Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

# Format failure hides inside accuracy: the GGUF quantization level that breaks is not predictable from its label

**Zhang Shaoxuan**

2023113033@stu.hit.edu.cn

Harbin Institute of Technology https://orcid.org/0009-0001-4103-6099

**Research Article**

**Keywords:** quantization, calibration, uncertainty, GGUF, small language models, empirical evaluation

**Posted Date:** November 12th, 2026

**DOI:** https://doi.org/10.21203/rs.3.rs-11704703/v1

**License:** This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

**Additional Declarations:** The authors declare no competing interests.

# Format failure hides inside accuracy: the GGUF quantization level that breaks is not predictable from its label

**Shaoxuan Zhang**

Faculty of Computing, Harbin Institute of Technology, Harbin, Heilongjiang, China

Corresponding author: 2023113033@stu.hit.edu.cn

## Abstract

Accuracy benchmarks score a non-answer and a wrong answer alike. We test whether that convention survives the 8-bit to 4-bit step of GGUF quantization. Across four instruction-tuned models from 0.5B to 3B, four tasks at 800 items each, and every published precision level, 70,394 paired responses separate the responses that answer from the responses that do not. Format failure is concentrated and not monotone in bit-width. On ARC-Challenge, Qwen2.5-0.5B produces no usable answer on 1.6% of responses at Q8_0, 5.4% at Q5_K_M, 19.4% at Q4_K_M, 9.1% at Q3_K_M and 10.4% at Q2_K: the level that fails is Q4_K_M, while the more compressed levels answer more often. Qwen2.5-3B fails at Q3_K_M instead, at 30.6% against 3.1% at Q4_K_M, and Gemma-3-1B and Llama-3.2-3B never exceed 0.1% at any level. Counting non-answers as errors raises the measured cost of the 4-bit step at 0.5B from 9.0% to 10.9%; on responses that do answer, the cost falls with scale, 9.0%, 6.8% and 3.1% against a 0.4% gain at Llama-3.2-3B. A label does not tell a practitioner whether the artifact will answer, and the level that fails cannot be inferred from its neighbours.

**Keywords:** quantization, calibration, uncertainty, GGUF, small language models, empirical evaluation

## 1. Introduction

Accuracy is the number a deployment decision is made on, and it is computed by counting correct answers. A response that echoes the instruction's placeholder, or that names a news category outside the permitted four, is scored wrong in exactly the same way as a response that picks the wrong option. That convention is reasonable for a model that always answers, and it is what makes an accuracy benchmark cheap to run. It stops being reasonable when quantization changes not only which answer a model gives but whether it gives one.

Quantization is the default when a language model has to fit a memory budget, and the practice is common enough that most consumer inference stacks ship a 4-bit artifact by default. The label on that artifact is treated as a specification: Q4_K_M is expected to cost about the same accuracy wherever it is applied. We find that it does not, and that part of the reason is a failure the accuracy metric cannot name.

That assumption rests on a particular body of evidence. The largest study of the trade-off sweeps 19M to 176B parameters and concludes that 4-bit is nearly universally optimal for a fixed memory budget, and that the bit-level scaling curves are almost parallel across model size, so that lowering precision costs about the same regardless of scale [1]. The models in that study are base language models, quantized with a scheme built for the experiment, and evaluated as perplexity and zero-shot accuracy.

What is deployed is a different object. Consumer and edge deployments use instruction-tuned models of a few billion parameters or fewer, quantized to the GGUF k-quant format, which is post-hoc, needs no calibration data, and allocates different precision to different tensors. Whether a trade-off mapped under one set of conditions carries to the other is an empirical question. It is also a question with a direct practical consequence: if the cost of the 4-bit step grows as models shrink, then the standard precision choice is least safe exactly where it is most used.

We ask two questions. Does the cost of a given nominal level hold across models, or does it vary with the model it is applied to? And when a quantized model loses accuracy, is the loss one of knowledge, or partly a failure to answer at all? The second matters because the two have different remedies. A model that knows less needs a different precision level; a model that has stopped following the output format needs a different prompt or a different artifact. An accuracy score that counts both as errors points at neither.

We measure the 8-bit to 4-bit step, and the intermediate levels, across four instruction-tuned models from 0.5B to 3B parameters, quantized as GGUF k-quants, on four tasks at 800 items each, for 70,394 scored responses. Every configuration sees the identical seeded item sample, so all contrasts are paired at the item level. Three protocol choices are load-bearing. The item sample is fixed across configurations, which removes item-difficulty variance from the contrast. Token probabilities are read only over the span of the answer value, because schema-constrained decoding forces the surrounding structural tokens and contaminates any average taken over them. And the run-to-run disagreement of the pipeline is measured rather than assumed to be zero, so every claimed effect can be compared against it.

Three results follow.

The same nominal level costs different amounts on different models. Relative to each model's own 8-bit accuracy, Q4_K_M costs 10.9% at 0.5B, 7.9% at 1B, 4.0% at Qwen2.5-3B and a negligible 0.3% at Llama-3.2-3B. A difference-in-differences contrast on matched items gives −0.026 against 1B and −0.038 against 3B when the 3B tier enters as the mean of both its models, both $p \leq 0.0004$, though not against Qwen2.5-3B alone ($p=0.19$). The spread does trend with size, but the trend rests on the smallest model, and that is the model whose Q4_K_M artifact is anomalous: it scores 3.0 points below the coarser Q3_K_M on matched items, and replacing the level with Q3_K_M reverses the ordering. The contrast between the two sub-2B sizes is not resolved at this sample size.

The memory saved and the accuracy lost are badly matched across the ladder. At 0.5B, dropping from 8-bit to 5-bit removes 106 MB and costs 1.0 accuracy point, an interval that spans zero; dropping from 5-bit to 4-bit removes 21 MB and costs 4.2 points at $p<0.001$. The same ordering holds at 3B, where 1,012 MB costs 0.3 points and 281 MB costs 2.7. The step that saves the least memory is the one that costs the most accuracy.

Stated confidence degrades on a different axis than accuracy. At 0.5B the 4-bit step raises stated confidence from 0.86 to 0.92 while accuracy falls: the model becomes more confident as it becomes less correct, which is the failure an accuracy-only acceptance test cannot see. At 1B and 3B stated confidence is flat or lower, and the calibration cost there is largely a restatement of the accuracy cost. We had reasoned that reduced precision might damage a model's self-assessment before its correctness. That

ordering is not present in our data, and the over-confidence we do find arrives on the same ladder step as the accuracy loss.

We make two main contributions and two supporting ones.

First, a paired, item-matched measurement of format failure across 22 model-level cells. It is concentrated in two of the four models, the level at which it occurs differs between them, and it is not monotone in bit-width, so a failure at one level carries no information about its neighbours.

Second, a decomposition of the measured quantization cost into a format component and a capability component. At 0.5B this changes the headline from 10.9% to 9.0%.

Our third and fourth contributions are supporting. The level inversion we report is not an artifact of one download, since a file from a second publisher reproduces it. And the calibration cost decomposes into what a model says about its confidence and what its token probabilities imply, measured under a paired, noise-floor-referenced protocol that also produced an account of three scoring defects we found and corrected.

---

---

## 2. Related work

### 2.1 Bit-width, scale, and the parallel-scaling assumption

The relationship between parameter count, bit-width, and accuracy has been mapped at scale. Dettmers and Zettlemoyer [1] ran over 35,000 zero-shot experiments across four model families and 19M to 176B parameters, sweeping 3 to 16 bits, and concluded that 4-bit precision is almost universally optimal for a fixed memory budget. The result that matters for the present study is their observation that the bit-level scaling curves are *almost parallel* across model sizes: lowering precision costs approximately the same amount regardless of how large the model is. Their recommendation follows directly, that 4-bit with a small block size and a float data type should be the default.

That conclusion was established under two conditions that differ from the setting we study. Quantization there is applied as a zero-shot scheme to base language models, with a quantization implementation constructed for the experiment, and performance is read out as perplexity plus zero-shot accuracy on four benchmark tasks. The format that practitioners on consumer hardware deploy is a different object: GGUF k-quants are post-hoc, require no calibration data, and are *mixed-precision by construction*, assigning different numbers of bits to different tensors. Instruction-tuned models are also what is deployed, and their response distributions differ from base models on exactly the tasks where a small model is likely to be used.

The parallel-scaling assumption has since been formalized at the level of task structure. Zhou et al. [3] fit task-stratified scaling laws for post-training quantization, using GPTQ on the OPT family across six sizes from 125M to 13B in 384 configurations, varying effective bit-width, group size, and calibration-set size, and stratifying performance into knowledge memorization and knowledge utilization. This is the closest prior work to ours, and two differences in design matter.

The first is structural. Their law is a multiplicative power function of the form

$$Acc \approx C \cdot N^{\alpha} \cdot [\log_2 C_b]^{\beta} \cdot G^{\gamma} \cdot [\log_2 B_e\mathrm{ff}]^{\delta},$$

which is separable by construction: it contains no model-size × bit-width cross term. Under this form an interaction between scale and precision is not estimated; it can only degrade the goodness of fit, which they report as an adjusted $R^2$ of about 0.83. Their framework therefore assumes the separability that the parallel-scaling description predicts rather than testing it. Our design tests the interaction directly, with a difference-in-differences contrast on matched items and a reported interval.

The second is the quantization regime. Their quantization is calibrated, optimizer-based PTQ, in which the calibration set and the group size are tuning parameters and the effective bit-width accounts for metadata overhead. GGUF k-quants use no calibration data and are mixed-precision across tensors. Whether a separable law fitted under calibrated PTQ transfers to the uncalibrated consumer format is an empirical question, not an implication.

Neither line of work measures uncertainty. The "calibration set size" of Zhou et al. refers to the data used to estimate quantization parameters, which is a different quantity from the calibration of a model's predictive confidence.

We treat parallel scaling as the hypothesis under test. What we measure is how far the "almost parallel" description holds in the regime where the 4-bit default is actually applied, and how much of any deviation is attributable to differing headroom.

