COMMUNITY RESEARCH REPORT · NOT PEER-REVIEWED · V1.1

Show Your Work: Can Small Open Models Read SEC Filings with Evidence a Machine Can Check?

Filing-Evidence: a pre-registered benchmark of ten open models on one consumer GPU

Krishna Nandula · Pramana Research 17 November 2026 Version 1.1

Status: an independent community research report, published by Pramana Research on 17 November 2026. It has **not been peer-reviewed** and has not been submitted to or accepted by any conference or journal. It is also available as a preprint on arXiv: arXiv:2611.24990 [cs.SE]. Version 1.1 (20 November 2026) brings this page in line with the arXiv version: a stricter rule for questions whose units are stated off-page (30 questions rather than 24), "verbatim quote" in place of "verbatim line", and a few different worked examples on the model pages; no headline figure changed. Corrections and feedback are welcome on GitHub.

In brief

We asked ten openly available AI models, small enough to run on one gaming graphics card, to find 1,000 numbers in recent SEC filings. Every answer had to come with the page and the exact line it was read from, so a computer could check it, or to say "not found" when the number wasn't there.

84.5%

answered correctly with a checkable quote by the best model (Qwen3.5-9B, check + retry)

16 in 1,000

wrong answers that still passed every automatic check, at best (Qwen3.5-9B, thinking). Most models: 64–167

57%

of those wrong-but-passing answers had the right digits in the wrong units ("thousands" vs "millions")

4 of 5

models with a thinking mode made significantly fewer wrong-but-passing answers with it

1. **Small open models can do much of this job, and the gap between them is large.** In the plain setup accuracy ranges from 17.1% to 82.7%. Qwen3.5-9B is ahead of every other model by a statistically clear margin.
2. **Demanding a checkable quote works, but it isn't enough on its own.** The checker stops most wrong answers. Yet the answers that pass every check include 2.4–31.8% wrong ones, depending on model and setup. Those are the dangerous errors: correct-looking evidence, wrong number.
3. **Those errors are specific and fixable.** They are almost all units read wrongly or the neighbouring column (another year or quarter) of the right line, not invented numbers.
4. **Letting the model think helps more than letting it retry.** Thinking cut wrong-but-passing answers for every model that has the mode, at 8–67 times the generation cost. Telling a model which check failed and letting it retry adds 0.1–3.9 points of accuracy but never reduces the dangerous errors, and for 4 models it increases them: told that its quote failed, a model often finds a line that contains its wrong number.

Abstract

Language models are increasingly used to read financial documents, but a fluent answer is not a verified one. We ask whether small open models, the kind that run on a single consumer GPU, can extract numbers from SEC filings *with evidence a machine can check*: the page and the verbatim line the number was read from. We built Filing-Evidence, 1,000 questions over 90 recent 10-K and 10-Q filings of 45 US companies, with gold answers taken from the filers' own inline XBRL tags, in eight kinds of content including segment tables, numbers in prose and questions whose answer is not on the pages shown. The protocol, metrics and statistics were registered before any model ran. We evaluated ten instruct models from ten companies (7–21B parameters) on one RTX 4090 in three setups: plain, check-and-retry, and the model's own thinking mode. Accuracy with checkable evidence ranges from 17.1% to 82.7% in the plain setup; the best result is 84.5% (Qwen3.5-9B). An automatic checker rejects most wrong answers, yet 16 to 167 answers per 1,000 pass every check and are still wrong; across the ten models 57% of these are scale errors and 30% the wrong column of the right line. Thinking reduces such errors significantly for

> 4 of 5 models; retrying after a failed check raises accuracy by up to 3.9 points but never reduces them. We release the questions, every model answer, the code and the protocol.

1 · Introduction

An analyst who copies a number from a filing can show where it came from. A language model that does the same job usually can't, or it gives a citation nobody checks. For a model to be trusted with this work its answers have to be *auditable*: each number should come with evidence that can be verified quickly, preferably by software.

The question is simple to state. If every answer must name the page and quote the exact line containing the number, how often are small open models right, how often are they wrong in a way the quote doesn't reveal, and do two cheap, training-free interventions help: checking the answer automatically and letting the model try again, or switching on the model's own reasoning mode? We ask it of models that fit on one 24 GB consumer GPU, because those are the models an organisation can run on its own hardware, with its own documents, at low cost.

We registered three research questions before running any model (the full protocol is in the repository):

- **RQ1.** How reliably do ~10B-parameter open instruct models extract a requested number from an SEC filing with evidence that can be checked automatically, and answer "not found" when the number isn't on the pages provided?
- **RQ2.** Where do they fail: in which kinds of filing content, and with which kinds of error?
- **RQ3.** How much do (a) automatic checking with one retry and (b) the model's own thinking mode help, and at what cost in tokens and time?

Our working thesis was that a small open model on one consumer GPU can be a trustworthy reader of SEC filings *if* every answer must carry evidence a machine can check. The results support the first half of that sentence more than the second: requiring checkable evidence makes most errors visible, but a specific class of errors passes through it. Section 7 discusses what closes that gap.

2 · Benchmark design

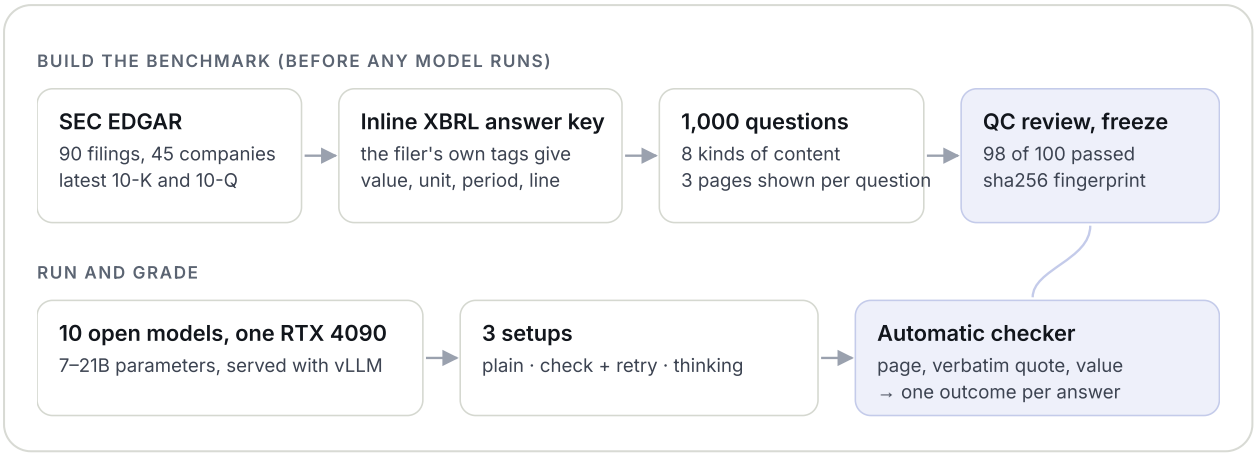


Figure 1. How the benchmark was built and run. Everything above the line was finished, reviewed and fingerprinted before any model saw a question.

2.1 Filings and answer key

We took the most recent 10-K and 10-Q filed on or before 15 November 2026 by 45 large US companies (90 filings): forty in nine sector groups chosen to cover different statement layouts (technology, banks, other financials, insurers, REITs, utilities, healthcare, consumer, industrials and others), plus five added for segment coverage. Banks, insurers, REITs and utilities are included deliberately, because their statements use line items and layouts unlike an industrial company's.

