.056 and +0.143 AUROC, respectively). Agreement wins only on factual lookup (0.687 vs. 0.629), where the answer is typically a single stated value and cross-sample consistency provides a strong signal. One possible explanation is that comparative and interpretive questions more often require synthesizing evidence across passages, creating more opportunity for CAST's competing-hypothesis comparison to distinguish correct from incorrect answers.

Figure 5 shows why this margin varies in size. For calculation and factual lookup, correct and incorrect answers both pile up densely in CAST's 0.9–1.0 score bin, leaving little room to separate the two classes — consistent with their more modest AUROC ($\leq$0.63). Interpretive questions instead show real separation: most incorrect answers cluster in a distinct middle range (0.6–0.8) while correct answers concentrate at the top (0.85–1.0), giving CAST substantially stronger separation than on calculation and factual lookup.

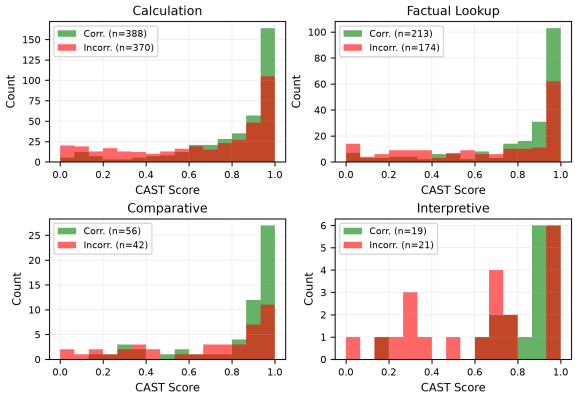

Figure 5: CAST score distributions by question type, split by correctness.

7 Discussion

7.1 Insights

Algorithmic. Testing a generated answer against a constructed alternative, rather than scoring it in isolation, is what lets CAST turn contradiction into a directly measured quantity rather than something a single-hypothesis check is structurally blind to (§2); this paired-hypothesis design is the main source of CAST's discrimination advantage over sampling-based and self-evaluation baselines (§6). That advantage is not uniform, however: CAST's margin over Agreement grows with question complexity and is largest on comparative and interpretive questions, while Agreement remains stronger on direct factual lookup, where cross-sample consistency is a cheap, sufficient signal (§6).

Tool/Demo. CAST's citation-level support and contradiction scores give a reviewer something to audit: which passage disagrees, and how strongly, which a bare scalar from any baseline cannot provide (§4). The cost of producing this explanation is a deployment choice, not a fixed property of

the method: scoring each citation with its own LLM call, as our production deployment does, isolates each judgment for cleaner per-citation rationale but scales linearly with the number of citations; batching all citations into a single call per hypothesis instead reduces this to a small constant number of calls, trading some independence between citation judgments for substantially lower cost. This cost–granularity tradeoff, rather than a single fixed overhead number, is what a deployment should tune to its own latency and budget constraints. Together, these findings position CAST as best suited to evidence-heavy, higher-stakes review tasks where the value of an auditable answer outweighs the cost of scoring its evidence.

7.2 Limitations

CAST is evaluated on the Financial Report subset of MMDocRAG using a single RAG pipeline. Gemini 2.5 Flash serves in three roles: it instantiates the alternative answer, judges citation-level support, and evaluates correctness. Using the same model in all three roles may introduce shared failure modes, and generalization to other tasks, retrieval pipelines, or LLMs remains to be validated.

Correctness labels come from an LLM judge rather than human annotation, so reported metrics measure agreement with the judge, not absolute ground truth, which is a common limitation of LLM-as-judge evaluation (Zheng et al., 2023).

CAST treats retrieved citations as independent when aggregating $\bar{\Delta}_d$, though citations from the same table or section can be correlated rather than independent; this may affect calibration of the aggregate score, though each citation's support level remains individually interpretable.

CAST assumes retrieved evidence is sufficient to distinguish the generated answer from the constructed alternative; under poor retrieval or ambiguous passages, the resulting score may be less reliable. Improving robustness under imperfect retrieval is left to future work.