### 2.2 Calibration under reduced precision

Reduced precision is known to damage a model's uncertainty estimates. Zhong et al. [4] derive an analytic upper bound on calibration error for quantized models, separate the error contributed by correct and incorrect predictions, and show across several models and datasets that quantized models are consistently worse calibrated than their full-precision counterparts. They go further and propose a soft-prompt post-hoc correction that recovers much of the lost calibration.

This establishes the existence and the direction of the effect, and offers a repair. It does not establish how the magnitude varies with model scale, which is the quantity that determines whether a given deployment can absorb the loss or must budget for the correction. A practitioner choosing between a 0.5B model at 4-bit and a 3B model at 4-bit is not asking whether quantization damages calibration but by how much, at each scale, relative to the accuracy they are also giving up.

Two further results bear directly on our second question, and we state their relation to ours plainly.

Tamiosso et al. [7] benchmark small language models from 135M to 20B parameters for English-to-Portuguese translation at FP16, Q8_0 and Q4_K_M, and report that 4-bit quantization reaches statistical significance in roughly half of their configurations with effect sizes that are negligible to small, with measurable degradation concentrated in lower-capacity models. That is the closest match to our accuracy result, and it uses the same artifact format. Two features of their design bound what it can show. The precision grid is three points — FP16, Q8_0 and Q4_K_M — so a step such as 5-bit to 4-bit, which is where our own cost concentrates, cannot be observed there at all; our six-level ladder is what makes that comparison possible. And the evaluation is translation only, scored with BLEU, chrF and BERTScore, so

it speaks to lexical and semantic fidelity rather than to knowledge or multi-step reasoning. Our accuracy result is therefore a replication across task families together with a finer ladder.

Bouguerra et al. [6] evaluate quantized vision-language models across more than 700,000 runs and find that reduced precision can improve accuracy, calibration and out-of-distribution detection, attributing the gain to quantization damping high-rank spectral components. Their result and ours point in opposite directions. The difference is one of regime: their setting is a vision-language model assessed on distribution-shift robustness, while ours is a text-only instruction-tuned model assessed on knowledge and reasoning tasks, where the calibration loss we observe grows as models shrink. Section 5.3 returns to this.

Neither line measures calibration for instruction-tuned language models in the consumer quantization format, which is the axis this study adds.

### 2.3 The gap this study occupies

Four things are established: that 4-bit is close to optimal in a memory-normalized sense and that scaling is almost parallel across sizes; that quantization sensitivity depends on task type; that in at least one small-model translation setting the 4-bit degradation concentrates in lower-capacity models; and that quantized models are worse calibrated, with a counterexample in quantized vision-language models where the opposite holds. What is not established is whether the cost of a given nominal level belongs to the label or to the model it is applied to, where on the ladder the cost concentrates, and whether the calibration cost follows the accuracy cost.

We report a paired, item-matched measurement of the 8-bit to 4-bit step across four instruction-tuned models spanning 0.5B to 3B in the GGUF k-quant format, on both accuracy and calibration. The degradation is 10.9% at 0.5B, 7.9% at 1B, and 1.9% at 3B pooled; the interaction is significant at $p = 0.0004$ against 1B and $p = 0.0002$ against 3B; and the calibration cost traces the same gradient. Where our data contradict a hypothesis we advanced, that 4-bit would degrade calibration before accuracy, we report the contradiction (Section 4.3).

---

---

## 3. Materials and methods

### 3.1 Study design

We ask how the cost of reduced numerical precision depends on model scale. The study is a full factorial design over three factors: model family and size, quantization level, and task. Every configuration is evaluated on an identical, seeded sample of items, so all contrasts are paired.

Parameter count is an experimental factor. The primary contrast is within-family: Qwen2.5 at 0.5B and 3B, a six-fold increase holding architecture and training recipe fixed. Additional families at intermediate sizes provide a check that the size effect is not an artefact of one lineage.

Two outcomes are measured throughout: task accuracy, and the calibration of the model's stated confidence. We test whether the accuracy cost of quantization grows as models shrink, and whether calibration degrades earlier than accuracy does.

Four protocol commitments close specific routes by which an evaluation of this kind can produce a confident but wrong answer. The item sample is fixed across configurations, making comparisons paired. The context window is pinned, since leaving it unset lets the serving stack choose a different window between runs and silently truncate prompts. Token probabilities are read only over the span of the answer value, because schema-constrained decoding forces the surrounding structural tokens. Run-to-run disagreement is measured rather than assumed zero, and every effect is judged against it.

A fifth commitment emerged while building the pipeline and is reported in §3.13: every scoring rule is verified against the raw text before it is trusted, and raw outputs are retained so that any later correction can be applied without re-running the model.

### 3.2 Models

We evaluate instruction-tuned models spanning roughly 0.5B to 3B parameters: Qwen2.5-0.5B-Instruct, Qwen2.5-3B-Instruct, Llama-3.2-3B-Instruct, and Gemma-3-1B-IT. This is the range in which consumer hardware forces a precision decision, and where quantization damage is large enough to measure.

All inference is local, via Ollama 0.40.6, which exposes token log-probabilities through its generation API. The runtime is the binary distributed with that Ollama release.

### 3.3 Quantization levels

Models are obtained as GGUF k-quants in six configurations: Q8_0, Q6_K, Q5_K_M, Q4_K_M, Q3_K_M, Q2_K. Q4_K_M is the reference, being the default shipped for most models in this ecosystem.

GGUF k-quants are used rather than GPTQ, AWQ, or bitsandbytes NF4 because they are what practitioners on consumer hardware actually run, and because they require no per-model calibration. The consequence is that conclusions are specific to this quantization family: weight-only 4-bit formats on datacenter accelerators use different rounding and grouping, and the effects reported here should not be assumed to transfer without re-measurement.

All GGUF artifacts were downloaded from the Hugging Face mirror of the bartowski repository for each model: `Qwen2.5-0.5B-Instruct-GGUF`, `Qwen2.5-3B-Instruct-GGUF`, `Llama-3.2-3B-Instruct-GGUF` and `google_gemma-3-1b-it-GGUF`. Filenames follow `<model prefix>-<quant>.gguf`. The byte count of every artifact is recorded in `figures/source_data/artifact_sizes_mb.json`, read from the server's Content-Length at download time and matching the files on disk; those byte counts are also the memory axis of Fig. 2c. Llama-3.2-3B has no Q3_K_M or Q2_K artifact published in that repository, which is why that model stops at four levels.

### 3.4 Tasks

Four tasks span distinct capability profiles.

*GSM8K* (1,319 test items): grade-school word problems with a single numeric answer, extracted from the reference solution after the `####` delimiter. Multi-step arithmetic with a free-form response.

*ARC-Challenge* (1,172 test items): science multiple-choice with three to five options.

*AG News* (7,600 test items): four-way topic classification of news openings.

*MMLU* (14,042 test items, 57 subjects): four-option multiple-choice across academic subjects.

We draw 800 items per task for the main experiment, 200 for the pilot, 20 for pipeline verification. The cap is fixed a priori.

GSM8K is from[8], ARC-Challenge from[9], AG News from[10] and MMLU from[11]. All four were read from the test split of their Hugging Face dataset mirrors `openai/gsm8k`, `allenai/ai2_arc`, `fancyzhx/ag_news` and `cais/mmlu`. The AG News artifact is identified by its SHA-256 hash in Section 3.7.

### 3.5 Sampling and pairing

Within each task the sample is drawn once with a generator seeded deterministically from the task name, and that sample is reused for every model and quantization level. The design is therefore paired at item level: item $i$ is the same question in every cell. Configuration contrasts are computed within items, which removes the large between-item variance in difficulty. Given that the effect of a quantization level is small relative to that spread, pairing is what makes the effect detectable.

### 3.6 Output protocol

The output protocol is chosen per task, because a single protocol cannot serve both closed-set and reasoning tasks.

Closed-set tasks (ARC-Challenge, AG News, MMLU) use schema-constrained JSON with keys `answer` and `confidence`. Constrained decoding guarantees a parseable object, and for a field whose permitted values form a small closed set the resulting distribution over those values is a meaningful categorical distribution.

GSM8K does not use schema constraint in the main configuration. Constraining a model to emit JSON from the first token leaves no room for a scratchpad, and measurement showed the consequence directly: the same Qwen2.5-3B model scores approximately 0.13 on GSM8K under schema constraint against approximately 0.70 when permitted to reason. The 0.13 measures arithmetic intuition without working memory, not mathematical reasoning. GSM8K therefore generates freely and reports its answer in prose.

We also evaluated a third option for GSM8K, a schema containing a `reasoning` string field. It restored parseability but degraded generation: at 0.5B the model produced summaries of intended arithmetic rather than the arithmetic itself, accuracy fell to 0.00 on a 40-item check, the `answer` field contained expressions such as `"$2500 * 0.30 = $750"`, and stated confidence was pinned at 100 with no variance. That option was rejected on those measurements.

### 3.7 Answer extraction and scoring

Numeric answers are normalized by removing separators and currency symbols, parsed as floating point, and compared after integer casting where integral. Multiple-choice answers are reduced to the first option letter. A response that cannot be parsed is counted as incorrect, so that answers consumed by downstream code are treated as failures regardless of intent; parse rate is reported separately so that an accuracy loss from malformed output can be distinguished from one from wrong content.

**AG News label mapping.** The parquet artifact used here stores labels zero-indexed (0 = World, 1 = Sports, 2 = Business, 3 = Sci/Tech), not the original 1..4. This was verified against the raw text: label 1 is a report of Michael Phelps winning gold, label 2 is a report on pension negotiations. Mapping the field as

1..4 shifts every gold label by one class and drives accuracy below the 0.25 chance level. The artifact used is the Hugging Face mirror snapshot of ancyzhx/ag_news, SHA-256 71de87ec66bc5737....

