Quantized Reasoning Models Think They Need to Think Longer, but They Do Not

A token-level look at why aggressive PTQ makes reasoning models overthink.

Sanae Lotfi[1], Polina Kirichenko[1], Steven Li[2], Zechun Liu[2]
[1]FAIR at Meta [2]Meta AI

Paper

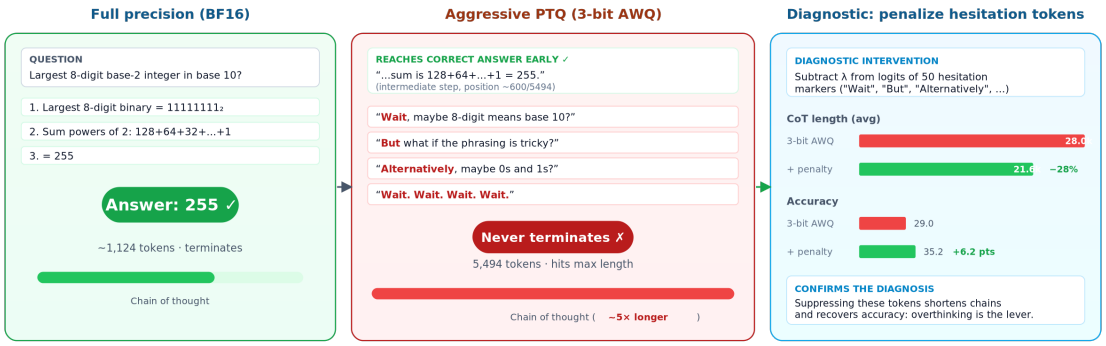


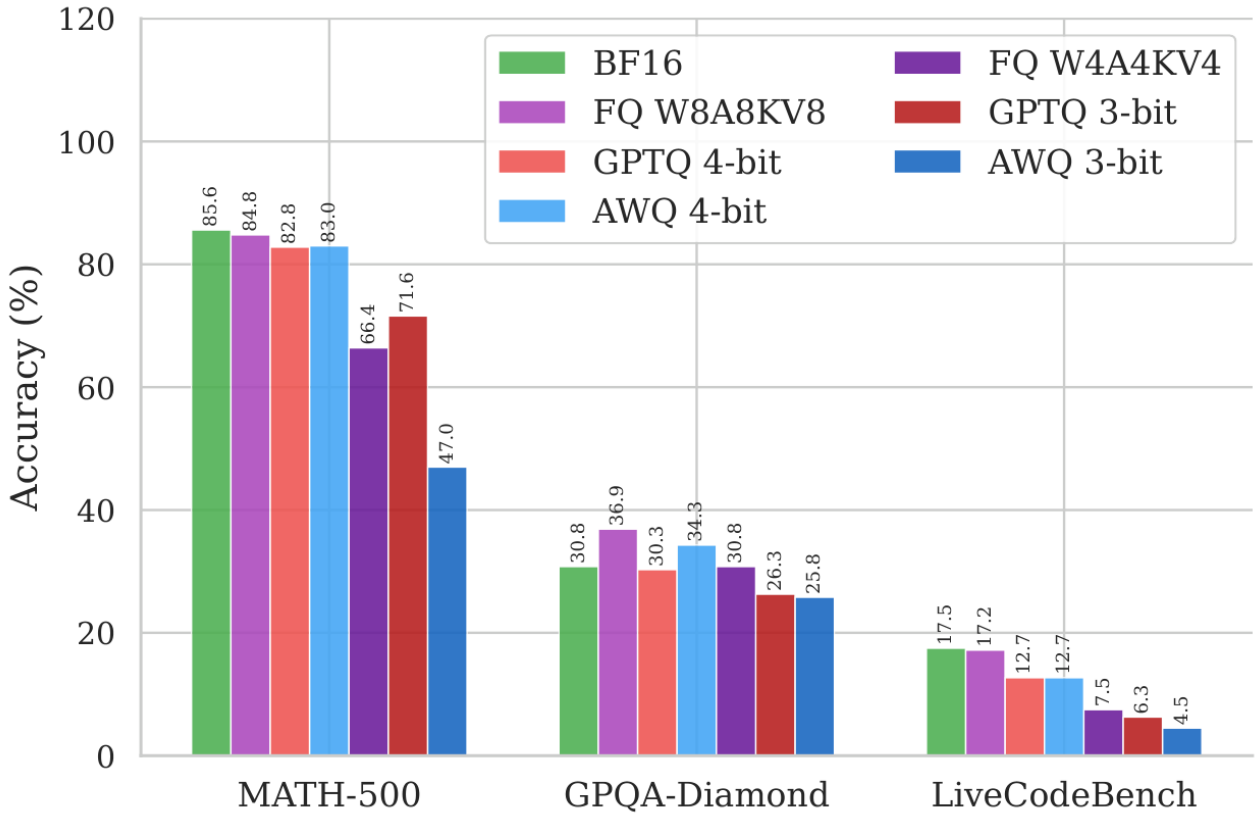
Figure 1. Quantization exacerbates overthinking in reasoning models. Left: The BF16 model reaches 255 and commits. Middle: The 3-bit AWQ model reaches 255 too — then opens new branches ("Wait... but... alternatively...") and never terminates. Right: Suppressing 50 hand-curated hesitation tokens at decode time confirms the diagnosis: chains shrink (−28%) and accuracy improves (+6.2 points) on MATH-500 under 3-bit AWQ.