Gold answers come from the filings' own **inline XBRL** tags: the machine-readable layer that filers embed in the document to label each number with its concept, period, unit, scale and dimensions. We use only tags whose number is *displayed* on a page, and record the page and the line where it is shown. The answer key is therefore the filer's own statement, not our annotation.

2.2 Questions

Each question names an item, a company and a period in words ("What was Apple Inc.'s total liabilities as of June 27, 2026?") and comes with three pages of the filing: the page that shows the answer, an adjacent page and a random distractor, in random order. Questions fall into eight **slices** of 125 each:

Slice	What it tests	Companies
S1	Main statements, current period	44
S2	Main statements, prior period	44
S3	10-Q quarter vs year-to-date	36
S4	Notes tables	44
S5	Segments	29
S6	Numbers in prose	43
S7	Units and per-share	42
S8	Not in the provided pages	45

S8 asks for something that is *not* on the pages provided: a period not shown, an item the company doesn't report, a quarter asked of an annual report, or an item that is elsewhere in the filing but not on these pages. The correct answer is "not found". Strict rules decide which facts can become questions: the number must be a column figure on a row with a printed label, the row must be fully populated, the period's year must be visible on the page, and the question names the item by its printed row label when that label is unique on the shown pages.

2.3 Quality control and freeze

Before any model ran, 100 questions (stratified across slices) were reviewed one by one against the rendered filing by an AI assistant, under the author's direction, in each round; the protocol required at least 98 correct. Rounds one to three failed on wording (87, 76 and 96 of 100), never on gold values: all 400 gold values reviewed across the four rounds were judged right. Round four passed with 98 of 100, and the dataset was frozen with a SHA-256 fingerprint (`4fa8f252292167fb…`). Every result cites it, and the published data verifies against it.

2.4 The checker

Models answer in a fixed JSON format: the value, its scale ("in millions"), the page number and the line quoted from the page, or `{"status": "not_found"}`. The checker then applies four automatic tests (Figure 2) and compares the value with the answer key. The sign is not graded, because filers display negative numbers in inconsistent ways; values are compared after scale, and per-share amounts and percentages as displayed.

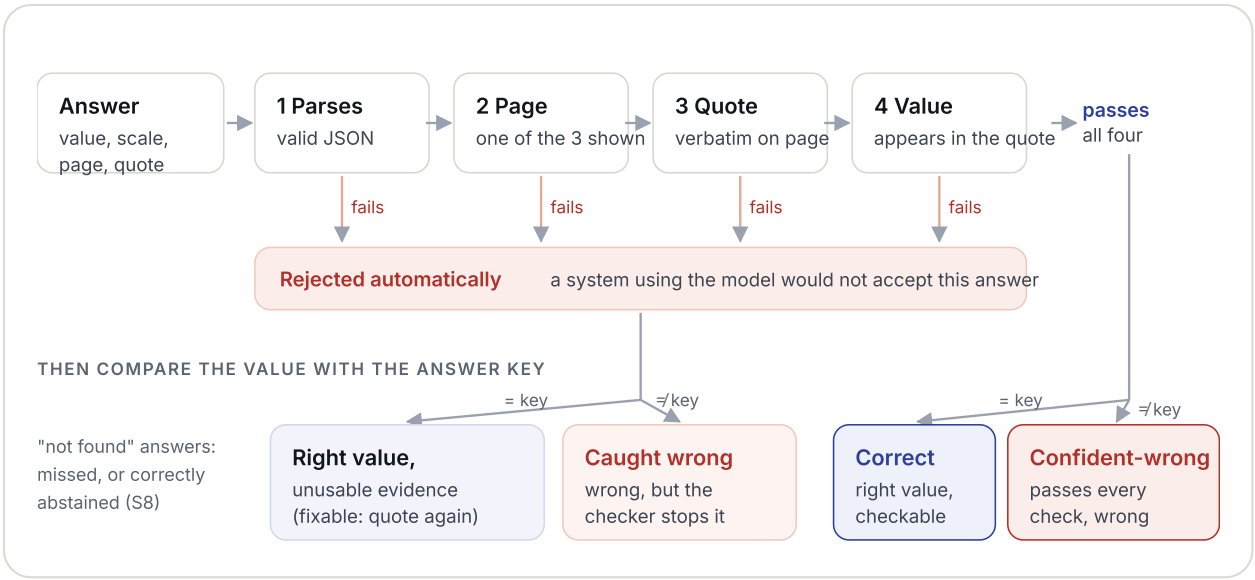


Figure 2. Four checks, then the answer key. The checker sees only the answer and the pages, never the gold, so it could run in production. Its most important outcome is the one it can't detect: **confident-wrong**, a wrong value whose evidence passes every check.

Every answer receives exactly one outcome. **Accuracy** is the share of the 875 answerable questions (S1–S7) answered correctly *with evidence that passes the checks*: a right number with an unusable quote does not count. **Confident-wrong rate** is the share of all 1,000 answers that pass every check but are wrong, and is the metric that matters most for a system that would act on answers the checker accepts. **Abstention accuracy** is the share of S8 questions answered "not found". Every rate carries a 95% Wilson interval; differences are tested with paired exact McNemar tests on the same questions and Holm-corrected within each family of comparisons.

3 · Models and setups

Ten general instruct models, each from a different company, all served with vLLM 0.33.0 on one NVIDIA RTX 4090 (24 GB) at temperature 0, with each model's own chat template and its exact Hugging Face revision pinned. Models are used as published: no fine-tuning, no finance-specific models, no prompt tuning per model; the prompt is identical for every model. Qwen3-8B, the model of the pilot study that motivated the benchmark, was run as a supplementary baseline and is reported separately.

Model	Company	Params	Precision	Released	Thinking mode	Licence
Qwen3.5-9B	Alibaba	9.7B	bf16	2026-02	yes	Apache 2.0
Granite-4.2-8B	IBM	8.8B	bf16	2026-08	yes	Apache 2.0
Ministral-3-8B	Mistral	8.9B	bf16	2025-10	—	Apache 2.0
OLMo-3-7B	AllenAI	7.3B	bf16	2025-11	—	Apache 2.0
Nemotron-Nano-9B-v2	NVIDIA	8.9B	bf16	2025-08	yes	NVIDIA Open Model
gpt-oss-20b	OpenAI	20.9B	mxfp4	2025-08	yes	Apache 2.0
Phi-4	Microsoft	14.7B	fp8	2024-12	—	MIT
Gemma-3-12B	Google	12.2B	fp8	2025-03	—	Gemma Terms of Use
GLM-4.6V-Flash	Zhipu (Z.ai)	10.3B	bf16	2025-12	yes	MIT
Llama-3.1-8B	Meta	8.0B	bf16	2024-07	—	Llama 3.1 Community
Qwen3-8B (suppl.)	Alibaba	8.2B	bf16	2025-04	yes	Apache 2.0

Parameters as published (Qwen3.5-9B and GLM-4.6V-Flash include a vision encoder that was not loaded; gpt-oss-20b is a mixture-of-experts model with about 3.6B parameters active per token). "Released" is the month the Hugging Face repository was created. FP8 is weight quantisation by vLLM at load, used where bf16 weights would not fit; gpt-oss-20b uses its native MXFP4 weights.