**GSM8K prose extraction.** Because GSM8K generates freely, the final answer is recovered in priority order: the contents of `\boxed{...}`; an explicit `ANSWER:` line; the number following "final answer is" or "answer:"; and only then the last number in the response, after sentences mentioning confidence have been removed. That last exclusion is load-bearing. Responses frequently terminate with a stated confidence, and without removing it the trailing percentage is read as the answer; in the observed failure a response whose correct answer was 6 scored as 99.

### 3.8 Confidence signals

Verbalized confidence is the integer the model reports, scaled to [0, 1], with 0–1 reporting detected and rescaled. For GSM8K it is parsed from prose and is available only for the subset of responses that state a confidence; coverage is reported alongside every calibration metric rather than imputed.

Token-probability confidence is the geometric mean of per-token probabilities over the span of the answer value only. Structural tokens are excluded. This exclusion is essential and easy to omit: under schema-constrained decoding the opening brace and the following quote are forced, and measurement on this pipeline gives them log-probabilities near −14.1 and −7.9 while the answer token sits near 0. A confidence averaged over the whole response is then dominated by formatting the decoder had no choice but to emit.

### 3.9 Calibration metrics

We use *calibration* throughout in the sense of the agreement between a model's stated confidence and its empirical accuracy. This is not the same quantity as the calibration-set size used to estimate quantization parameters in calibrated post-training quantization; the two are unrelated.

We measure calibration with expected calibration error [5], computed over 15 equal-width bins,

$$\mathrm{ECE} = \Sigma_b \, (n_b / n) \cdot |\mathrm{acc}(b) - \mathrm{conf}(b)|,$$

with empty bins skipped rather than treated as zero. MCE is the bin-level worst case. Brier score and negative log-likelihood (probabilities clipped at 1e−6) are binning-free complements: Brier is reported alongside the ECE comparison in Table 4, and MCE and NLL are reported in the archived source data rather than in the main tables.

ECE is computed within each task and then averaged over tasks. Pooling items across tasks before computing ECE would mix tasks with very different accuracy and confidence distributions; measured here, that alternative raises the 0.5B estimate from 0.071 to 0.093.

### 3.10 Statistical analysis

Intervals for per-configuration metrics use percentile bootstrap over items with 2,000 resamples. Configuration contrasts in accuracy and in difference-in-differences use a paired bootstrap with 5,000 resamples, scoring both configurations on the same resampled index so that shared item difficulty cancels. Differences in expected calibration error use a paired bootstrap of 2,000 resamples that resamples within task, matching the within-task aggregation rule, and for that reason their intervals are not numerically

identical to the per-configuration ones. Appendix A.6 lists the resample count for each reported quantity. Unpaired intervals are several times wider and would render most contrasts indistinguishable from noise.

Correction is applied within each table, and the family is the set of rows in that table: four models in Table 1, three contrasts in Table 2, four models in Table 3 and four models in Table 4. Benjamini-Hochberg at FDR 0.05 is applied within each family and each table labels its p column accordingly. The per-task pairwise tests reported in Section 4.5 are uncorrected and are described there as descriptive.

Mean effects are reported separately from worst-case effects. A configuration can pass on the mean and be unusable for the task a given user cares about.

**Relative rather than absolute degradation** is used for size comparisons. A model with lower baseline accuracy has less absolute room to lose, so the size contrast is reported both as Δacc and as Δacc / acc at the reference quant. The two can differ substantially and the choice is stated wherever a claim depends on it.

### 3.11 Scale–capability confound

Size and baseline capability are correlated by construction: the 0.5B model is both smaller and weaker. The within-family Qwen2.5 contrast holds lineage fixed but not capability. We therefore (i) report relative degradation alongside absolute, (ii) verify that the task is not near ceiling or floor for either model before comparing (cells with accuracy below 0.15 or above 0.85 are flagged and excluded from the headline), and (iii) treat the cross-family models as replication rather than as independent tests of the size effect.

### 3.12 Reproducibility and measurement noise

We measured the run-to-run disagreement of the pipeline rather than assuming it zero, and found that it depends on whether a run is inside one serving session or crosses a session boundary.

**Within a session**, greedy decoding is effectively bitwise deterministic. Three runs of the same configuration, each preceded by an explicit model unload, produced identical accuracy on both tasks for both models: 0.875/0.875/0.875 and 0.700/0.700/0.700 for Qwen2.5-3B at Q4_K_M on GSM8K and ARC-Challenge, and 0.450/0.450/0.450 and 0.310/0.310/0.310 for Qwen2.5-0.5B. Prediction-level agreement was 1.000 across all 400 matched items in each comparison.

**Across sessions**, a small drift appears. Comparing the present runs against results collected earlier in the same configuration, the 0.5B model differed on 2 of 200 ARC items (worst |Δaccuracy| = 0.010, prediction agreement 0.965), while the 3B model matched exactly on every item of both tasks.

The practical floor is therefore on the order of 1 to 2 items per 200, that is about 0.005–0.010 at n = 200 and 0.002–0.005 at n = 800. We report it as a count of items, because its magnitude scales with n.

An earlier pilot measurement on a 20-item sample suggested a much larger floor of 0.050. That figure was an artefact of the sample size: a single flipped item is worth 0.05 at n = 20 and 0.005 at n = 200. Measuring the floor at small n inflates it by construction, and the corrected characterization above supersedes it.

The 4-bit effects run from 0.002 to 0.052 in accuracy and from 0.004 to 0.071 in ECE. The larger of these sit an order of magnitude above the floor at n = 800; the smallest, Llama-3.2-3B at 0.002, do not, and for those we report the interval rather than an interpretation. Effects below 0.01 are not interpreted.

### 3.13 Measurement validation

Three scoring defects were found by checking each rule against raw output before interpreting any result. All three produced plausible numbers that would have been reported as findings.

First, the AG News label field was assumed one-indexed. Every gold label was shifted by one class, yielding accuracy below chance (0.140 for a 3B model whose corrected accuracy is 0.700).

Second, GSM8K was evaluated under schema constraint, denying the model a scratchpad and reporting 0.135 instead of the 0.820 the same model achieves when permitted to reason.

Third, the prose parser used for GSM8K treated a trailing confidence figure as the answer. Re-scoring stored outputs changed 112 of 605 records and recovered a stated confidence for 199 that previously had none.

Because raw output and log-probabilities are retained for every record, these corrections were applied by rescoring rather than regenerating, and every number reported here can be recomputed from the released artifacts. Each defect is a reproducible hazard for anyone building a similar evaluation, so we report them.

A fourth check concerned the quantization artifacts rather than the scoring rules. A defective artifact would mimic a real precision effect, so the artifact behind the one anomalous result in Section 4.5 was verified twice. Its SHA-256 digest was compared against the stored copy, and the file size was checked against the server's Content-Length: both matched, ruling out transport corruption. The same nominal level was then obtained from a second, independent quantizer, 90 MB larger, and measured on the same 800 items per task. It reproduced the inversion on both tasks, falling 6.9 points below Q3_K_M on ARC-Challenge and 5.8 points below on MMLU, against 7.0 and 7.0 for the artifact used in the main study. Repeated measurement of this artifact across two sessions differed by 1.2 and 1.3 points on the two tasks, at the upper end of the run-to-run variation characterised in Section 3.12 but well below the size of the gap it was used to test. Both artifacts, their digests, and the per-item records are retained with the results, so the check can be repeated.

### 3.14 Compute environment

Single machine: AMD Ryzen 7 7840H (8 cores, 16 threads), 15.5 GB system memory, NVIDIA GeForce RTX 4060 Laptop GPU with 8 GB, Windows 11. Analysis and figure generation use Python 3.14.6 with numpy 2.5.2, pandas 3.0.5, pyarrow 26.0.0 and matplotlib 3.11.1; all model inference uses Ollama 0.40.6. Decode throughput depends strongly on the quantization level, since smaller artifacts move less data per token: measured aggregate rates were about 23–30 tokens/s for the 3B models at Q8_0 and Q6_K, 75–105 tokens/s at Q4_K_M and below, and about 100–230 tokens/s for the sub-2B models. The complete main experiment occupied approximately 24 GPU-hours.

---

---

## 4. Results

### 4.1 Dataset and measurement integrity

The full experiment comprises 70,394 item-level records across four instruction-tuned models (Qwen2.5-0.5B, Gemma-3-1B, Llama-3.2-3B, Qwen2.5-3B), six GGUF quantization levels (Q8_0 through Q2_K), and four tasks, at 800 items per cell. Llama-3.2-3B is complete at four levels (Q8_0 through Q4_K_M); the Q3_K_M and Q2_K artifacts for that model failed to download within the study window and are reported as missing rather than imputed. Six further responses are absent: three GSM8K cells (Qwen2.5-0.5B at Q8_0 and Q6_K, Qwen2.5-3B at Q6_K) returned no usable generation for one to four items each, and those items were dropped rather than imputed. The design total is therefore 76,800 responses, the total after excluding the two missing Llama levels is 70,400, and the realised total is 70,394. Table 1 reports n = 3,199 for Qwen2.5-0.5B for the same reason.

Answer parsing succeeded for 99.9–100% of responses under the schema-constrained protocol. Stated confidence was available for 99.6% or more of responses from the two 3B models and for 84–87% of responses at Q4_K_M from the 0.5B and 1B models, because the smaller models frequently omitted the confidence field on GSM8K despite the instruction. Calibration metrics are computed only over responses that stated a confidence, and coverage is reported alongside every calibration figure.

A response that does not answer the question is scored incorrect, and two such failure modes turn out to be large and unevenly distributed. **Template echoes** reproduce the placeholder in the prompt itself, so that the extracted answer is the literal string `A-D` or `<letter A-D>`; a raw example is `{"answer": "<letter A-D>", "confidence": 95}`. **Off-label answers** use a vocabulary outside the permitted one, such as `Politics` or `Entertainment` for AG News, or give two letters at once, such as `D, C`. Together they account for 4,273 of the 70,394 responses, 6.1% overall. Their distribution across the ladder is the subject of Fig. 1a.