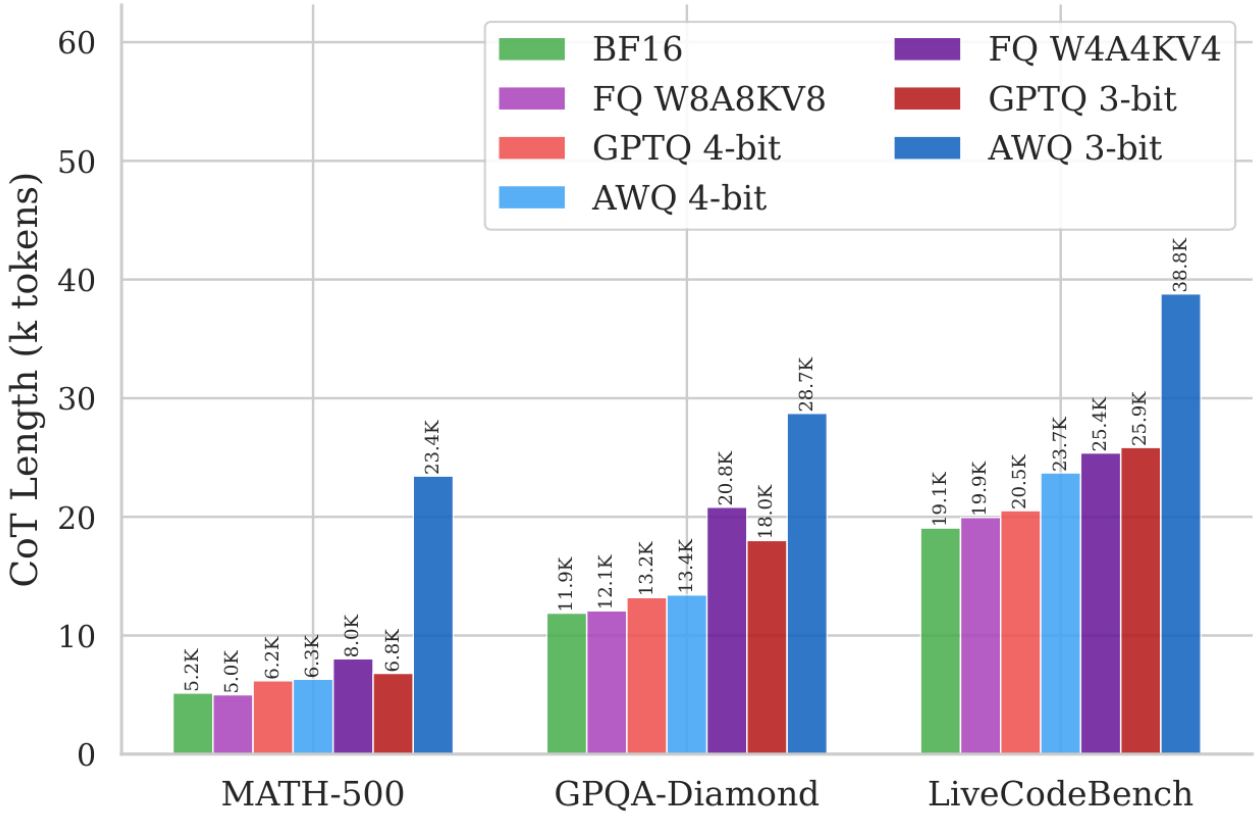
TL;DR. Aggressive post-training quantization (PTQ) doesn't just degrade reasoning models — it makes them **overthink**. In up to **52%** of 3-bit AWQ failures, the quantized model reaches the correct intermediate answer but spirals into hesitation tokens ("Wait", "But", "Alternatively") and never commits. Token-level KL divergence between the BF16 and quantized models concentrates exactly on these tokens, at exactly the decoding positions where BF16 is already uncertain.

Aggressive PTQ: lower accuracy, longer chains

We evaluate five reasoning models (1.5B–32B), three PTQ methods (GPTQ, AWQ, FlatQuant), and five benchmarks. Mild quantization is essentially free, but

pushing to 3-bit weights or W4A4KV4 hurts in both directions at once — accuracy drops and the chain of thought blows up.