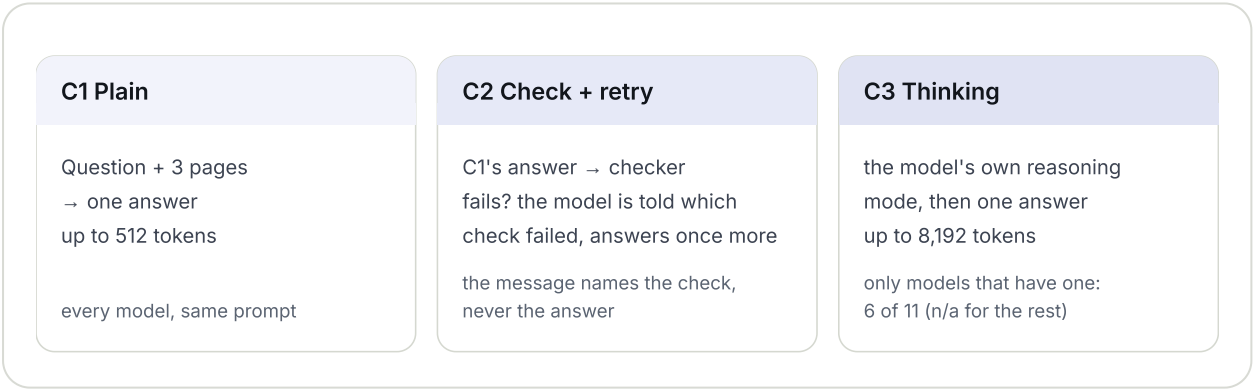


Figure 3. The three setups. C2 reuses C1's first answer, so the two differ only by the retry. gpt-oss-20b has no non-reasoning mode; its C1 and C2 use the lowest reasoning effort, C3 the highest.

4 · RQ1: How reliable are they?

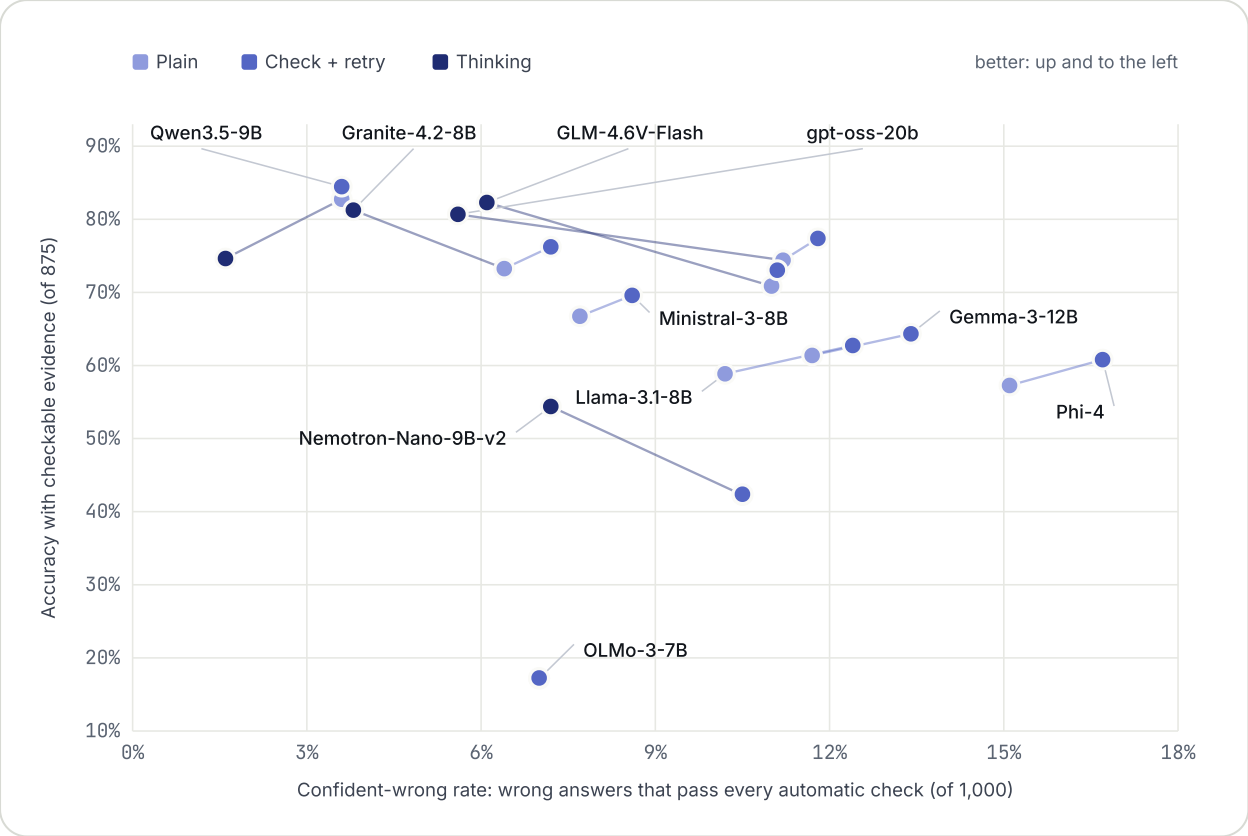


Figure 4. Accuracy against confident-wrong rate for each model in each of its setups (pale: plain; mid: check + retry; dark: thinking). The ideal model sits in the top-left corner. Hover a point for its values.

Accuracy in the plain setup spans a wide range, from 17.1% for OLMo-3-7B to 82.7% (80.1–85.1%) for Qwen3.5-9B. The paired tests separate the ten models into clear tiers (Figure 6): Qwen3.5-9B is significantly more accurate than every other model; gpt-oss-20b, Granite-4.2-8B and GLM-4.6V-Flash cannot be told apart from each other; then come Ministral-3-8B, a group of Gemma-3-12B, Llama-3.1-8B and Phi-4, and finally Nemotron-Nano-9B-v2 and OLMo-3-7B. The best single result is 84.5% (Qwen3.5-9B, check + retry).

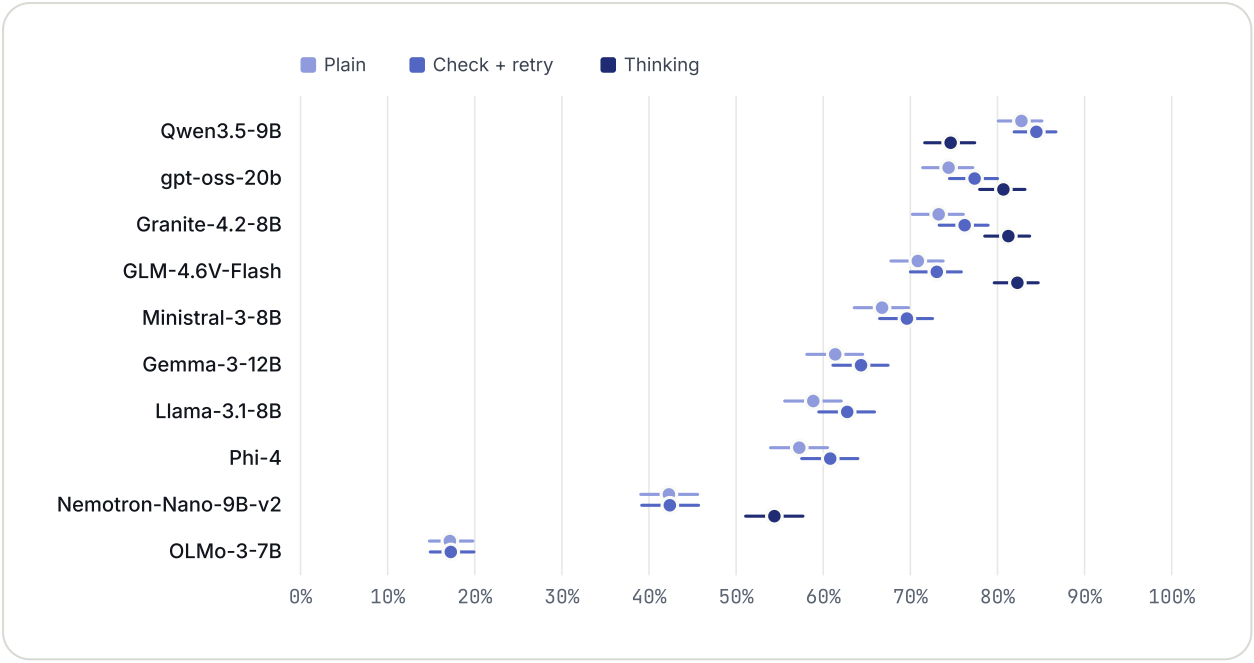


Figure 5. Accuracy with 95% Wilson intervals, per model and setup, ordered by the plain setup.