The distribution is not uniform. Neither failure occurs in Gemma-3-1B or Llama-3.2-3B at any level. In the two Qwen models it is concentrated at particular levels: at Qwen2.5-0.5B the echo rate rises from 1.5% of ARC-Challenge responses at Q8_0 to 19.2% at Q4_K_M, and at Qwen2.5-3B it reaches 30.5% at Q3_K_M. Across all four tasks the share of responses that fail to produce a usable answer goes from 4.0% to 20.2% between Q8_0 and Q4_K_M at 0.5B, from 5.0% to 8.0% at Qwen2.5-3B, and from 1.5% to 2.0% at Gemma-3-1B. Fig. 1b breaks the same rate down by task, which shows that the failure is absent from GSM8K at every level and therefore specific to schema-constrained decoding.

Because of this we report every accuracy contrast under two accounting conventions. The **served** convention counts every response, so a failure to answer is an error; this is what a deployment receives. The **capability** convention restricts the comparison to items that produced a usable answer at both levels, so it measures how often the model is right when it answers. The two differ by the cost of the format failure, and both are reported throughout. Fig. 1c shows that difference for each model.

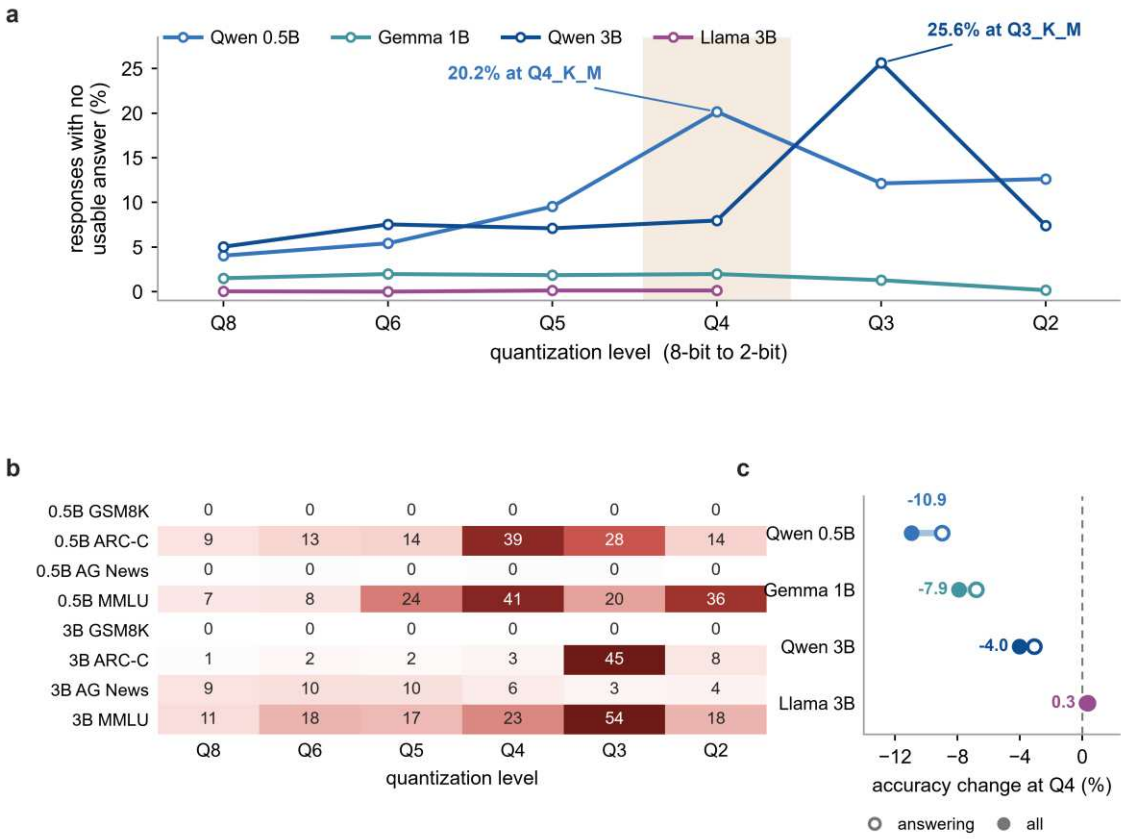



**Fig. 1 Format failure is concentrated at one level per model a**, Share of responses that produced no usable answer, pooled over four tasks, at every published GGUF precision level for four instruction-tuned models. A response is unusable when the extracted answer is a template echo, meaning the prompt placeholder was returned verbatim as in `{"answer": "<letter A-D>", "confidence": 95}`, or an off-label reply, meaning a class name outside the permitted set. The shaded band marks Q4_K_M. **b**, The same rate broken down by task, for the two models that fail. Each cell gives the percentage; the colour scale runs from 0 to 45%. **c**, Accuracy change at Q4_K_M counting every response (filled markers) and counting only items that produced a usable answer at both levels (open markers), as a percentage of each model's own 8-bit accuracy.

### 4.2 Which level breaks is not predictable from the label

Fig. 1a plots the ladder counted as non-answers rather than as accuracy, and the two readings differ in where the damage sits; the accuracy version is Fig. 2a. On ARC-Challenge the non-answer rate for Qwen2.5-0.5B is 1.6% at Q8_0, 5.4% at Q5_K_M, **19.4% at Q4_K_M**, 9.1% at Q3_K_M and 10.4% at Q2_K. This is not a gradient in compression. Q3_K_M and Q2_K are both more compressed than Q4_K_M and both answer more often, so what fails at Q4_K_M is that level and not the coarse end of the ladder. Qwen2.5-3B has the same shape one rung over, at 3.1% for Q4_K_M and 30.6% for Q3_K_M. Gemma-3-1B and Llama-3.2-3B stay at or below 0.1% at every level they have. Pooled over the four tasks the same shape appears with a smaller amplitude: Qwen2.5-0.5B peaks at 20.2% at Q4_K_M and falls to 12.1% at Q3_K_M and 12.6% at Q2_K, while Qwen2.5-3B peaks at 25.6% at Q3_K_M (Fig. 1a). The practical consequence is that adjacent levels do not predict each other: a deployer who measured Q5_K_M and Q3_K_M for Qwen2.5-0.5B would conclude that the ladder is well behaved.

Our primary accuracy comparison is Q8_0 against Q4_K_M, the ladder step from 8-bit to the level most commonly shipped. Three of the four models lose accuracy at Q4_K_M, and the magnitude varies by more than an order of magnitude across them (Table 1). The full ladder is shown in Fig. 2a: both sub-2B models are already below their 8-bit accuracy at Q4_K_M, whereas Qwen2.5-3B loses 4.0% at Q4_K_M ($p<0.0001$) and Llama-3.2-3B does not lose accuracy at any level measured. The two 3B models therefore disagree, and the tier is not homogeneous.

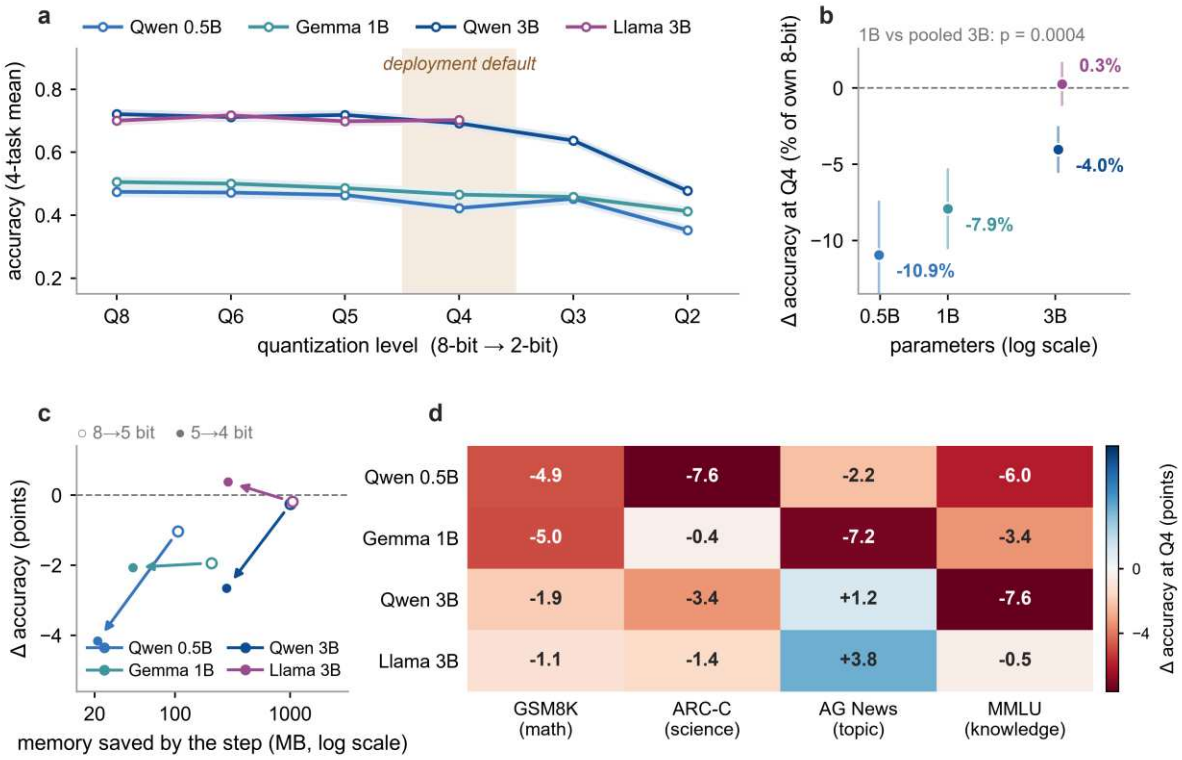



**Fig. 2 Accuracy change at the 4-bit step ranges from a 0.3% gain to a 10.9% loss, concentrated in the step that saves least memory a**, Accuracy across the quantization ladder for four instruction-tuned models (4-task mean over GSM8K, ARC-Challenge, AG News and MMLU; 800 items per model–task cell). Points are means; bands are 95% percentile-bootstrap intervals over items (2,000 resamples). The shaded band marks Q4_K_M. **b**, Accuracy change from Q8_0 to Q4_K_M as a percentage of each model's own 8-bit accuracy, against parameter count (log scale). Vertical bars are 95% paired-bootstrap intervals over matched items (5,000 resamples). **c**, The two steps a deployment decision turns on, drawn as one arrow per model from its 8→5-bit step (hollow marker) to its 5→4-bit step (filled marker). Memory saved is the difference between GGUF artifact sizes. **d**, Per-task accuracy change at Q4_K_M relative to Q8_0, in percentage points.