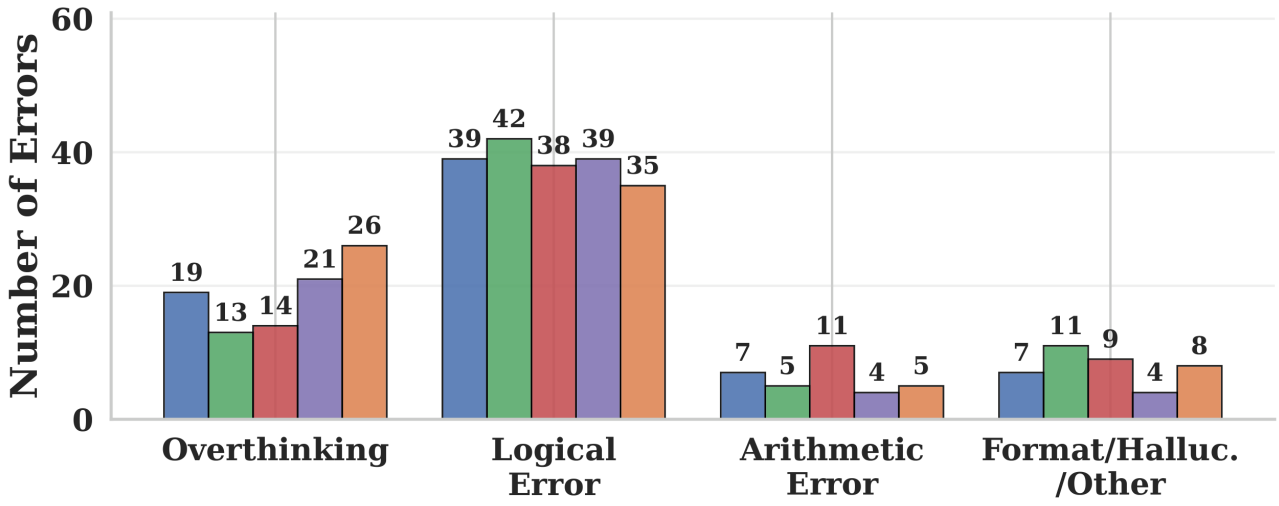


DeepSeek-R1-Distill-Qwen-1.5B. On MATH-500, 3-bit AWQ drops accuracy from **85.6→47.0%** while inflating CoT from **5.2K→23.4K** tokens (4.5×).

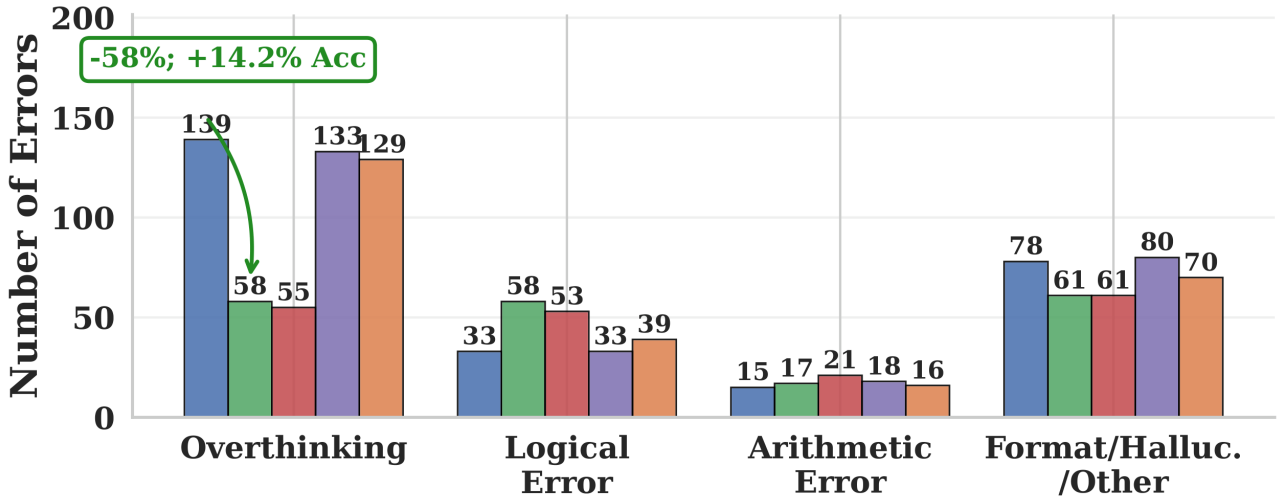
Across all 28 model×quantization pairs, the Spearman correlation between accuracy loss and CoT length increase is **ρ = −0.73**. The models that lose the most accuracy also generate the longest chains — suggesting the extra reasoning isn't a side effect of failure, but a cause of it.

