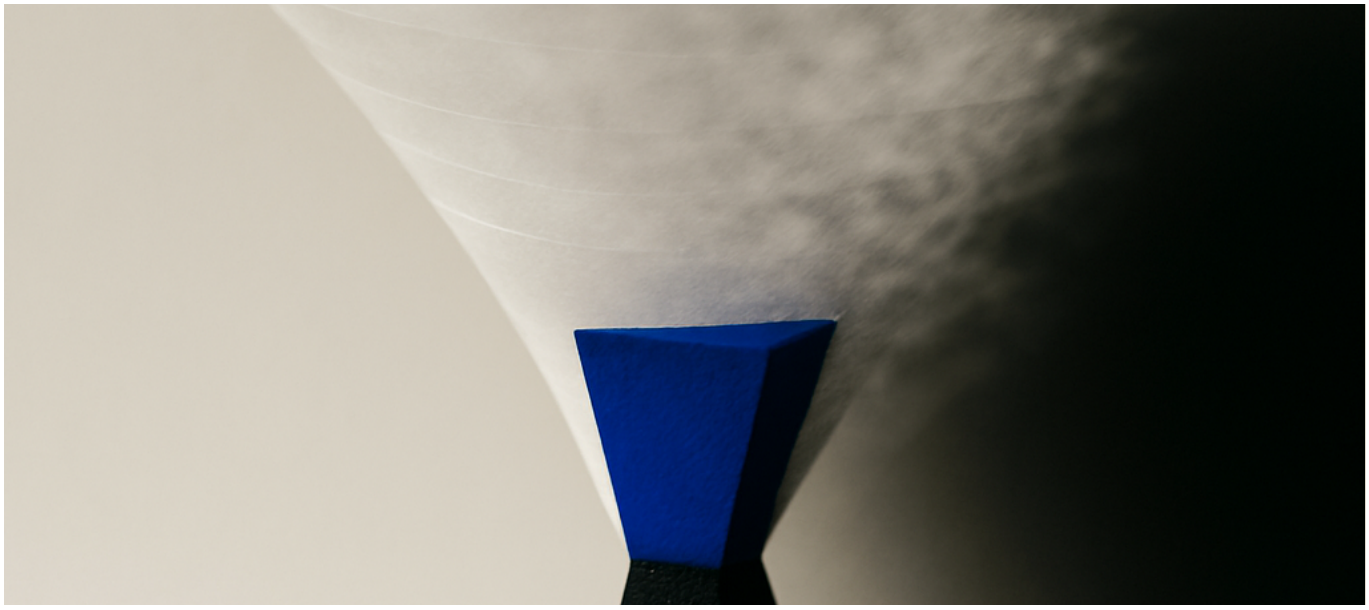

172 Billion Tokens Show Every LLM Fabricates Above 10% at 200K Context

The best of 35 open-weight models still invented answers 1.19% of the time about entities that provably did not exist in the source document. Push the context window to 200K and every model crossed 10%.

The Paper

How Much Do LLMs Hallucinate in Document Q&A Scenarios? A 172-Billion-Token Study Across Temperatures, Context Lengths, and Hardware Platforms was submitted to cs.CL on 9 March 2026 and last updated 1 July 2026. Single author: JV Roig, affiliated with Kamiwaza AI, which also publishes the underlying RIKER evaluation framework documentation.

The one-sentence finding: across 35 open-weight models, three context lengths, four temperature settings and three accelerator platforms — 172 billion tokens over

more than 4,000 runs — no configuration reached zero fabrication, and fabrication scales sharply with context length.

The Method

RIKER's core trick is a design decision, not a model: **build questions around entities that are provably absent from the document, so any factual-sounding answer is unambiguously a fabrication.**

The author calls it a "ground-truth-first evaluation methodology that enables deterministic scoring without human annotation." Translated for engineers: instead of asking a model a question and then arguing about whether its answer was correct, you construct the question so that the only correct behaviour is refusal. If the model names a value, cites a figure, or describes a clause, it made it up. No judge model, no annotator panel, no scoring ambiguity.

The study tracks a second failure mode alongside fabrication: *coherence loss*, meaning infinite generation loops, measured through output truncation rate. That turns out to matter as much as the headline number.

Three hardware platforms were tested — NVIDIA H200, AMD MI300X and Intel Gaudi 3 — with results consistent across all three. The author's conclusion: "deployment decisions need not be hardware-dependent."

The Results

There is no zero

GLM 4.5, the best performer, fabricated in 1.19% of cases at 32K context under its optimal temperature setting. That is the floor of the entire 35-model field, achieved under the friendliest conditions the study offers.

Long context is the failure amplifier

Fabrication nearly triples moving from 32K to 128K. At 200K, every model tested exceeded 10%.