**Table 1. Accuracy at Q8_0 and Q4_K_M, pooled over four tasks, counting every response, so a response with no usable answer is scored wrong (served convention). Paired bootstrap, 5,000 resamples; BH within table.**

| Model | Size | n | Q8_0 | Q4_K_M | Δ [95% CI] | Rel. served | Non-answer |
|---|---|---|---|---|---|---|---|
| Qwen2.5-0.5B | 0.5B | 3,199 | 0.474 | 0.423 | −0.052 [−0.068, −0.035] | **−10.9%** | 4.0% → 20.2% |
| Gemma-3-1B | 1B | 3,200 | 0.505 | 0.465 | −0.040 [−0.053, −0.027] | −7.9% | 1.5% → 2.0% |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Qwen2.5-3B | 3B | 3,200 | 0.721 | 0.692 | −0.029 [−0.040, −0.019] | −4.0% | 5.0% → 8.0% |
| Llama-3.2-3B | 3B | 3,200 | 0.700 | 0.702 | +0.002 [−0.008, +0.012] | +0.3% | 0.0% → 0.1% |

**Table 2. The same contrast restricted to items that produced a usable answer at both levels (capability convention). Paired bootstrap, 5,000 resamples; BH within table.**

| Model | n | Q8_0 | Q4_K_M | Δ [95% CI] | Rel. capability |
|---|---|---|---|---|---|
| Qwen2.5-0.5B | 3,199 | 0.474 | 0.423 | −0.043 [−0.059, −0.026] | **−9.0%** |
| Gemma-3-1B | 3,200 | 0.505 | 0.465 | −0.035 [−0.048, −0.022] | −6.8% |
| Qwen2.5-3B | 3,200 | 0.721 | 0.692 | −0.023 [−0.033, −0.013] | −3.1% |
| Llama-3.2-3B | 3,200 | 0.700 | 0.702 | +0.003 [−0.007, +0.012] | +0.4% |

Both conventions give the same ordering: the cost is largest at 0.5B and smallest at Llama-3.2-3B. The conventions differ in magnitude, and the difference is the format cost. At 0.5B that cost is 1.9 accuracy points, or 1.9% of the 8-bit baseline; at Qwen2.5-3B it is 0.9 points. At Gemma-3-1B it is 1.1 points, which is the part of its loss that comes from the small rise in non-answers rather than from the levels being unanswerable.

Pooled by size under the served convention, the degradation is −0.052 at 0.5B, −0.040 at 1B, and −0.014 at 3B, corresponding to −10.9%, −7.9% and −1.9% of each model's own Q8_0 accuracy. A pooled figure concatenates the matched-item deltas of the models at that size, so the 3B pool draws on both Qwen2.5-3B and Llama-3.2-3B. The same quantity is plotted against parameter count in Fig. 2b, where the two 3B models share a single x position.

The trend holds across families but not across the two 3B models, which disagree: Qwen2.5-3B loses 4.0% under the served convention and 3.1% under the capability convention, while Llama-3.2-3B is flat under both.

Which convention a reader wants depends on the question. A deployment that will receive whatever the model emits should budget for the served figure. A reader asking how much knowledge 4 bits costs should use the capability figure. The gap between them is not noise; it is the model's loss of compliance with the output specification, and it is largest exactly where the served figure is largest.

The cost is also not confined to one task. Fig. 2d breaks the Q8_0→Q4_K_M change down by task. Both sub-2B models lose accuracy on all four, whereas the two 3B models gain on AG News (+1.2 and +3.8 points) and their losses are concentrated elsewhere. The largest single loss at 0.5B is ARC-Challenge at −7.6 points, but the same task costs the 3B models 3.4 and 1.4 points, so the pooled mean is not an artefact of one benchmark.

A difference-in-differences test on matched items confirms the interaction (Table 3). Comparing pooled sizes:

**Table 3. Size interaction, difference-in-differences on matched items, with the 3B tier entered as the mean of its two models (BH within table).**

| Contrast | n | DiD | 95% CI | p (BH) |
|---|---|---|---|---|
| 0.5B vs 1B | 3,199 | −0.0119 | [−0.032, +0.008] | 0.25 |
| 1B vs 3B | 3,200 | −0.0264 | [−0.041, −0.012] | 0.0004 |
| 0.5B vs 3B | 3,199 | −0.0383 | [−0.055, −0.021] | 0.0002 |

The 3B tier is not internally homogeneous: Qwen2.5-3B loses 4.0% at Q4_K_M while Llama-3.2-3B is flat, and the magnitude of the interaction depends on how that heterogeneity is handled. Against Llama-3.2-3B alone the two 3B contrasts are −0.0419 and −0.0538; against Qwen2.5-3B alone they are −0.0109 ($p = 0.19$, not resolved) and −0.0228 ($p = 0.017$). We report the pooled contrast as the primary estimate because it uses both models, and we state the alternatives because the choice changes the effect size by a factor of 1.6.

The adjacent-sub-2B contrast (0.5B vs 1B) is underpowered at this sample size: the two differ by 1.2 accuracy points in absolute terms, or 3.0 points in relative terms, which is below what 3,200 items per model can resolve. The contrast that the data do support is the one that matters for deployment: below 2B, 4-bit costs between 8% and 11% of accuracy, while at 3B it costs between 0% and 4%.

### 4.3 Stated confidence degrades on a different axis than accuracy

Calibration across the ladder is shown in Fig. 3a. ECE is computed within each task and then averaged over tasks. This definition matters and is stated because the alternative is wrong: pooling items across tasks before computing ECE mixes tasks with very different accuracy and confidence distributions and inflates the value, taking the 0.5B estimate from 0.071 to 0.093 — a change large enough to alter which comparisons look significant.

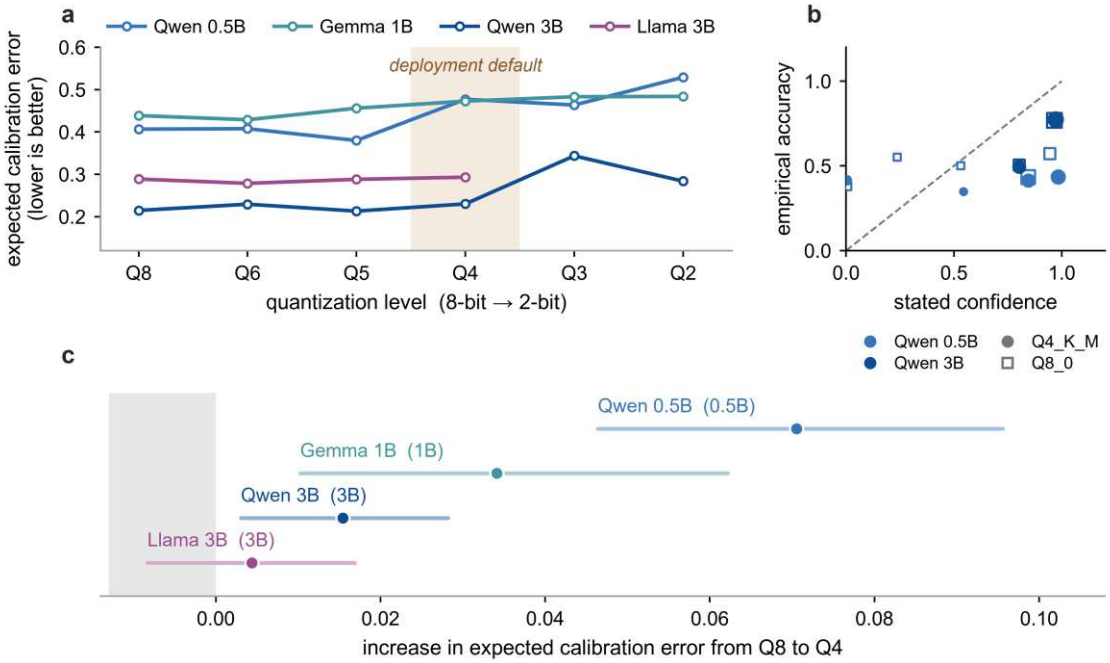



**Fig. 3 The 4-bit step raises calibration error in three of four models, most strongly at the smallest a**, Expected calibration error (ECE) across the quantization ladder for the same four models. ECE is computed within each task over 15 equal-width confidence bins and then averaged across tasks; lower is better. The shaded band marks Q4_K_M. **b**, Reliability of the smallest and largest Qwen models: empirical accuracy against stated confidence, binned at 10 equal widths with bins holding fewer than 20 items omitted. Filled circles are Q4_K_M and open squares are Q8_0; marker area is proportional to the number of items in the bin. Points below the dashed diagonal indicate over-confidence. **c**, Increase in ECE from Q8_0 to Q4_K_M for each model, with 95% paired-bootstrap intervals over items resampled within task (2,000 resamples per model). The grey band marks no change.

**Table 4. Expected calibration error at Q8_0 and Q4_K_M, with paired bootstrap difference (BH within table).**

| Model | Size | n (with conf.) | ECE Q8_0 | ECE Q4_K_M | Δ ECE | 95% CI | p (BH) |
|---|---|---|---|---|---|---|---|
| Qwen2.5-0.5B | 0.5B | 2,669 | 0.407 | 0.477 | **+0.071** | [0.048, 0.098] | <0.001 |
| Gemma-3-1B | 1B | 2,599 | 0.439 | 0.473 | **+0.034** | [0.010, 0.065] | 0.008 |
| Qwen2.5-3B | 3B | 3,190 | 0.215 | 0.230 | +0.015 | [0.003, 0.029] | 0.011 |
| Llama-3.2-3B | 3B | 3,192 | 0.289 | 0.293 | +0.004 | [−0.007, 0.016] | 0.50 |

Table 4 gives the per-model estimate for the stated-confidence channel. The estimates are ordered by size, 0.071 at 0.5B, 0.034 at 1B, and 0.015 or below at 3B (Fig. 3c). Three of the four models show a change whose 95% interval excludes zero; the Llama 3B model does not. The ordering itself is descriptive rather than tested, since the 0.5B and 1B intervals overlap ([0.048, 0.098] and [0.010, 0.065]).