The failure mode is overthinking, not capability loss

We hand-annotated MATH-500 failures and used GPT-5 (calibrated to >95% agreement) to scale the labels into four buckets: **overthinking** (reaches correct answer mid-trace then talks itself out of it), **logical error**, **arithmetic error**, and **other**.



BF16: 72 errors, 26% overthinking.

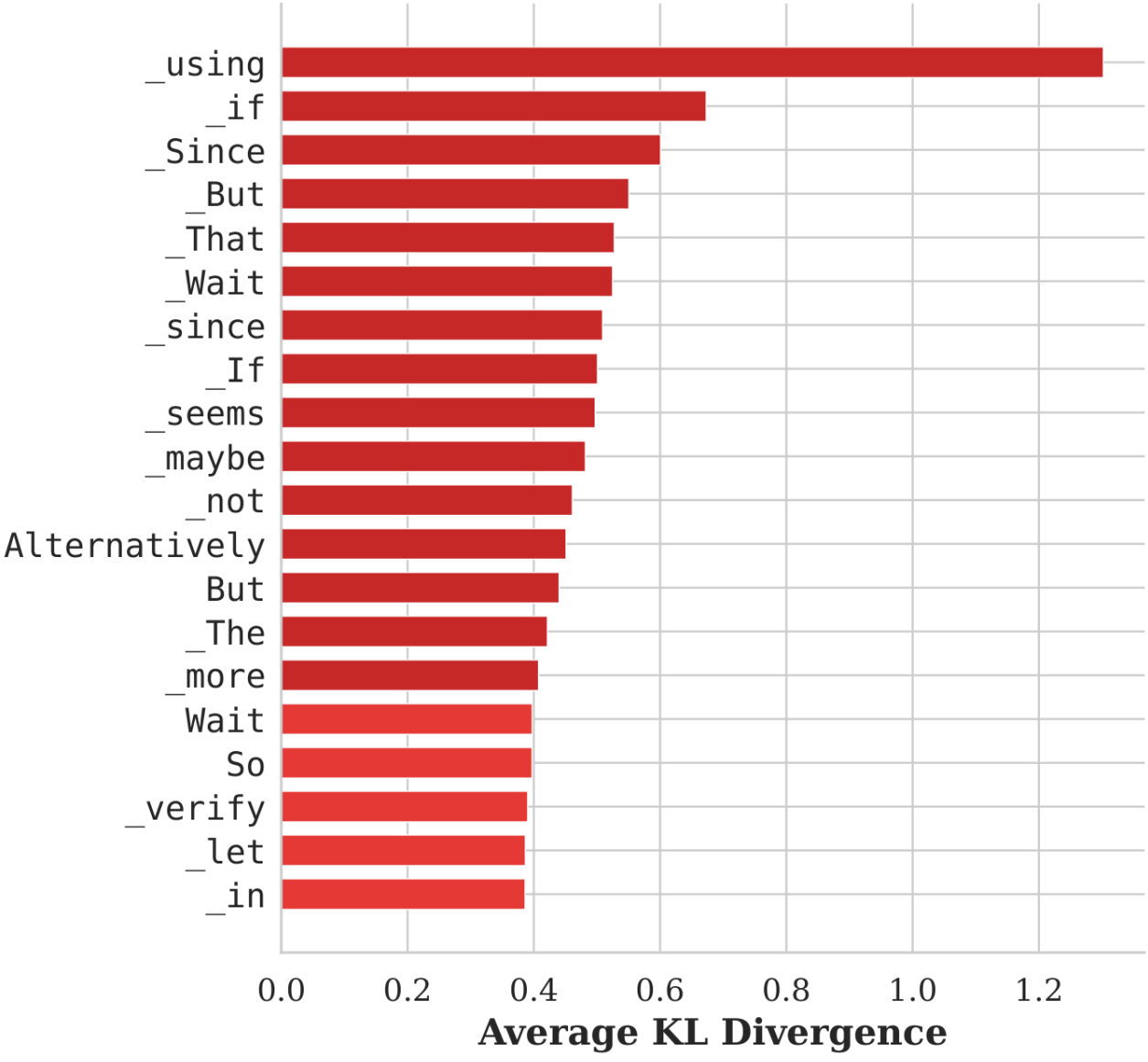


3-bit AWQ: 265 errors, **52% overthinking** (139 vs. 19 = **7.3×** the absolute count).