The current prototype does not yet implement a live production triage workflow with configurable approval and escalation policies. Such policies can be added by comparing C_{CAST} with an application-specific threshold, selected according to the risk tolerance and governance requirements of the target setting.

Finally, CAST has two free parameters, $K = 4$ and contradiction threshold $\tau_{H_A} = 8$ (Eq. 5), set qualitatively rather than swept. Larger K sharpens discrimination but risks overconfidence on borderline cases; smaller K is safer but less discriminative. Larger $\tau_{\bar{H}}$ requires stronger opposing evidence before penalizing confidence; smaller $\tau_{\bar{H}}$ risks over-penalizing weak or ambiguous evidence. Neither was tuned against a held-out AUROC, so other domains may require re-tuning both.

8 Conclusion and Future work

We have introduced CAST (Confidence via Answer Support Testing), a confidence scoring method for RAG systems that reuses retrieved evidence to explicitly test a generated answer against a constructed alternative, rather than relying on sampling or the model's own self-report.

Using MMDocRAG financial reports, CAST is one of two methods that has discrimination power to distinguish between correct and incorrect answers. Moreover, CAST achieves the strongest discrimination of any method evaluated (AUROC 0.634, Spearman ρ +0.233), while requiring fewer LLM calls than the sampling-based baseline method it outperforms. More importantly, for the human-in-the-loop setting motivating this work, CAST's score decomposes by construction into per-citation judgments, which evidence supports the answer and which contradicts it. This allows for an auditable outcome, which a bare confidence number does not provide.

CAST is not a universal replacement for existing confidence methods: its raw score currently requires post-hoc calibration before it can be read as a probability, and its advantage is demonstrated here on one domain in which structured, table-heavy, calculation-oriented questions dominate. We view this as a case for domain-aware confidence estimation rather than a single dominant method, and for evidence-grounded, explainable scores as a design goal in their own right, not merely a byproduct to optimize away in pursuit of higher AUROC.

Future work will explore three directions: (1) **evaluating CAST as a human-review tool** through user studies that examine whether citation-level support and contradiction signals help reviewers identify errors and make better escalation decisions; (2) **generalizing beyond two hypotheses** by testing the generated answer against multiple plausible alternatives; and (3) **generalizing across tasks and RAG configurations** by evaluating CAST on other tasks, QA settings, LLMs, retrievers and retrieval depths.

References

Anonymous. 2026. Replication kit for cast: An interpretable evidence-based confidence estimation tool for rag with human-in-the-loop verification.

Evan Becker and Stefano Soatto. 2024. Cycles of thought: Measuring llm confidence through stable explanations. *arXiv preprint arXiv:2406.03441*.

Kuicai Dong, Yujing Chang, Shijie Huang, Yasheng Wang, Ruiming Tang, and Yong Liu. 2026. Benchmarking retrieval-augmented multimodal generation for document question answering. *Advances in Neural Information Processing Systems*, 38.

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

Luyu Gao, Grégoire Mialon, Nouha Dziri, Asli Celikyilmaz, and Veselin Stoyanov. 2023. Rarr: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508. Association for Computational Linguistics.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 6577–6595. Association for Computational Linguistics.

Rahuj Joshi. 2025. Human-in-the-loop ai in financial services: Data engineering that enables judgment at scale. *Journal of Computer Science and Technology Studies*, 7(7):228–236.

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

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2024. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations (ICLR)*.

Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. 2023. Large language models in finance: A survey. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, ICAIF '23, page 374–382, New York, NY, USA. Association for Computing Machinery.

Xiaoou Liu et al. 2025. Trustworthy retrieval-augmented generation for large language models: A survey. *ACM Computing Surveys*.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023a. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023b. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017. Association for Computational Linguistics.

Kim Martineau, AI Explainable, and AI Generative. 2023. What is retrieval-augmented generation? *IBM Blog*.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878. Association for Computational Linguistics.