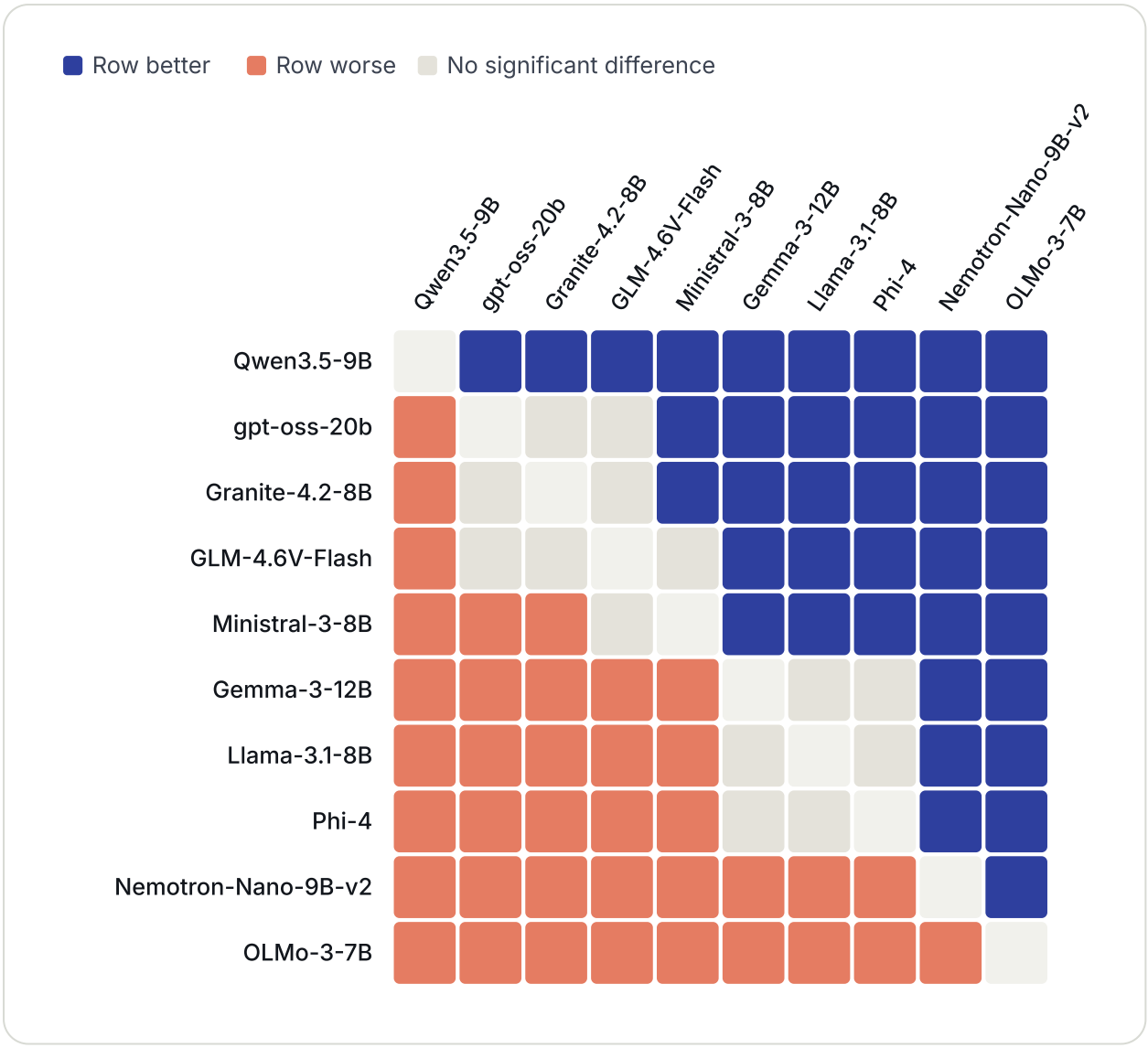


Figure 6. Which differences in plain-setup accuracy are real. Each cell compares the row model with the column model on the same 875 questions (paired exact McNemar, Holm-corrected over 45 pairs). Grey cells: the data cannot separate the two.

The checker is the reason these accuracy figures mean something. An answer only counts when its quote is on the cited page and contains the number. Without that requirement the figures would be higher for every model: a right number with an unusable quote is common for weaker models (Figure 7). For a reader of the output that difference is the whole point: a right number you can't verify is not much more useful than a wrong one.

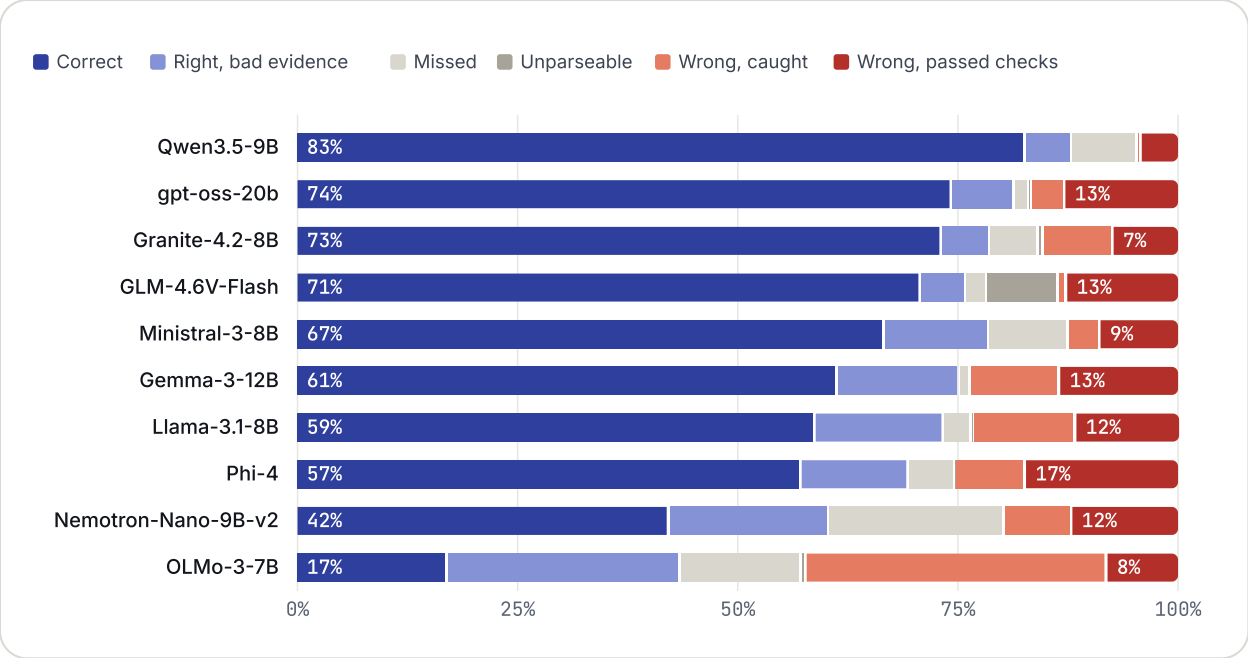


Figure 7. What happens to the 875 answerable questions in the plain setup. Left of the grey band, the model was right; right of it, wrong. The dark red segment is the one that matters most: wrong answers that pass every check.