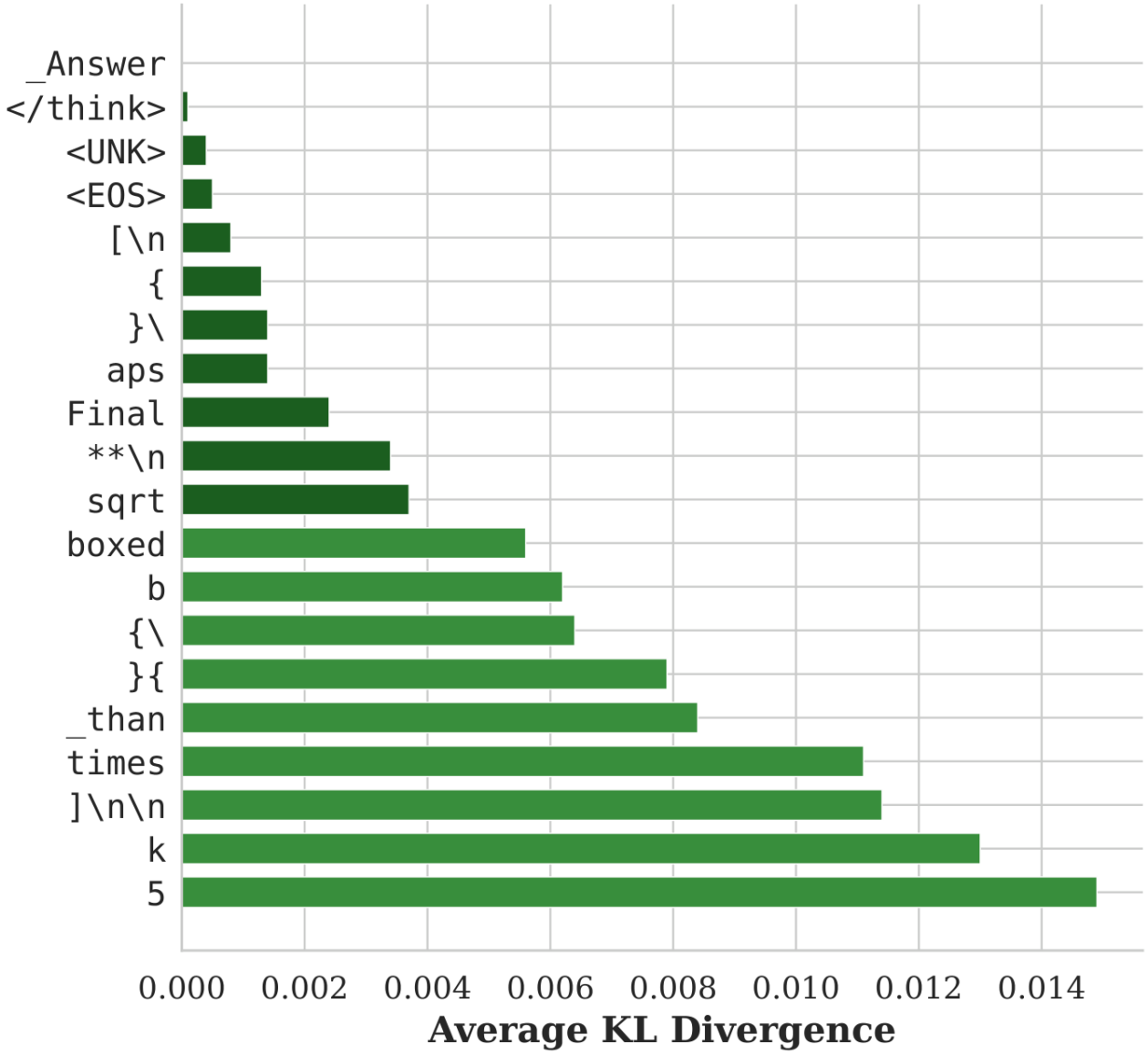
Why this rules out “quantization makes the model dumber”. If quantization were a uniform capability hit, all four error categories would grow proportionally. Instead, **overthinking inflates disproportionately** while arithmetic and logical errors stay roughly flat. The quantized model isn't losing its math — it's losing its ability to stop.