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

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

A Prompts

A.1 Correctness Judge

The following prompt (Gemini 2.5 Flash) produces the YES/PARTIAL/NO correctness labels used in §5.4. Few-shot examples are abbreviated for space.

> *You are a highly skilled answer evaluator. Judge whether the LLM ANSWER accurately contains the same information as the GROUND TRUTH.*
>
> **Accuracy:** Does the LLM ANSWER contain accurate information relative to the GROUND TRUTH? Equivalent formats (e.g., "55.6%" vs. "0.556") and more concise/detailed answers are still accurate if the information matches.

Completeness: Does the LLM ANSWER address all aspects covered in the GROUND TRUTH?

Contradiction: Does the LLM ANSWER contradict the GROUND TRUTH?

Return **yes** if completely accurate and equivalent; **partial** if accurate but incomplete; **no** if it contradicts the GROUND TRUTH. [5 few-shot examples omitted.] Respond in schema: `{reasoning: ..., answer: yes|partial|no}`.

A.2 Citation Hypothesis Testing

The following prompt (system instruction: *"You are an expert statistician and hypothesis tester"*) scores each citation against both hypotheses (§3.2), applied once with the generated answer as H and once with the constructed alternative as $\bar{H}$:

Given: a context (possibly); an input query {query}; an answer to the query {answer}; a list of citations from the context {citations}.

We have two hypotheses: Null hypothesis — the answer to the query is correct; Alternate hypothesis — the answer to the query is incorrect.

For each citation: (1) determine its relevance (strong / weak / indirect / no relevance); (2) determine its level of support for the null hypothesis; (3) determine its level of support for the alternate hypothesis, using the support-level definitions in Appendix B.

A.3 Alternative Answer Generation

CAST classifies each answer as *binary*, *numeric*, or *qualitative*; then applies the corresponding template with Gemini 2.5 Flash (temperature 0):

Binary: *Question: {query}. Generated answer: {answer}. Generate the opposite answer: if "yes," write "no"; if "no," write "yes." Keep it concise (1–2 sentences).*

Numeric: *Question: {query}. Generated answer: {answer}. Generate a wrong answer where ONLY the number changes: perturb by 10–40%, keep everything else identical.*

Qualitative: *Question: {query}. Generated answer: {answer}. Generate a plausible alternative answer: different entity/conclusion, same structure and detail level.*

B Support Rubric

The full 13-category rubric definitions used by the prompt above are given in Table 2.

Table 2: Citation-level support rubric used by CAST. Score 6 has two interpretations: balanced evidence and independence.

Score	Support level	Interpretation
12	Absolute Support	The citation explicitly and exhaustively reflects all claims, with no omission, interpretation, or speculation required.
11	Near-Absolute Support	The citation reflects essentially all claims, with only a negligible omission or an extremely minor interpretive step.
10	High-To-Absolute Support	Almost all claims are explicitly reflected; any omission is trivial and requires at most negligible inference.
9	High Support	The vast majority of claims are clearly reflected, with only minor details missing or requiring slight inference.
8	Moderate-High Support	The core claims are well reflected, but secondary claims are missing or require moderate interpretation.
7	Moderate Support	The central claims are partially reflected, with notable omissions or clear interpretive assumptions required.
6	Balanced or Neutral Support	The citation is relevant but ambiguous, providing neither strong validation nor strong contradiction.
5	Moderate-Low Support	The citation only superficially aligns with the hypothesis and contains substantial omissions or minor discrepancies.
4	Low Support	The citation shows little alignment and contains major gaps, omissions, or inconsistencies that weaken the hypothesis.
3	Very Low Support	The citation strongly diverges from the hypothesis and undermines a substantial portion of its claims.
2	Minimal Support	The citation conflicts with almost all claims; only trivial or incidental details remain uncontradicted.
1	Negligible Support	The citation is close to complete contradiction, leaving almost no meaningful alignment with the hypothesis.
0	Zero Support	The citation directly contradicts all claims of the hypothesis.