"Not found" is handled well by most models. Abstention accuracy in the plain setup ranges from 77.6% to 100.0%: models rarely invent a number for something that isn't on the page. Most invented answers come from one kind of trap: a **quarterly figure asked of an annual report**, which the filing can't contain. Gemma-3-12B, the weakest abstainer among the ten (77.6%), answered 17 of those 31 correctly, and the supplementary Qwen3-8B only 6.

5 · RQ2: Where do they fail?

5.1 By kind of content

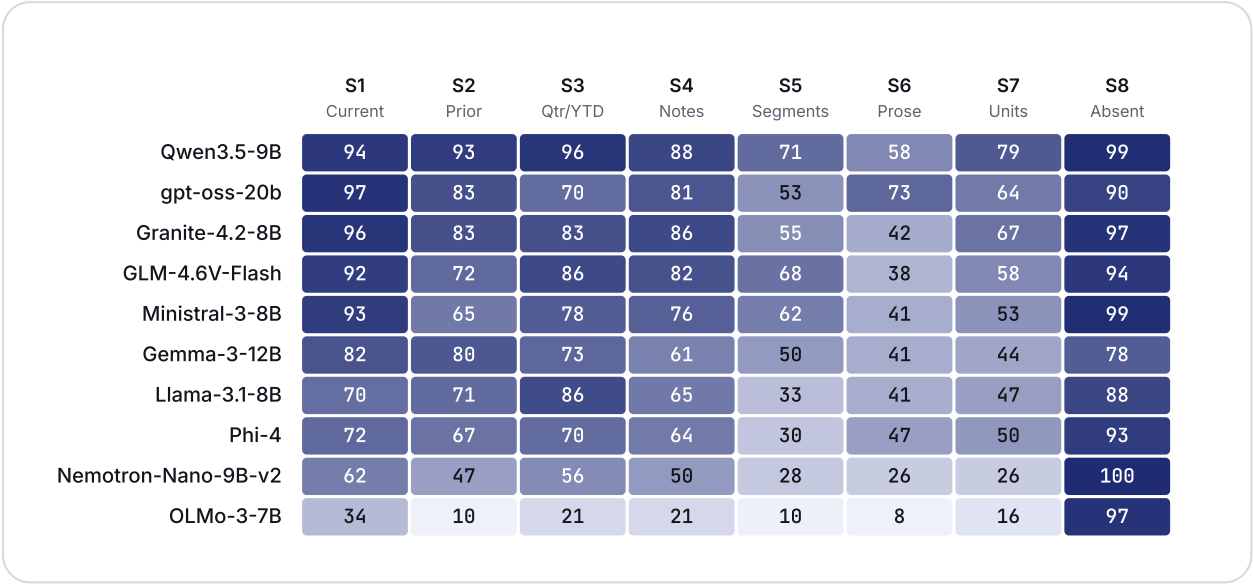


Figure 8. Plain-setup accuracy by slice (S8: correct "not found"). Darker is better.

The main statements are the easiest content: pooled over the ten models, 79.2% of current-period questions are answered correctly. Accuracy falls on prior-period columns (67.1%), in notes (67.4%), and most of all for **segments** (46.0%), **numbers in prose** (41.4%) and **units and per-share** figures (50.5%). Even Qwen3.5-9B, strong everywhere else, answers only 57.6% of prose questions correctly. These are the places where a filing's layout stops doing the work: a segment table's columns are business lines rather than years, and a number in a sentence has no column header at all.

5.2 What the dangerous errors are

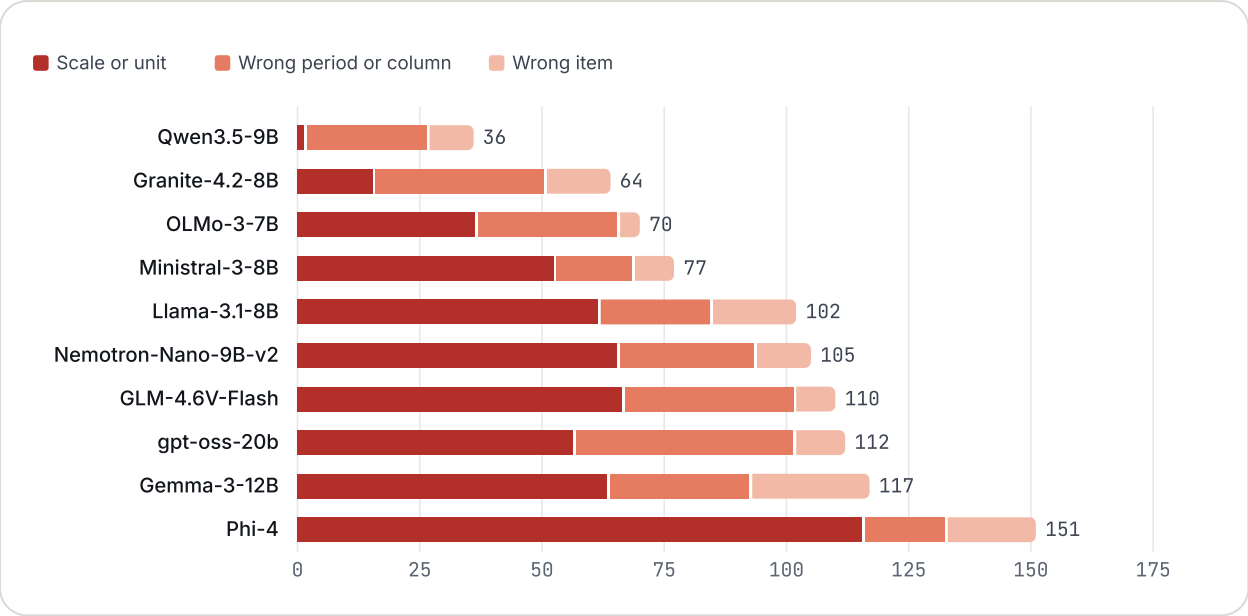


Figure 9. The confident-wrong answers of each model in the plain setup, by kind of error. Scale or unit: the right digits with the wrong multiplier. Wrong period or column: another figure from the right line. Wrong item: a different line.

Across the ten models there are 944 confident-wrong answers in the plain setup. Only 13% are a wrong item. The rest are near misses on the right line:

- **Scale or unit (57%).** The model quotes the correct line and gives the correct digits, but states the wrong scale, most often "none" for a table headed "(In millions)", or reads a share count as shares rather than millions of shares. The checker can't catch this, because the quote is genuine and contains the number; the units are stated once, at the top of the table.
- **Wrong period or column (30%).** The model quotes the right line and takes the figure for another year, quarter or year-to-date column. The quote again contains the number, so it passes.

Scale errors are concentrated in a few models: they are 116 of Phi-4's 151 confident-wrong answers, but 2 of Qwen3.5-9B's 36. The secondary metric *accuracy ignoring scale* shows how much of each gap is units rather than finding the number: for Phi-4 it is 70.5% against an accuracy of 57.3%.

Bank of America Corporation · 10-Q · slice S2 · page 48 (excerpt)

“What was Bank of America Corporation's net change in derivatives for the nine months ended September 30, 2025?”