Quantization hits hesitation tokens at uncertain positions

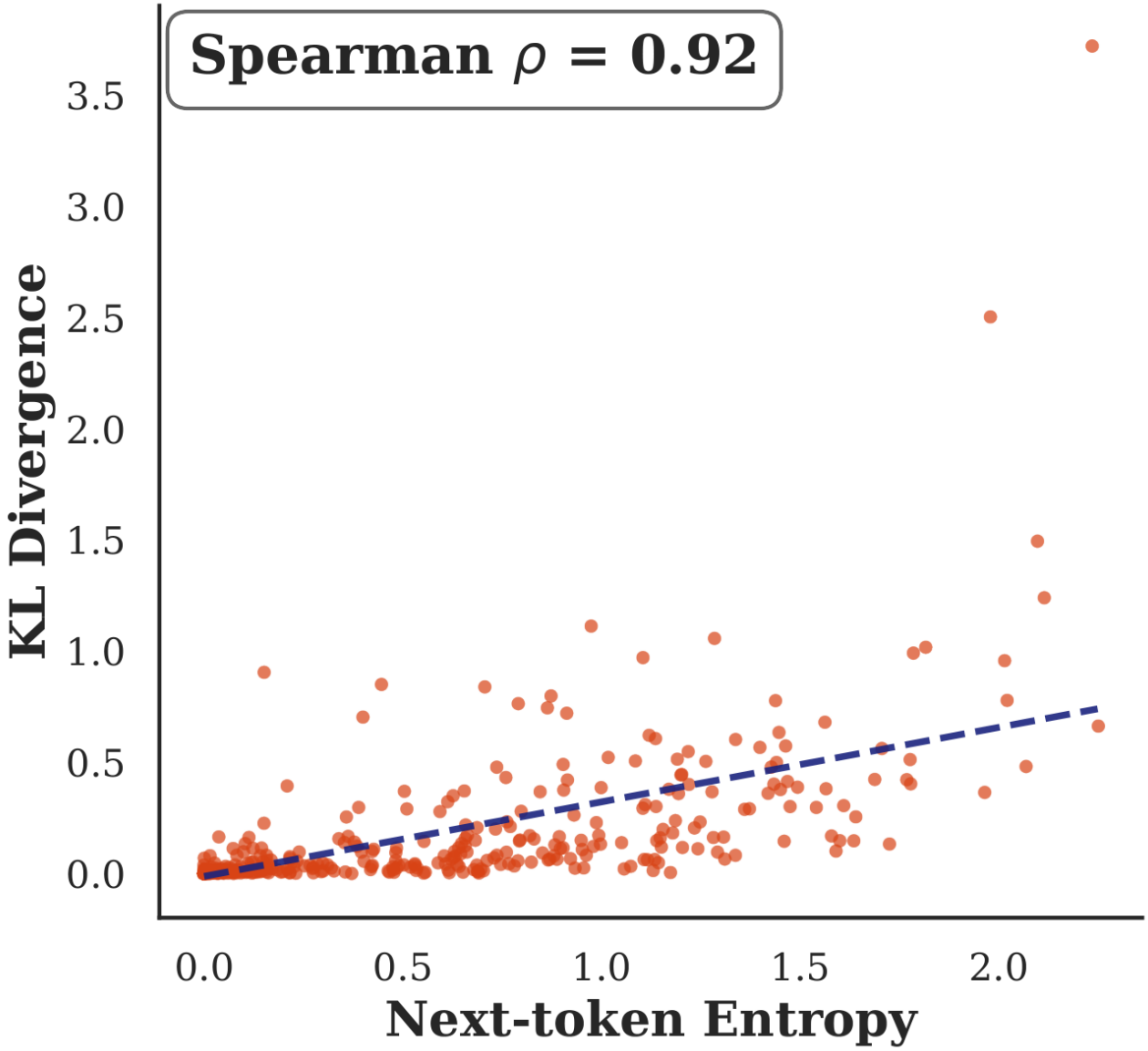
To localize where the two models diverge, we run BF16 and 3-bit AWQ on the same MATH-500 prompts under identical generation prefixes and measure per-position KL divergence $D_{KL}(p_t \| q_t)$, then associate each KL value with the token the quantized model actually sampled.



Highest-KL tokens are **hesitation** markers.



Lowest-KL tokens are **math/formatting**.



KL vs. BF16 next-token entropy: **ρ = 0.92**.

Three things fall out:

- **High-KL tokens are hesitation/branching** (“Wait”, “But”, “Alternatively”, “maybe”).
- **Low-KL tokens are mathematical/formatting** — quantization barely changes how the model samples digits, operators, or LaTeX.
- **Position-level KL tracks BF16 next-token entropy almost perfectly** (ρ = 0.92). Quantization noise has the most leverage precisely where the BF16 model was already uncertain — and at those high-entropy positions, hesitation tokens are 2–4× more likely to appear in the top-20 than at low-entropy ones.

Putting it together. At high-entropy positions, (1) the logit margin between the top token and a marginal “Wait” is small, and (2) hesitation tokens are already crowding the top of the distribution. Quantization noise nudges the sample over the line — the model opens a new reasoning branch and overwrites the answer it had.

