Can Frontier LLMs Read UK Company Accounts? I Tested Five — Then Audited My Own Benchmark

Updated July 2026. This is a substantial revision. The original version of this article reported two model failure modes — inventing £0 turnover and sign errors on net liabilities — that my own subsequent audits falsified. Both turned out to be faults in my ground truth, not in the models. The correction changes the headline finding and, I think, makes it more interesting. The original claims and their retirement are documented in the benchmark's public honesty notes.

UK company accounts are genuinely awkward for a machine. Most small companies file *filleted* accounts that legally leave out the profit-and-loss account, so there's no turnover to read. Micro-entities file a balance sheet barely longer than a receipt. Figures come in £'000 that you have to scale up, negatives hide inside parentheses, two years sit side by side, and the whole thing uses British accounting vocabulary that isn't the American vocabulary these models mostly learned on.

So I expected large language models to struggle. I built a benchmark to find out, tested five frontier models — the proprietary big three and two open-weight models you can self-host — and then did something most benchmark authors don't: I audited my own answer key, three times, using the models themselves as the error detector. That audit is the real story.

The test

- **1,000 "clean" questions** — extraction, comparison, ratios, and disclosure reasoning over structured UK financials, every answer computed from source data.
- **350 "hard" questions** — read the figure straight out of the *raw* filing (personal data removed), no hints, plus a trap: ask for turnover on a filleted company that doesn't disclose it, and see whether the model invents a number.
- **A contamination control** — a second 350-question hard set built *only* from filings that Companies House published after every tested model was released. Text published after a model's release cannot be in its training data, so this slice answers the memorisation objection outright.

The results

Clean data:

Model	Clean (n=1,000)
Claude Opus 4.8	99.6%
DeepSeek V4 Pro *(open)*	99.5%
GPT-5.5	98.4%
GLM 5.2 *(open)*	98.4%
Gemini 2.5 Pro	98.1%

Raw filings — as first published in June, and after the full ground-truth audit:

Model	June, as published	June, audited (n=331)	Unseen filings (n=348)
GPT-5.5	97.9%	99.7%	100.0%
Claude Opus 4.8	97.3%	99.1%	100.0%

Model	June, as published	June, audited (n=331)	Unseen filings (n=348)
Gemini 2.5 Pro	96.7%	98.5%	99.4%
DeepSeek V4 Pro *(open)*	96.7%	98.5%	99.4%
GLM 5.2 *(open)*	95.8%	97.6%	100.0%

Every difference between the published and audited columns is a fault that was mine, not the models'. And note the third column: on filings the models provably never saw in training, they score higher still. Memorisation does not explain any of this.

The other result that matters: **the open, self-hostable models match the proprietary leaders.** For finance teams whose data-governance rules forbid sending client data to a third-party API, DeepSeek or GLM on your own hardware reads UK accounts just as reliably.

The audit that changed the story

The method is simple and I now think every benchmark should use it: when four or five independent models give the *same* "wrong" answer, the probability they are all wrong the same way is far lower than the probability that your answer key is wrong. Treat unanimous disagreement as an alarm on the ground truth, then adjudicate against the source document.

Run three times across three evaluation rounds, that alarm found nineteen faults in my June answer key and seven more across two later rounds — and every single one was mine. The models caught, among other things:

- a parser bug that read net liabilities as positive when the filing put the parentheses outside the tagged number;
- a fallback that silently recorded *last year's* figure when the current year showed "—";
- trap questions asserting turnover was "not disclosed" in filings that printed it — one of them three times, in a full profit-and-loss account;
- contexts where a filing's raw CSS had buried the balance-sheet figure the question asked about.

Two of those bugs were live in my data pipeline. The benchmark's most valuable output was debugging the product it was built to justify.

And the trap? Across roughly **970 valid trap trials over five models and three rounds, not one model ever invented a turnover figure for a filleted company.** Every recorded trap "failure" was a dormant company — where £0 is defensible — or my own artifact. The scare story I originally led with is dead, and I killed it with my own data.

What errors actually remain

The models' genuine residual errors — a handful per model per 350 questions — are quieter and, for financial use, worse than hallucination folklore suggests, because they are plausible and silent:

- **Wrong-line grabs under adversarial layouts** — the prior-year column when the current year is nil; a note whose column order flips against the balance sheet; the *group* figure when the question asks about the *company* in consolidated accounts.
- **Occasional under-extraction** — "not disclosed" for a figure that is present.
- **Digit-level slips** — 290 read as 390.

No warning, no error bar, no reproducibility: run it again and the mistake moves somewhere else. Fine for a demo. Disqualifying when a number feeds a lending decision — unless something checks it.

Why UK accounts trip up assumptions: the UK-GAAP vs US-GAAP gap

If your mental model of company accounts is American, UK filings quietly break it:

- **Different words for the same things.** UK "turnover" is US "revenue"; "debtors" are "receivables"; "creditors" are "payables"; "net assets" / "shareholders' funds" is "total equity"; "stocks" are "inventory".
- **Filleted accounts.** Under the small-companies regime, a company can file just the balance sheet and notes — no profit-and-loss account at all. Ask for turnover and the correct answer is often "not disclosed", not a number.
- **Micro-entities (FRS 105).** An even more stripped-down regime — a tiny balance sheet, almost no notes. Most of the register looks like this.
- **Presentation quirks.** Figures in £'000 that must be multiplied up; negatives shown as "(13,597)" — sometimes with the brackets *outside* the tagged number; and current-plus-prior columns where picking the wrong one gives a plausible wrong answer. My parser fell for two of these. So, occasionally, do the models.

The real lesson

I started this project believing the pitch was "models make dangerous errors, so buy verified data." The audit forced a better conclusion. The models read superbly — better than my parser on individual documents, as it turned out. But their rare errors are silent and non-reproducible, per-figure provenance does not exist at any accuracy, and a full pass over 3.7 million filings via API costs five to six figures against roughly nothing for a pipeline.

The durable asset is the **verification loop**: deterministic extraction and model reading, cross-checked, with every disagreement adjudicated against the source and every correction dated and published. Each side catches the errors the other cannot see — the audit above is that loop running in public, on my own benchmark, at my own expense. That loop, applied at scale to 3.7 million messy, filleted, multi-taxonomy filings, is what produces clean, verified, provenance-tracked data. That's what I built, and the week I spent proving my models right and my answer key wrong is the best evidence I have that "verified" means something.

Try it / use it

- **The benchmark** (questions, answers, evaluation harness, and the full correction history — run it against your own model): github.com/dacheah/uk-accounts-llm-benchmark
- **The pipeline** (source-available): github.com/dacheah/uk-accounts-pipeline
- **The datasets** — UK company financials, and a structured-finance SPV/lender map — are on Hugging Face (training-data licences, refreshed feeds, and bespoke cuts). Enquiries: **contact@danielcheah.com**.

— **Daniel Cheah** · danielcheah.com · LinkedIn