The line on page 48, as the model saw it:

> Net change in derivatives (1,496) 636 (2,874) 3,145

Model's answer	`-2874` in millions, page 48
Its quote	`Net change in derivatives (1,496) 636 (2,874) 3,145`
Checks	parses ✓ · page shown ✓ · quote verbatim ✓ · value in quote ✓
Answer key	`3,145`
Outcome	CONFIDENT-WRONG the right line, another column (**model** vs **answer key**)

Figure 10. A real confident-wrong answer from the strongest model in the plain setup. Every check passes, because the quoted line is genuine and contains the number; only the column is wrong. Of the ten models on this question, 0 answered correctly.

6 · RQ3: Retry or think?

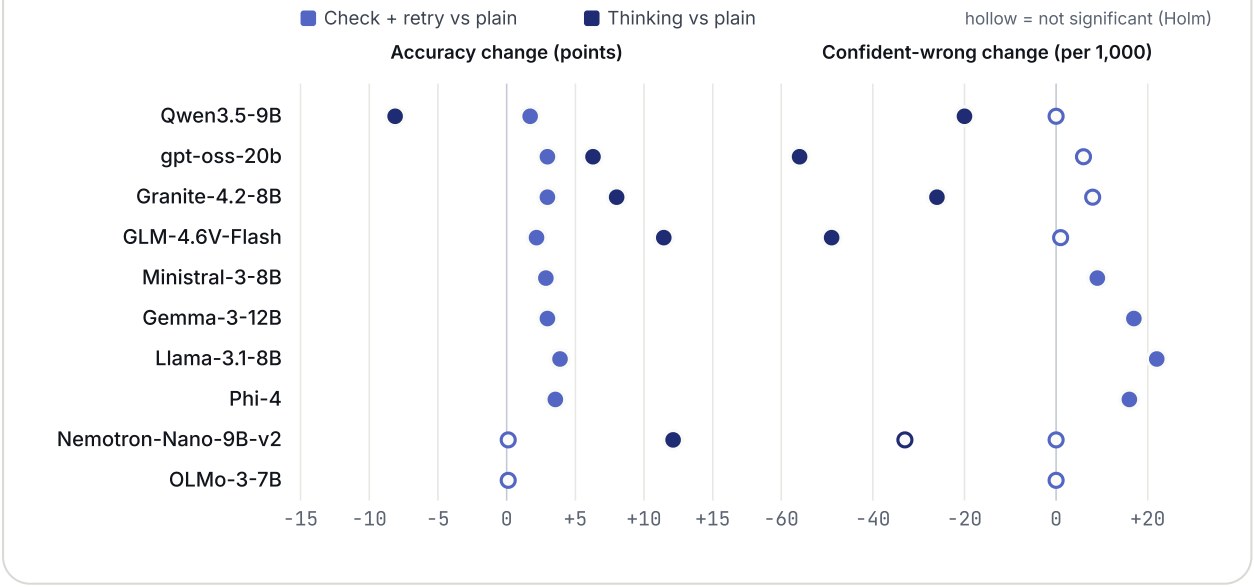


Figure 11. Change from the plain setup, per model. Left: accuracy (points). Right: confident-wrong answers per 1,000 (negative is better). Filled: significant after Holm correction across all within-model comparisons.

6.1 Check + retry

When an answer fails a check, the model is told which one ("the quoted line does not appear on page 54") and answers once more; it is never told the answer. This raises accuracy by 0.1 to 3.9 points, significantly for 8 of the ten models, at little cost: only answers that fail a check are asked again (5–54% of answers, depending on the model).

But retrying **never reduces** confident-wrong answers, and for 4 models (Ministral-3-8B, Phi-4, Gemma-3-12B, Llama-3.1-8B) it significantly increases them. The mechanism is visible in the answers: told that its quote failed, a model often keeps its wrong number and finds a line that contains it, which now passes. For Phi-4, 16 of 177 retried answers moved from "caught" to "confident-wrong" in this way. A retry teaches the model to satisfy the checker, not to be right. Nemotron-Nano-9B-v2 shows the opposite failure: asked to retry, it answered "not found" almost every time.

6.2 Thinking

Five of the ten models have a reasoning mode (as does the supplementary Qwen3-8B). Thinking improved accuracy for 4 of them, by up to 11.4 points (GLM-4.6V-Flash), and reduced confident-wrong answers for all 5: Qwen3.5-9B 36→16, Granite-4.2-8B 64→38, Nemotron-Nano-9B-v2 105→72, gpt-oss-20b 112→56, GLM-4.6V-Flash 110→61 answers per 1,000 (4 of these reductions are significant). The lowest confident-wrong rate in the study is Qwen3.5-9B with thinking: 16 in 1,000.

The exception is Qwen3.5-9B, the strongest plain model, whose accuracy fell from 82.7% to 74.6% with thinking. The cause is the token budget, not the reasoning: it reasoned at length (mean 3,574 tokens) and hit the pre-registered 8,192-token limit on 180 questions, leaving 88 with no parseable answer. On the questions it finished, its accuracy was 85.1%.

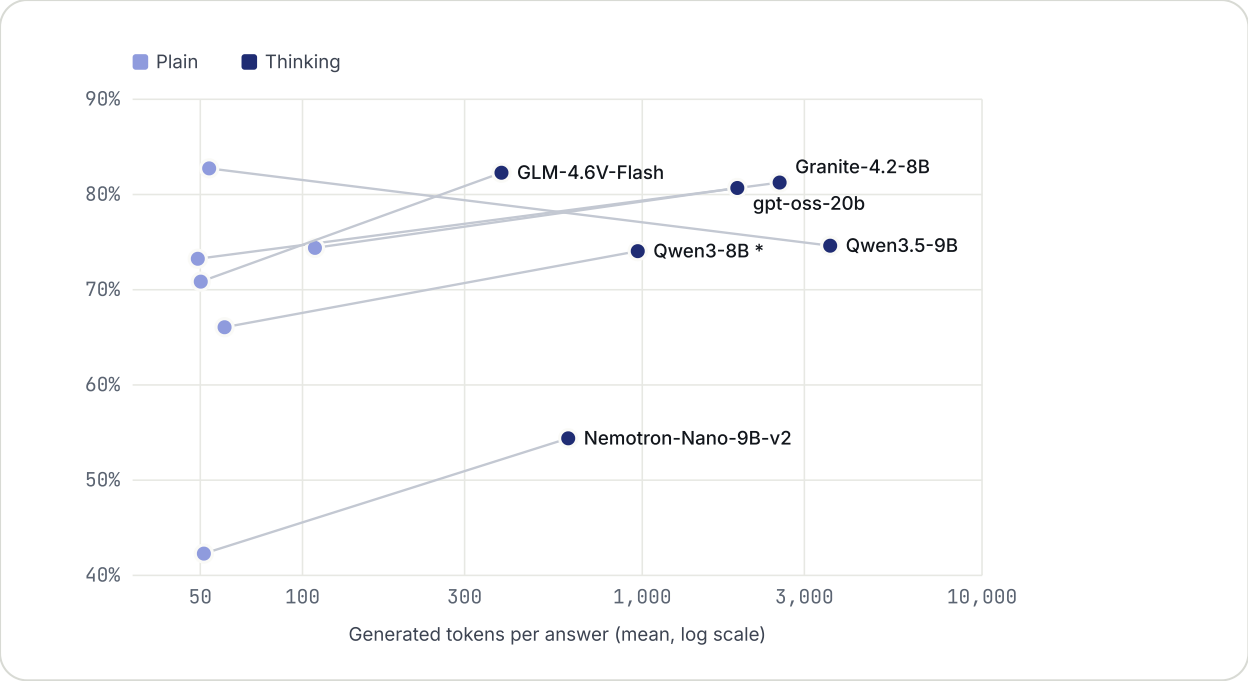


Figure 12. The cost of thinking: mean generated tokens per answer (log scale) against accuracy. * supplementary baseline.