Confirming the diagnosis

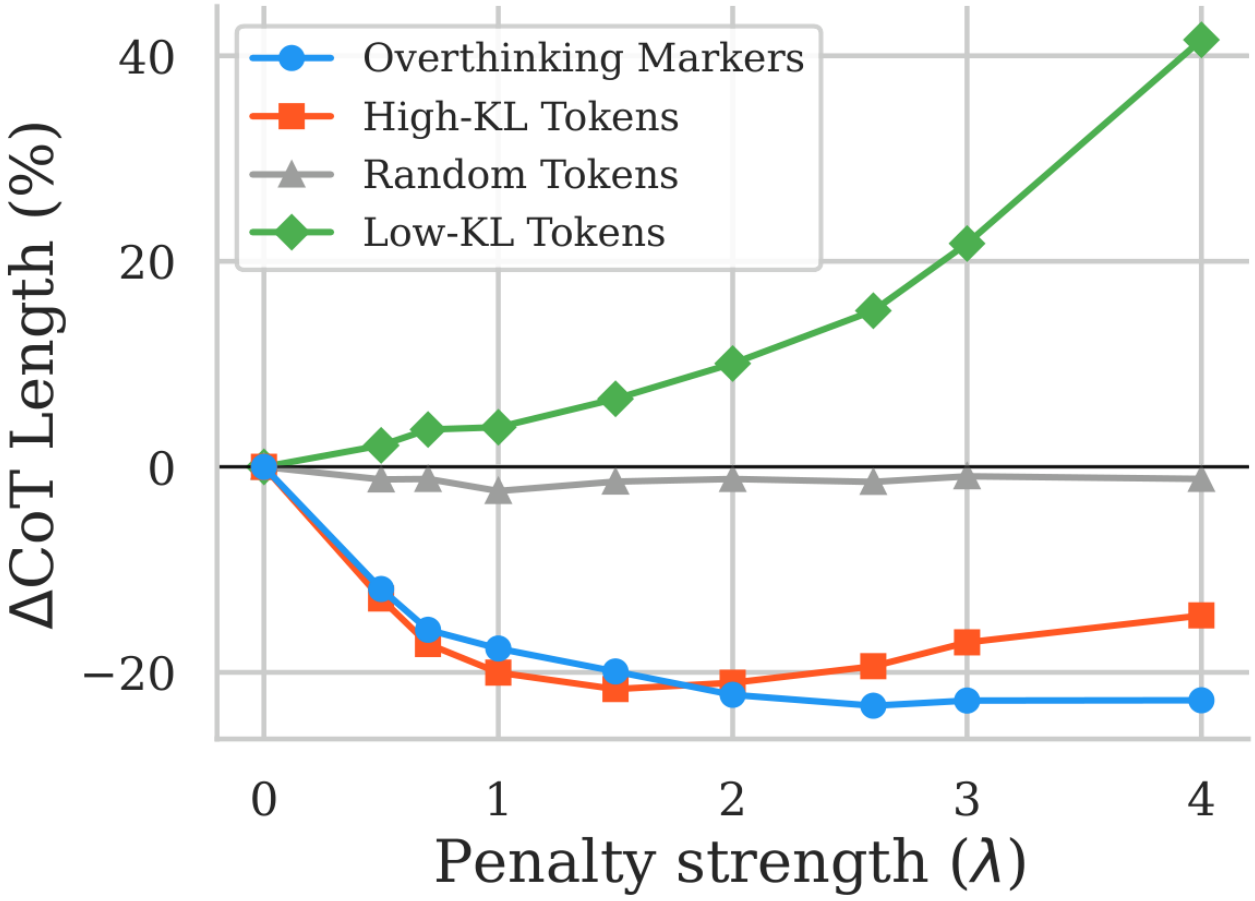
If hesitation tokens at high-entropy positions really drive the failure, suppressing them at decode time should specifically shorten chains, reduce the overthinking error bucket, and leave the other categories alone. We hand-curate $|S| = 50$ overthinking markers and subtract a fixed penalty $\lambda > 0$ from their logits at every decoding step:

$$z'_t(v) = z_t(v) - \lambda \quad \text{if } v \in S, \qquad \text{otherwise } z_t(v)$$