Long context windows are marketed as the fix for large document sets. This data

says the fix measurably degrades the property you were relying on. One practitioner framing circulating after publication called it a result that “should terrify every RAG developer” — because the industry’s default answer to “we have too many documents” has been “use a bigger window.”

Coherence loss follows the same curve, only worse. Qwen3-4B-Instruct went from 0.3% coherence loss at 32K to 37.0% at 200K — a 123x increase. The RIKER documentation notes that aggregation queries requiring multi-document synthesis trigger coherence loss far more than single-document queries, which stay below 0.5%.

Read that pairing again: the query type most likely to loop is exactly the one you built a 200K window to serve.

The temperature result is genuinely counterintuitive

T=0.0 produced the best overall accuracy in roughly 60% of cases. But higher temperatures reduced fabrication for the majority of models.

And T=0.0 was the worst setting for looping by a wide margin — up to 48x higher coherence loss than T=1.0. The RIKER documentation cites one case dropping from 184 truncation instances to 4 simply by raising temperature.

So the greedy-decoding default that most production RAG stacks inherited from tutorials optimises accuracy while maximising another failure mode. **Deterministic decoding is not the safe default; it is a trade you probably made without knowing it was a trade.**

Limitations

The author states the scope constraints plainly, which I appreciate:

- **Open-weight models only.** No proprietary frontier APIs. We do not know from this data whether GPT-class or Claude-class models sit above or below the 1.19% floor.
- **Three context lengths.** 32K, 128K, 200K. The shape between and beyond those points is interpolated, not measured.
- **Narrow fabrication definition.** Claiming to find non-existent information — not all hallucination subtypes.

Two more I would add as reader caveats. First, this is a single-author study, and the evaluation framework is published by the author's own employer; independent reproduction on RIKER would strengthen the result considerably. Second, it is an arXiv preprint, not peer-reviewed as of the 1 July 2026 update.

My take: the narrow fabrication definition is a feature, not a weakness, and it makes these numbers a floor rather than an estimate. Asking about a provably absent entity is the easiest possible case for a model to get right — the correct answer is "not in the document," full stop. Behaviour on harder cases (present-but-ambiguous information, conflicting sources, partial matches) is almost certainly worse, not better. I would treat these figures as best-case bounds for grounding reliability.

The hardware consistency finding I read as the most immediately actionable and least surprising result. Fabrication is a property of weights and decoding, not of the accelerator — which means you can stop suspecting your inference stack when grounding fails.

Why Practitioners Should Care

Stop treating long context as a retrieval strategy

If your RAG design stuffs 200K tokens into the window because chunking and reranking felt like too much work, this paper prices that decision: a fabrication rate above 10%, and on aggregation queries a coherence-loss risk that can reach double digits.

The counter-move is unglamorous: tighter retrieval, smaller contexts, more aggressive filtering before the model sees anything. **The cheapest hallucination reduction available in 2026 is sending the model less text.**

Test fabrication resistance separately from retrieval accuracy

The paper's sharpest structural claim is that "grounding ability and fabrication resistance are distinct capabilities — models that excel at finding facts may still fabricate facts that do not exist."

Most eval suites I see in the wild measure only the first. They ask whether the model found the answer, never whether it invents one when there is nothing to find.

If your model selection process has no absent-entity test set, you have no data on half of the relevant behaviour.

Build one. It is cheap: take your real corpus, generate questions about entities you know are not in it, and score refusal rate. That is the RIKER idea in miniature, and it requires no annotators.

Tune temperature per failure mode, not per vibe

Treat decoding temperature as a trade among accuracy, fabrication and looping — and measure all three. A configuration that wins on accuracy in 60% of cases while running up to 48x more infinite-generation loops is not obviously right for a production endpoint with timeouts and token budgets.

This pattern is not confined to document Q&A

The failure generalises to any place a model is asked to name something that should exist. Sonatype found AI coding assistants hallucinate 27.75% of package upgrades, recommending over 10,000 non-existent versions — in a completely different task domain. When the model does not know, it produces something plausible rather than nothing. That is the shape of the problem, and it does not care whether the missing entity is a contract clause or a semver tag.

What I would do Monday

- **Measure your own floor.** Absent-entity test set against your real corpus, at the context length you actually run in production. Not 32K if you deploy at 128K.
- **Sweep temperature and log truncation rate**, not just accuracy. If you have never counted infinite-generation loops, you do not know your current coherence-loss number.
- **Put a refusal path in the product.** Given that no configuration across 35 models reached zero fabrication, “the model might invent this” is a permanent design constraint, not a bug to be patched. Citations, span highlighting and explicit “not found in source” states are the mitigation layer.

The honest reading of this paper is not that LLMs are unusable for document Q&A. It is that the fabrication rate is a measurable system parameter with known drivers — context length, model family, decoding temperature — and most teams currently

have no measurement at all.

Long context is not a retrieval strategy; it is a fabrication multiplier, and the only teams who will be able to prove otherwise are the ones already measuring their own floor.