Thinking is expensive. It multiplies generated tokens by 67× (Qwen3.5-9B), 52× (Granite-4.2-8B), 12× (Nemotron-Nano-9B-v2), 18× (gpt-oss-20b), 8× (GLM-4.6V-Flash). On one RTX 4090 with 16 concurrent requests, a full 1,000-question run took 14 minutes for GLM-4.6V-Flash with thinking and 182 for Granite-4.2-8B, against 3–7 minutes in the plain setup.

7 · Discussion

Checkable evidence is necessary, not sufficient. Requiring a verbatim quote turned most errors into visible, rejectable ones: the checker is cheap, needs no model and no gold answer, and could run in any system that uses these models. What it cannot do is check that the number is the *right* figure on the line, or that its units are right. The residue is small for the best configurations (1.6% of answers at best) and substantial for most (up to 16.7%).

The residue is structured. 87% of confident-wrong answers are the right line read wrongly: the units header misapplied, or the neighbouring column taken. Both are checkable in principle. A stronger checker could require the model to quote the column header and the units statement as well as the line, and verify that the chosen figure sits under the quoted header; a units check could compare the stated scale with the table's own heading. These are engineering steps, not research problems, and they are the natural next version of the benchmark.

Retrying is not a free win. Feeding a model the name of the failed check improves accuracy a little, but it also turns some caught errors into uncaught ones. A system that retries should treat a passed retry with more suspicion than a passed first answer, or re-verify it more strictly.

Model choice matters more than setup. The spread between the best and worst model in the plain setup is 66 points; the largest gain from any setup is about 11. And the strongest model in the plain setup, Qwen3.5-9B, reaches 84.5% with no reasoning at all.

8 · Limitations

- **One prompt, temperature 0, one run.** Results are for a single fixed prompt shared by all models; a prompt tuned per model could change rankings. Each configuration was run once.
- **US large-cap filings, text only.** 45 companies, recent 10-K and 10-Q filings. No small caps, foreign filers, scanned documents, charts or images.
- **Three pages per question.** The model is shown the right page plus two others; finding the page in a whole filing is a harder, separate task.
- **Units stated off-page.** After the results we found 30 answerable questions (3.4%) whose page shows no units statement, because the filer states it once on an earlier page. The frozen dataset was not changed. Excluding them moves no model's accuracy by more than 2.3 points. The plain and check-and-retry rankings do not change; with thinking, Granite-4.2-8B moves ahead of GLM-4.6V-Flash (Appendix A).
- **Quality control was AI-assisted.** The review of the 100-question samples was performed by an AI assistant under the author's direction. The author did not independently verify the full sample.
- **Error categories were not reviewed by hand.** The protocol planned a hand review of 20 answers per model to confirm the automatic error categories; it was not carried out (recorded as a deviation). Outcomes themselves are exact, by construction; the categories are rule-based.
- **Thinking budget.** The 8,192-token reasoning limit was fixed in advance; it truncated 180 of Qwen3.5-9B's and 97 of gpt-oss-20b's answers. A larger budget would likely raise their thinking-mode results.
- **Contamination.** Models may have seen older filings in training. 89 of the 90 filings were filed in 2026, and eight of the ten models were released before 2026. A memorised answer must still come with a verbatim quote from the pages shown.
- **Precision.** Phi-4 and Gemma-3-12B were run with FP8 weights to fit in 24 GB, and gpt-oss-20b in its native MXFP4; results at full precision may differ slightly.

9 · Reproducibility

Everything needed to check or repeat this study is public. The code, the protocol (with its deviations log) and each run's summary are on GitHub (https://github.com/pramana-research/filing-evidence); the questions, page texts, every model reply including its reasoning, and the server logs are on Hugging Face (https://huggingface.co/datasets/pramana-research/filing-evidence). Each run records the model revision, the exact vLLM command, the prompt hash and the dataset fingerprint.

```
git clone https://github.com/pramana-research/filing-evidence.git && cd
filing-evidence
uv sync --all-extras
```

```
uv run python hf_data.py pull                          # data, verified
against the frozen fingerprint
uv run python runner.py qwen3.5-9b C1 --regrade        # re-grade stored
replies (no GPU)
bash run_all.sh qwen3.5-9b                             # rerun a model (one 24
GB GPU)
```

Re-grading every stored reply reproduces every number on this page. These pages are generated from the stored results by `report/build.py` in the same repository.

Results per model

Each model has its own page with its scorecard, its profile by kind of content, what retry and thinking changed, and real examples of its correct and wrong answers.

Qwen3.5-9B
Alibaba

Plain 82.7% Best 84.5%

gpt-oss-20b
OpenAI

Plain 74.4% Best 80.7%

Granite-4.2-8B
IBM

Plain 73.3% Best 81.3%

GLM-4.6V-Flash
Zhipu (Z.ai)

Plain 70.9% Best 82.3%

Ministral-3-8B
Mistral

Plain 66.7% Best 69.6%

Gemma-3-12B
Google

Plain 61.4% Best 64.3%

Llama-3.1-8B
Meta

Plain 58.9% Best 62.7%

Phi-4
Microsoft

Plain 57.3% Best 60.8%

Nemotron-Nano-9B-v2
NVIDIA

Plain 42.3% Best 54.4%

OLMo-3-7B
AllenAI

Plain 17.1% Best 17.3%

Qwen3-8B
Alibaba · supplementary

Plain 66.1% Best 74.1%

References

1. U.S. Securities and Exchange Commission. EDGAR full-text and submissions APIs. sec.gov/edgar (https://www.sec.gov/edgar).
2. XBRL International. Inline XBRL Specification 1.1. xbrl.org (https://www.xbrl.org/specification/inlinexbrl-part1/rec-2013-11-18/inlinexbrl-part1-rec-2013-11-18.html).
3. E. B. Wilson (1927). Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association* 22(158), 209–212.
4. Q. McNemar (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* 12(2), 153–157.
5. S. Holm (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics* 6(2), 65–70.
6. W. Kwon et al. (2023). Efficient memory management for large language model serving with PagedAttention. *SOSP '23*. (vLLM)
7. Model cards of the eleven models evaluated, on Hugging Face, at the pinned revisions listed on each model page.

Appendix

A · Sensitivity: questions without an on-page units statement

Model	C1 all 875	C1 without 30	C2 all 875	C2 without 30	C3 all 875	C3 without 30
Qwen3.5-9B	82.7%	82.7%	84.5%	84.5%	74.6%	76.1%
gpt-oss-20b	74.4%	76.0%	77.4%	79.1%	80.7%	82.4%
Granite-4.2-8B	73.3%	74.4%	76.2%	77.3%	81.3%	83.6%
GLM-4.6V-Flash	70.9%	72.3%	73.0%	74.6%	82.3%	83.0%
Ministral-3-8B	66.7%	68.8%	69.6%	71.7%	n/a	n/a
Gemma-3-12B	61.4%	63.4%	64.3%	66.5%	n/a	n/a
Llama-3.1-8B	58.9%	60.4%	62.7%	64.4%	n/a	n/a
Phi-4	57.3%	58.9%	60.8%	62.5%	n/a	n/a
Nemotron-Nano-9B-v2	42.3%	43.8%	42.4%	43.9%	54.4%	56.0%
OLMo-3-7B	17.1%	17.4%	17.3%	17.5%	n/a	n/a