Whether this is a calibration finding or a restatement of the accuracy finding depends on what happens to stated confidence itself. At near-constant confidence, ECE is approximately one minus accuracy, so a rise

in ECE would carry no information beyond the accuracy result. Table 5 separates the two components. It uses the three tasks for which token probabilities were retained, where Table 4 uses all four, so the 0.5B contrast is +0.114 there against +0.071 here. Dropping GSM8K removes a quarter of the task weight but only about a tenth of the responses, which is why the two estimates differ by 60 percent. The four-task figure is the headline and the three-task figure is the like-for-like comparison with the token channel.

For an over-confident channel the change in ECE is, to the accuracy of the binning, the change in confidence minus the change in accuracy; the ΔECE column is therefore an accounting identity rather than an independent estimate, and the two component columns carry the information.

**Table 5. Confidence and accuracy changes at the 4-bit step, on both confidence channels, over the three tasks for which token probabilities were retained (2,400 matched items per model; paired bootstrap, 2,000 resamples).**

| Model | Channel | Conf. Q8 | Conf. Q4 | Acc. Δ | ECE Q8 | ECE Q4 | ΔECE [CI] | ΔBrier |
|---|---|---|---|---|---|---|---|---|
| Qwen2.5-0.5B | stated | 0.862 | 0.922 | −0.053 | 0.387 | 0.501 | +0.114 [+0.095, +0.132] | +0.106 |
| Qwen2.5-0.5B | token | 0.534 | 0.549 | −0.053 | 0.066 | 0.126 | +0.060 [+0.034, +0.077] | +0.021 |
| Gemma-3-1B | stated | 0.949 | 0.949 | −0.037 | 0.451 | 0.487 | +0.036 [+0.022, +0.049] | +0.032 |
| Gemma-3-1B | token | 0.967 | 0.963 | −0.037 | 0.470 | 0.501 | +0.031 [+0.017, +0.045] | +0.025 |
| Qwen2.5-3B | stated | 0.916 | 0.899 | −0.033 | 0.242 | 0.257 | +0.015 [+0.004, +0.028] | +0.014 |
| Qwen2.5-3B | token | 0.951 | 0.955 | −0.033 | 0.278 | 0.313 | +0.035 [+0.022, +0.047] | +0.037 |
| Llama-3.2-3B | stated | 0.983 | 0.991 | +0.006 | 0.327 | 0.329 | +0.001 [−0.010, +0.011] | −0.001 |
| Llama-3.2-3B | token | 0.861 | 0.870 | +0.006 | 0.199 | 0.202 | +0.003 [−0.008, +0.012] | −0.002 |

Three readings follow.

First, only the 0.5B model becomes more confident as it becomes less correct. Its stated confidence rises from 0.862 to 0.922 while its accuracy falls by 5.3 points. That is over-confidence in the strict sense, and it is largest where the model is smallest.

Second, at 1B and 3B the confidence term is flat or negative, at 0.000 for Gemma-3-1B and −0.017 for Qwen2.5-3B. Their ECE increases are what a near-constant confidence implies when accuracy falls. For these two models the calibration number carries little information beyond the accuracy number.

Third, the token-probability channel behaves differently. At 0.5B the implied confidence rises by 0.015 rather than 0.060, and the size ordering of the ECE changes does not survive: +0.060, +0.031, +0.035 and +0.003 for 0.5B, 1B, Qwen 3B and Llama 3B. On the Brier score the Qwen 3B model changes more than the 1B model, +0.037 against +0.025. The claim that the calibration cost follows the accuracy gradient therefore holds for what the model says and not for what its probabilities imply.

The gap between the two channels is itself a result. At 0.5B the token probabilities are close to calibrated at Q8_0 (ECE 0.066) while the stated confidence is not (0.387). The model represents an uncertainty that it does not report, and the 4-bit step widens that gap.

This bears directly on the hypothesis we set out to test. The stated-confidence channel degrades on roughly the same ladder step as accuracy, so an accuracy-only acceptance test on these models would not have missed it, and where accuracy is unharmed the stated confidence is unharmed too. The one place the damage has the form of over-confidence rather than of error is the smallest model, and there the two arrive on the same step rather than in sequence. The hypothesis is not supported.

The reliability panel in Fig. 3b shows the same effect at the level of individual confidence bins rather than as a summary statistic. Stated confidence is not continuous: it concentrates on round values, so the 10 equal-width bins collapse to 2-5 occupied bins per model, and the panel is drawn as markers with area proportional to bin count rather than as connected curves. The high-confidence bins of both models lie below the diagonal, which is the over-confidence the ECE reports. The 0.5B model also has bins above the diagonal at the opposite end, where it states a confidence of 0.3 to 0.6 percent and is right 38 to 41 percent of the time; we return to those values in Section 5.5. The 0.5B model at Q4_K_M places 1,488 of its 2,669 responses in a bin centred at 0.98 stated confidence while answering 43% of them correctly. The 3B model places 2,288 responses in a bin at 0.97 confidence and answers 77% correctly. For a routing rule that sends high-confidence queries for automatic handling, the two models are not interchangeable at the same precision.

### 4.4 Below 4-bit the picture inverts, and the inversion is an effect of headroom

At Q3_K_M and Q2_K the ordering reverses: the larger model loses more (Table 6).

**Table 6. Absolute accuracy change from Q4_K_M to the two lowest levels, referenced to Q4_K_M.**

| Size | Δ at Q3_K_M | Δ at Q2_K |
|---|---|---|
| 0.5B | **+0.030** | −0.070 |
| 1B | −0.007 | −0.053 |
| 3B (Qwen2.5) | −0.055 | −0.215 |

At Q3_K_M the ordering is already inverted: the 0.5B model gains 3.0 points over Q4_K_M while Qwen2.5-3B loses 5.5, and at Q2_K the 3B model loses more in absolute terms, 21.5 points against 7.0.

Headroom confounds that reading, and the comparison has to be made within one task rather than across a task mean and a single-task baseline. On ARC-Challenge the 0.5B model sits at 0.412 at Q8_0 against a 0.25 chance level and the 3B model at 0.759, so the available headroom is 0.162 and 0.509. Against that headroom the 3B model loses 78% of its margin at Q2_K and the 0.5B model 57%, a much smaller gap than the raw point differences suggest.

At Q2_K every model is degraded to the point of being unusable for these tasks, and cross-model comparisons at that level are confounded by differing headroom.

### 4.5 One anomaly: nominal bit-width does not always order quality

For Qwen2.5-0.5B, Q4_K_M scores 0.422 while Q3_K_M scores 0.452. The lower nominal bit-width is better by 3.0 points, and the difference is significant on three of four tasks (ARC +0.070, $p<0.0001$; MMLU +0.070, $p=0.0004$; AG News +0.025, $p=0.044$), with the opposite sign on GSM8K (−0.046, $p=0.008$). The cross-quantizer check below uses the same 800 items per task.

This does not generalize. Gemma-3-1B is strictly monotonic across all six levels and Qwen2.5-3B is monotonic apart from a 0.7-point Q5/Q6 inversion that is within noise. Llama-3.2-3B is also non-monotonic in the opposite direction, gaining 1.7 points from Q8_0 to Q6_K ($p<0.001$). That is smaller than the 3.0-point inversion that motivates this section, but it is significant and our mixed-precision account does not explain it away as noise. The effect appears in one of three models with a complete ladder.

GGUF k-quants are mixed-precision schemes that allocate different numbers of bits to different tensors, so the nominal label need not order quality. A defective artifact would produce the same signature, so we repeated the Q4_K_M measurement with an artifact produced by a different quantizer, the official Qwen repository, which is 90 MB larger at the same nominal level. That artifact scores 0.338 on ARC-Challenge and 0.329 on MMLU, against 0.336 and 0.316 for the artifact used elsewhere in this study. It therefore also falls below Q3_K_M on both tasks (0.406 and 0.386), by 6.9 and 5.8 points respectively, against 7.0 and 7.0 for the first artifact. The inversion is a property of the quantization level at this model size rather than of one implementation. We report it because it is statistically solid where it occurs and because it warns against assuming that a coarser label is always worse.

### 4.6 Where the degradation begins

Across the two 3B models, accuracy is flat from Q8_0 through Q5_K_M and falls only at Q3_K_M and below. For the sub-2B models the drop begins one step earlier, at Q4_K_M. This places the practical boundary differently for different model sizes: 5-bit costs at most 1.9 points and is indistinguishable from 8-bit in three of four models, 4-bit is safe at Llama-3.2-3B and costly elsewhere, and 3-bit is costly at every size where it was measured.

### 4.7 Summary of what the data do and do not support

Supported: that format failure is a substantial and unevenly distributed component of measured quantization cost, reaching 19.4% of ARC-Challenge responses at Qwen2.5-0.5B Q4_K_M and 30.6% at Qwen2.5-3B Q3_K_M while never exceeding 0.1% in the other two models; that the level at which it occurs differs between models and is not monotone in bit-width; that the gap between the served and capability conventions, up to 1.9 accuracy points at 0.5B, is this failure rather than knowledge loss; one artifact's Q4_K_M scores 3.0 points below its Q3_K_M on matched items, and an independent quantizer reproduces that; the interaction with size is significant against the pooled 3B tier ($p\leq0.0004$) but not against Qwen2.5-3B alone ($p=0.19$); stated confidence inflates most at the smallest model, which is the only one that grows more confident while becoming less correct; and the ranking between adjacent sub-2B sizes is not resolved at this sample size.