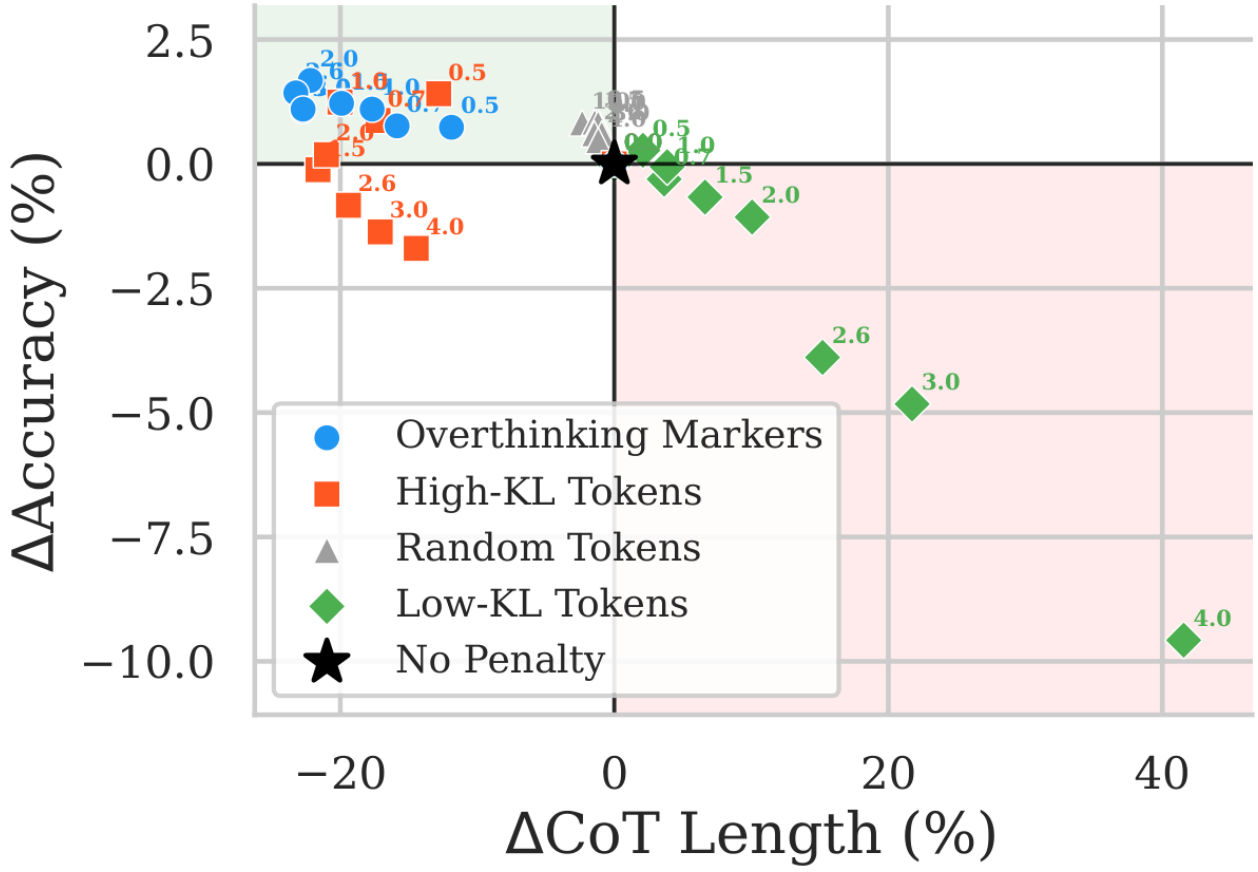
Diagnostic, not a method

We use this penalty as a **diagnostic**, not a proposed inference method. If our diagnosis is right, suppressing exactly these tokens should produce a specific, predictable change — and suppressing other lists (random, low-KL math) should produce nothing or the opposite. Both controls are run below.

Sweeping $\lambda \in [0.5, 4.0]$ across four token lists of equal size (overthinking markers, top-KL, low-KL, random) on Qwen-1.5B, averaged over 6 quantization configs and 5 benchmarks:



CoT length vs. λ.



Accuracy–length Pareto frontier across all λ.

Overthinking markers (our list): every λ shortens CoT by **12–23%** while preserving or improving accuracy. Owns the upper-left of the Pareto frontier.

Random tokens: negligible effect — clustered around the no-penalty baseline.

Low-KL math tokens: catastrophic — CoT grows by up to +41% and accuracy drops by up to −9.5%. Suppressing the tokens that compute the answer destroys reasoning.

A symmetric sanity check: boosting the same overthinking markers (negative λ) inflates CoT by up to **+445%** and drops accuracy by up to **−34%**. Random and low-KL tokens stay flat. The lever is real, and it points in both directions.

The same effect generalizes across all five models and all PTQ settings: the penalty consistently reduces CoT length by 4.1–28.0% on average while preserving or improving accuracy. On MATH-500 under 3-bit AWQ, overthinking errors fall from 139 to 58 (**−58%**), without inflating the other error categories — exactly what the diagnostic predicts.

Takeaways

- **A specific characterization of how PTQ degrades reasoning models.** Not a generic capability hit — an inflation of overthinking errors (up to 7.3× in absolute count on MATH-500 under 3-bit AWQ).
- **A token-level localization.** KL divergence between BF16 and quantized output distributions concentrates on hesitation tokens at high-entropy positions ($\rho = 0.92$ between position-level KL and BF16 next-token entropy).
- **A diagnostic that confirms the story.** A 50-token logit penalty cuts CoT length by 12–23% while preserving or improving accuracy; random and low-KL controls do not.
- **A reusable analysis recipe.** Token-level KL between a reference model and a compressed variant could serve as a general diagnostic for which aspects of generation are most vulnerable to compression — whether that's quantization, pruning, distillation, or low-rank approximation.

Caveats. S is hand-curated for English reasoning models; λ is fixed per decoding step; and our evaluation is restricted to math, coding, and science. Whether the same overthinking signature appears in agentic or open-domain reasoning is open.

BibTeX

```
@article{lotfi2026quantized,
  title  = {Quantized Reasoning Models Think They Need to Think Longer, but They Do No
  author = {Lotfi, Sanae and Kirichenko, Polina and Li, Steven and Liu, Zechun},
  year   = {2026},
}
```