B · Within-model comparisons

Model	Comparison	Accuracy: only A / only B	Holm p	Conf.-wrong: only A / only B	Holm p
Qwen3.5-9B	C1 vs C2	0 / 15	**0.000427**	0 / 0	1
Qwen3.5-9B	C1 vs C3	159 / 88	**7.35e-05**	27 / 7	**0.0107**
Qwen3.5-9B	C2 vs C3	162 / 76	**4.35e-07**	27 / 7	**0.0107**
Granite-4.2-8B	C1 vs C2	0 / 26	**4.77e-07**	0 / 8	0.0579
Granite-4.2-8B	C1 vs C3	91 / 161	**0.00011**	51 / 25	**0.0422**
Granite-4.2-8B	C2 vs C3	96 / 140	**0.0201**	56 / 22	**0.00209**
Ministral-3-8B	C1 vs C2	0 / 25	**7.75e-07**	0 / 9	**0.0422**
OLMo-3-7B	C1 vs C2	0 / 1	1	0 / 0	1
Nemotron-Nano-9B-v2	C1 vs C2	0 / 1	1	0 / 0	1
Nemotron-Nano-9B-v2	C1 vs C3	100 / 206	**2.68e-08**	86 / 53	0.0579
Nemotron-Nano-9B-v2	C2 vs C3	101 / 206	**3.91e-08**	86 / 53	0.0579
gpt-oss-20b	C1 vs C2	0 / 26	**4.77e-07**	0 / 6	0.156
gpt-oss-20b	C1 vs C3	67 / 122	**0.000462**	76 / 20	**1.32e-07**
gpt-oss-20b	C2 vs C3	74 / 103	0.105	79 / 17	**2.09e-09**
Phi-4	C1 vs C2	0 / 31	**1.96e-08**	0 / 16	**0.000458**
Gemma-3-12B	C1 vs C2	0 / 26	**4.77e-07**	0 / 17	**0.000244**
GLM-4.6V-Flash	C1 vs C2	0 / 19	**4.2e-05**	0 / 1	1
GLM-4.6V-Flash	C1 vs C3	49 / 149	**1.45e-11**	59 / 10	**2.62e-08**
GLM-4.6V-Flash	C2 vs C3	52 / 133	**4.08e-08**	60 / 10	**1.6e-08**
Llama-3.1-8B	C1 vs C2	0 / 34	**2.56e-09**	0 / 22	**8.11e-06**
Qwen3-8B	C1 vs C2	0 / 11	**0.00488**	0 / 7	0.0938
Qwen3-8B	C1 vs C3	66 / 136	**1.14e-05**	67 / 10	**3.73e-10**
Qwen3-8B	C2 vs C3	70 / 129	**0.000278**	73 / 9	**3.17e-12**

"Only A": questions right (or, for confident-wrong, wrong-but-passing) in the first setup only. Blue p-values are significant at α = 0.05 after Holm correction within each family.

C · Confident-wrong rate

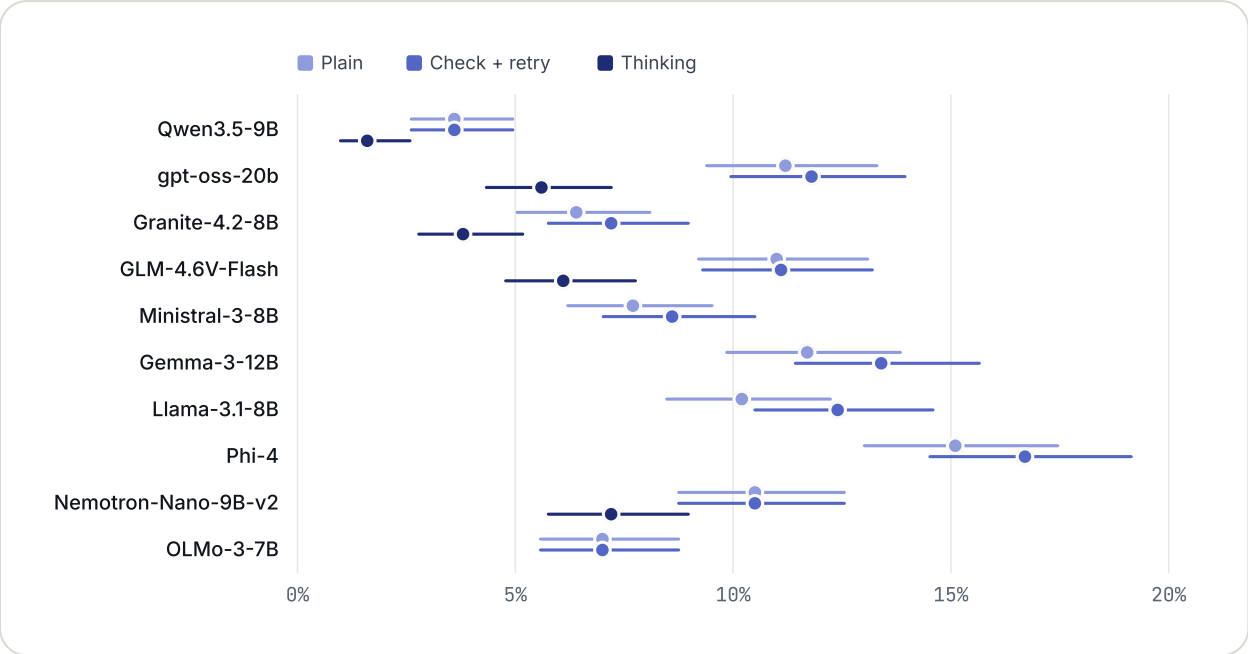


Figure 13. Confident-wrong rate with 95% Wilson intervals, per model and setup.

D · Deviations from the protocol

All changes made after the protocol was registered are logged with their date and reason in PROTOCOL.md §12 (https://github.com/pramana-research/filing-evidence/blob/main/PROTOCOL.md). The ones that affect how results are read: the served context is 16,384 tokens (8,192 would have truncated thinking); C2 reuses C1's first answer; Qwen3-8B is a supplementary baseline; gpt-oss-20b's plain setup uses its lowest reasoning effort; a secondary accuracy-ignoring-scale metric was added; the error taxonomy counts a misplaced decimal point as a scale error; and the planned hand review of error categories was not carried out. All except the last were made before any main result.

Disclaimer

This is an independent community research report, written in a personal capacity; it does not represent the views of any employer or other organisation. It is not affiliated with, sponsored or endorsed by any company or model developer named here; all names and trademarks belong to their owners. Results describe specific model versions under one fixed setup and are not a general judgement of any model. The data, results and code are provided as is, without warranty, and may contain errors; check any figure against the original filing. Nothing here is financial, investment or legal advice.

How to cite

```
@misc{pramana2026filingevidence,
  title         = {Show Your Work: Can Small Open Models Read SEC Filings with
```

```
Evidence a Machine Can Check?},
  author        = {Nandula, Krishna},
  organization  = {Pramana Research},
  year          = {2026},
  month         = {nov},
  eprint        = {2611.24990},
  archivePrefix = {arXiv},
  primaryClass  = {cs.SE},
  doi           = {10.48550/arXiv.2611.24990},
  url           = {https://arxiv.org/abs/2611.24990},
  note          = {Community research report, not peer-reviewed. Filing-
Evidence benchmark, report version 1.1}
}
```

This is a community research report; it has not been peer-reviewed. Text, data and figures are licensed under CC BY 4.0 and the code under the Apache License 2.0. Model outputs remain subject to each model's licence. Nothing here is investment advice.

© 2026 Pramana Research · Text, data and figures: CC BY 4.0 · Code: Apache 2.0

Model outputs are not investment advice. Check every figure against its source.