Not supported: that a nominal bit-width fixes the cost of a quantized deployment; that the size trend rests on more than one model; that calibration degrades before accuracy; that the size gradient in calibration survives on a measure that does not depend on what the model says; that large models are more fragile at low precision; that the Q4-vs-Q3 inversion generalizes beyond the model in which it was observed.

Not measured: models above 3B, weight-only 4-bit formats other than GGUF k-quants, sampling-based uncertainty at temperature above zero, token-probability confidence on GSM8K, and (for Llama-3.2-3B) the two lowest quantization levels.

---

---

## 5. Discussion

### 5.1 The 4-bit default is not where the memory is saved

The accuracy cost answers how much the 4-bit step costs. It does not answer where on the ladder the cost accrues, and the two answers differ. For a fixed model, the memory saved and the accuracy lost are badly matched across adjacent levels, and Fig. 2c plots that trade for the two steps a deployment decision actually turns on, with the underlying numbers in Table 7.

**Table 7. Accuracy change and memory saved for adjacent quantization steps (GGUF file size).**

| Model | Step | Memory saved | Δ accuracy | 95% CI | p |
|---|---|---|---|---|---|
| Qwen2.5-0.5B | Q8_0 → Q5_K_M | 106 MB | −0.010 | [−0.025, +0.004] | 0.17 |
| | **Q5_K_M → Q4_K_M** | **21 MB** | **−0.042** | [−0.058, −0.026] | <0.001 |
| Gemma-3-1B | Q8_0 → Q5_K_M | 208 MB | −0.019 | [−0.030, −0.009] | <0.001 |
| | Q5_K_M → Q4_K_M | 43 MB | −0.021 | [−0.033, −0.008] | 0.002 |
| Qwen2.5-3B | Q8_0 → Q5_K_M | 1,012 MB | −0.003 | [−0.010, +0.005] | 0.55 |
| | **Q5_K_M → Q4_K_M** | **281 MB** | **−0.027** | [−0.037, −0.017] | <0.001 |
| Llama-3.2-3B | Q8_0 → Q5_K_M | 1,049 MB | −0.002 | [−0.010, +0.006] | 0.64 |
| | Q5_K_M → Q4_K_M | 289 MB | +0.004 | [−0.006, +0.013] | 0.45 |

Memory saved is the difference between the two GGUF artifact sizes on disk, not the nominal bit-width ratio, so it is the saving a deployer actually obtains.

At 0.5B, dropping from 8-bit to 5-bit removes 106 MB and costs 1.0 accuracy point, though that interval spans zero ([−0.025, +0.004], p=0.17); dropping from 5-bit to 4-bit removes 21 MB and costs 4.2 points ($p<0.001$). The ratio of a significant effect to a null is not formally identified, but the ordering reverses, and it reverses the same way at 3B. The same ordering holds at 3B, where 1,012 MB is nearly free and 281 MB is not.

This suggests that a practitioner running a model of a few billion parameters or fewer on GGUF k-quants should test Q5_K_M against Q4_K_M before committing to the standard 4-bit artifact. The memory difference is small next to the total footprint, and for three of our four models it is the expensive step.

The recommendation rests on four models, four tasks, and one quantization family.

### 5.2 Why the smallest model may lose most

We have no mechanism evidence, so the following are hypotheses our design cannot test.

*Redundancy.* A smaller model has fewer parameters to distribute the same functional load, so a perturbation of fixed relative size may remove a larger share of the computation. If this is the mechanism, the loss should be traceable to a few layers with unusually large quantization error, and the depth profile of error should differ between the 0.5B and 3B models.

*Outlier structure and block allocation.* GGUF k-quants assign bits per tensor, and the allocation was chosen for a family of models rather than tuned per checkpoint. Outlier features are known to dominate quantization damage in large models [2]. A small model may have a different ratio of outlier to typical weights, so a fixed block layout could fit it worse. This would predict that a per-tensor reallocation recovers more of the loss at 0.5B than at 3B.

*Margins.* An instruction-tuned small model may arrive at many correct answers by narrow logit margins, so the same amount of numerical noise flips more of them. This predicts that the fraction of items whose top-2 logits differ by less than a threshold should be larger at 0.5B, and should correlate with the items that changed between Q8_0 and Q4_K_M.

Each of these is testable with data we did not collect: per-layer quantization error, weight and activation outlier statistics, and logit margins on the evaluation items. We report the gradient and leave the mechanism open.

### 5.3 Relation to the parallel-scaling description

Our results are consistent with the existing description over most of the range. Between 8 and 5 bits the degradation is at most 1.9 points and reaches significance in one of four models, Gemma-3-1B at −0.019 ($p<0.001$); in the other three it is within noise. The deviation appears at 4 bits, where the cost is 10.9% at 0.5B, 7.9% at 1B, and 1.9% pooled at 3B.

That pattern matters for how such conclusions are tested. A scaling law of separable multiplicative form, of the kind used in prior task-stratified work, has no cross term to estimate; an interaction of the size we observe would appear only as a reduction in goodness of fit, where it would be difficult to distinguish from noise in the fit. Testing separability requires a design that contrasts configurations directly, as the difference-in-differences contrast does here.

The separability assumption is reasonable, is supported over most of the bit-width range in our own data, and fails only where the compression is aggressive.

The direction of the calibration result is not universal.[6] report that quantization can improve calibration for vision-language models, and attribute the gain to the suppression of high-rank spectral components. Our data show the opposite for text-only instruction-tuned models on knowledge and reasoning tasks. The

two evaluations stress different things: theirs, distribution shift and out-of-distribution detection, where they argue that discarding high-rank directions acts as a regulariser; ours, in-distribution knowledge and multi-step reasoning, where the discarded directions appear to carry task-relevant signal. Separating these accounts requires applying their spectral analysis to our checkpoints. We did not do that, and it is the most direct follow-up this study suggests.

### 5.4 What this implies for evaluating quantized models

Three practices follow from the results and are cheap to adopt.

Report per size, not pooled. Our pooled 1B and 3B figures conceal a range from +0.3% to −4.0% at 3B and would be misleading on their own.

Measure at the bit-width you intend to deploy. A study that evaluates at 8-bit and extrapolates, or that reports a single 4-bit number for a model family, would not have shown the Q5-to-Q4 step.

Do not treat one model's sensitivity as a family property. The two 3B models differ by 4.3 percentage points in their response to the same quantization step, which is larger than the difference between the 1B and 3B pooled estimates.

### 5.5 Limitations

One behaviour enters the calibration numbers directly and deserves separate statement. The 0.5B model occasionally reports a confidence of exactly zero: 638 of the 15,928 responses for which it stated one, of which 544 are on GSM8K. The raw output shows this is not a parsing artefact, since the model writes `"confidence": 0` in the JSON field and, in free-form responses, sentences such as "the confidence level is 0". It is a failure of instruction following rather than a genuine report of low confidence. Those responses land in the lowest confidence bin, where the model is nonetheless right 38 to 41 percent of the time, so they are under-confident and they pull the calibration error down. Excluding them raises the 0.5B estimate from 0.407 to 0.414 at Q8_0 and lowers it from 0.477 to 0.472 at Q4_K_M. The Q8-to-Q4 increase therefore falls from 0.071 to 0.058. The zero-confidence responses inflate the reported change rather than deflating it, and the 0.071 figure should be read as an upper bound under this choice. The behaviour is an instruction-following failure rather than a size-graded one: Gemma-3-1B produces five such outputs, Qwen2.5-3B two, and Llama-3.2-3B eighty-four, so the 3B Llama model produces more than the 1B Gemma model.

Our conclusions are specific to GGUF k-quants. The 4-bit weight-only formats used on datacenter accelerators apply different rounding and grouping, and our numbers should not be transferred to them without measurement.

The models are at most 3B parameters. We have no evidence about whether the gradient continues, flattens, or reverses above that range.

Llama-3.2-3B is missing the Q3_K_M and Q2_K levels, so the low-precision analysis rests on three models rather than four.

Calibration is computed within task and averaged across tasks. The alternative of pooling items across tasks before computing ECE raises the 0.5B estimate from 0.071 to 0.093, so our calibration numbers are not comparable to studies that pool.

Stated confidence was available for 84–87% of responses from the two smaller models at Q4_K_M and 99.6% or more from the 3B models. The paired estimate in Table 4 is computed on the intersection of items carrying a stated confidence at both levels, which is 81–83% of the 3,200 per model; the two percentages answer different questions and both are reported. GSM8K in particular contributes accuracy for every item but calibration only for the subset that states a confidence in prose. Calibration for the smaller models is therefore estimated on a subset that may not be representative, and we cannot rule out that the responses which omitted a confidence were also the least reliable.

Verbalized confidence is poorly calibrated in absolute terms even at full precision, which is why the analysis emphasizes changes across precision levels rather than absolute calibration quality. Table 5 shows that this is a property of the channel rather than of the model: at 0.5B the token probabilities are close to calibrated at Q8_0 (ECE 0.066) while the stated confidence is not (0.387). The token-probability signal is reported as an independent check that does not depend on the model's willingness to state a confidence.

Four tasks cannot cover the space of capabilities. Long-context behavior, code generation, and multilingual performance are not represented; the last is a natural target for extension, and would be the first thing we would add.

Size and baseline capability are correlated by construction (Section 3.11). The pooled 3B figures hold lineage fixed only within the Qwen family, and the cross-family comparison at 3B cannot separate parameter count from general competence. The mechanism behind the gradient is therefore not identified by this design.

Only greedy decoding is used. Sampling-based estimates of uncertainty at temperature above zero are not characterized.

Finally, we have no mechanism evidence. The pattern is measured, not explained, and Section 5.2 lists what would be needed to explain it.

---

---

## 6. Statements and Declarations

### Ethics statement

This study uses publicly available benchmark datasets and open-weight models served locally. It involves no human participants, no personal data and no animal experimentation, and required no ethics review.

### Data and code availability

The analysis code, the source-data tables behind every figure panel and behind the two-convention and robustness analyses, and the cross-quantizer verification records are archived at https://github.com/ZhangShaoXuan-HIT/gguf-4bit-audit. `finalize_stats.py` produces the statistics tables, `build_source_data.py` the figure source data, and `make_figures_nature.py` the figures; every number reported here can be recomputed from those files. The per-item records for all 70,394 responses, including the raw model output and token log-probabilities for each item, total 2.3 GB and are deposited

separately at **[Zenodo DOI, to be inserted on acceptance]**. The quantized artifacts are third-party downloads and are identified in the paper by repository, file name and byte count; they are not redistributed.

### Author contributions

Shaoxuan Zhang: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization.


### Funding

This research received no specific grant from any funding agency.


### Competing interests

The author has no relevant financial or non-financial interests to disclose.

### Use of AI-assisted tools

The author used a generative AI assistant (DeepSeek-V4.1-Flash 10.17) for literature verification, typesetting adjustments, and article translation. The author verified all results, analyses, and references and takes full responsibility for the content.

---

## Appendix A. Measurement protocol

This appendix gives the exact elicitation and extraction rules, so that an

independent group can reproduce the measurements from the description alone

rather than from the code. Section references are to the main text.

### A.1 Prompts

Every item is sent as a single user message. No system prompt is used. The four

templates are reproduced verbatim, with the variable parts in angle brackets.

**GSM8K** (free-form protocol, `NUM_PREDICT` = 512)

`

Solve the following grade-school math problem.

Problem: <question>

Reason step by step, then finish with exactly these two lines:

ANSWER: <the final number, digits only>

CONFIDENCE: <an integer from 0 to 100>

`

**ARC-Challenge** (JSON protocol, `NUM_PREDICT` = 96). The template as implemented caps the answer at D although the task admits five options; two of the 800 sampled items offer five, so the cap affects at most two items.

```
Answer the multiple-choice science question.

Question: <question>

A) <choice A>

B) <choice B>

...

Return JSON only: {"answer": "<letter A-D>", "confidence": <0-100 integer>}
```

**AG News** (JSON protocol, `NUM_PREDICT` = 96). The article text is truncated to its first 1,200 characters.

```
Classify the news article into exactly one topic from: World, Sports, Business, Sci/Tech.

Article: <text>

Return JSON only: {"answer": "<one of the four topics>", "confidence": <0-100 integer>}
```

**MMLU** (JSON protocol, `NUM_PREDICT` = 96). The subject name is inserted into the first line; the option letters span the number of choices actually offered.

```
Answer the multiple-choice question (subject: <subject>).

Question: <question>

A) <choice A>

B) <choice B>

...

Return JSON only: {"answer": "<letter A-D>", "confidence": <0-100 integer>}
```

### A.2 Decoding

All configurations use the same settings, held fixed across models and precision levels.

| Parameter | Value | Note |
|---|---|---|
| temperature | 0.0 | greedy |
| top_k | 1 | redundant with temperature 0, recorded to make intent explicit |
| top_p | 1.0 | |
| seed | 42 | passed to the serving API |
| num_ctx | 4096 | pinned |
| num_predict | 512 (GSM8K), 96 (others) | |

The context window is pinned deliberately. Leaving it unset lets the serving stack choose, and a different choice between runs silently truncates long prompts. On this machine an unpinned rerun of identical weights moved one task by 0.10 in accuracy.

The JSON tasks are decoded under a schema constraint supplied through the `format` field of the Ollama generation API:

```json
{"type": "object",
"properties": {"answer": {"type": "string"},
"confidence": {"type": "integer", "minimum": 0, "maximum": 100}},
"required": ["answer", "confidence"]}
```

GSM8K is deliberately excluded from this constraint. Schema-constrained decoding forces a JSON object from the first token, which leaves no room for a scratchpad; under that protocol the same weights scored 0.135 rather than 0.820 on GSM8K (Section 3.13).

### A.3 Answer extraction

**JSON protocol.** The response is parsed as JSON and the `answer` field is read.

The four tasks reduce it as follows: ARC-Challenge and MMLU take the first option letter; AG News matches against the four class names; a response that cannot be parsed is scored incorrect and counted in the parse rate.

**GSM8K free-form protocol.** Four rules are tried in order, and the first that yields a number is used:

1. the contents of `\boxed{...}`;
2. the text following an `ANSWER:` line, anywhere in the response;
3. a number introduced by "final answer" or "answer is";
4. the last number in the response.

Before rule 4 runs, every sentence mentioning confidence, certain or sure is removed. Without that step a correct response ending "**Confidence level: 99%**" was read as the answer, and one item whose gold value was 6 was scored as 99 (Section 3.13).

### A.4 Confidence extraction

**JSON protocol.** The `confidence` field is an integer from 0 to 100. A branch intended to rescue models that report a proportion instead takes a value falling in [0, 1] and multiplies it by 100, but the value is then divided by 100 again before clamping, so the branch is an identity and changes nothing. A reported 0 therefore maps to 0, which is the origin of the zero-confidence responses in Section 5.5, and a reported 1 maps to 1.0, that is to full confidence. Of the 64,239 responses that stated a confidence, 35 reported the integer 1 and the remainder of that group reported 100; the confusion therefore affects 0.05% of the responses and does not change the calibration results. The rule is recorded here as implemented rather than as intended.

**GSM8K free-form protocol.** Three patterns are tried in order: a `CONFIDENCE:` line, the first integer within 30 characters of the word "confidence", and a percentage sign. Confidence is therefore available only for the subset of

responses that state one, and coverage is reported alongside every calibration metric.

**Token-probability confidence.** For the three JSON tasks the confidence signal is also computed from the token log-probabilities over the span of the answer value: the geometric mean of the per-token probabilities, as the model's own distribution over the tokens it emitted. No tokens outside the answer span enter, because the schema constraint forces the surrounding structural tokens and would contaminate any average taken over them. GSM8K does not store answer-span log-probabilities, so the token-probability channel covers three tasks and Table 4 is restricted accordingly.

### A.5 Item sampling

The sampler is seeded once and per task, so that every model and precision level sees an identical sample and every contrast is paired at the item level.

- Base seed: `SAMPLE_SEED = 20261108`.
- Per-task seed: `SAMPLE_SEED + int(md5(task_name)[:6], 16)`, where the hex prefix is interpreted as an integer.
- Items are drawn without replacement from the test split with `random.Random(seed_task).sample(range(n_items), n)`.
- The sample size is 800 per task for the main experiment, 200 for the pilot and 20 for pipeline verification.

Because the draw depends only on the task name and the base seed, an independent run reproduces the same items.

### A.6 Resampling and multiplicity

Intervals are percentile bootstrap intervals, and every contrast is computed on matched items with the same resampling index applied to both sides.

| Quantity | Resamples |
|---|---|
| Table 1 differences in accuracy | 5,000 |
| Table 2 difference-in-differences | 5,000 |

| | |
|---|---|
| Table 3 difference in ECE | 2,000 |
| Table 4 difference in ECE and Brier | 2,000 |
| Figure 1a and 1b intervals | 2,000 (point) / 5,000 (difference) |
| Figure 2c intervals | 2,000 |

ECE differences resample items independently within each task and average the within-task estimate over tasks, so the interval reflects the same aggregation rule as the point estimate.

Benjamini-Hochberg correction at FDR 0.05 is applied within each table, and the family is the set of rows in that table: four models in Table 1, three contrasts in Table 2, four models in Table 3. The per-task pairwise tests in Section 4.5 are reported uncorrected and are described as descriptive.

### A.7 Reproducibility of the numbers in this appendix

The prompts, the schema, the extraction rules and the seeds above are the implemented values in `config.py`, `data_loaders.py` and `tasks.py` in the archived code (https://github.com/ZhangShaoXuan-HIT/gguf-4bit-audit). Running `run_eval.py --stage full` with those files reproduces the per-item records; the statistics and figures follow from `finalize_stats.py`, `figures/build_source_data.py` and `figures/make_figures_nature.py`.

---

## References


[1] Dettmers T, Zettlemoyer L (2023) The case for 4-bit precision: k-bit inference scaling laws. In: Proceedings of the 40th International Conference on Machine Learning. arXiv:2212.09720

[2] Dettmers T, Lewis M, Belkada Y, Zettlemoyer L (2022) LLM.int8(): 8-bit matrix multiplication for transformers at scale. In: Advances in Neural Information Processing Systems 35

[3] Zhou C et al (2026) Task-stratified knowledge scaling laws for post-training quantized large language models. In: Findings of the Association for Computational Linguistics: ACL 2026, pp 23268--23285. https://doi.org/10.18653/v1/2026.findings-acl.1165

[4] Zhong M, Wang G, Chuang YN, Zou N (2025) Quantized can still be calibrated: a unified framework to calibration in quantized large language models. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics, pp 30503--30517. https://doi.org/10.18653/v1/2025.acl-long.1473

[5] Guo C, Pleiss G, Sun Y, Weinberger KQ (2017) On calibration of modern neural networks. In: Proceedings of the 34th International Conference on Machine Learning

[6] Bouguerra A, Montoya D, Gomez-Villa A, Mraidha C, Arnez F (2026) Less precise can be more reliable: a systematic evaluation of quantization's impact on VLMs beyond accuracy. In: Proceedings of the International Conference on Machine Learning. arXiv:2509.21173

[7] Tamiosso GL, Nunes RO, Balreira DG (2026) Evaluating small language models for English-to-Portuguese translation: impact of model scale and quantization. In: Proceedings of the 17th International Conference on Computational Processing of Portuguese (PROPOR 2026), pp 943--952

[8] Cobbe K et al (2021) Training verifiers to solve math word problems. arXiv:2110.14168

[9] Clark P et al (2018) Think you have solved question answering? Try ARC, the AI2 reasoning challenge. arXiv:1803.05457

[10] Zhang X, Zhao J, LeCun Y (2015) Character-level convolutional networks for text classification. In: Advances in Neural Information Processing Systems 28

[11] Hendrycks D et al (2021) Measuring massive multitask language understanding. In: Proceedings of the International Conference on Learning Representations