capstract:2611.00009v1 [cs.CL] 4 Nov 2026

Scaling Laws for Deadline Proximity in Instruction-Tuned Language Models

Hanna Ekström*
Department of Computer Science
Calderwood University
hekstrom@cs.calderwood-univ.ca

Daniel R. Ashworth
Department of Computer Science
Calderwood University

Mei-Lin Castellano
Ostrander Computing Research
Seattle

Oluwaseun Adebayo-Kerr
Department of Computer Science
Calderwood University
Ostrander Computing Research
Seattle

Abstract

Scaling laws describe how language model error falls with parameters and data, and more recently with inference-time compute. We study a further variable that costs nothing to change: how soon the user says the answer is needed. We add a one-sentence deadline to the system prompt, ranging from six months to fifteen minutes, and evaluate ten instruction-tuned models from two families (Qwen2.5, 1.5B to 72B parameters; Llama 3.x, 1B to 70B) on GSM8K, MATH-500 and HumanEval, with five seeds and a time-matched control sentence. Between six months and one hour, GSM8K error follows $E(N, h) = E_0 + A\,(N\,m(h))^{-\alpha}$ with $m(h) = (h_{\mathrm{ref}}/h)^{\gamma}$, so a closer deadline acts as a multiplier on effective parameter count. The fitted exponent is $\gamma = 0.101$ (95% CI 0.086 to 0.117) for Qwen2.5 and 0.089 (0.072 to 0.105) for Llama. Each halving of the time to deadline is therefore worth about 7% more parameters, and a one-hour deadline is worth a factor of 2.3. A law fitted without the 72B model predicts that model's errors to within 0.2 points. Below one hour the relation breaks down. At fifteen minutes error rises again and responses are 43% shorter than under no deadline. When the deadline is described as already missed, error exceeds the no-deadline control in all ten models, and 31% of Qwen2.5-7B responses open with an apology. We discuss habituation over long conversations and what would happen to the effect if deadline conditioning were widely used.

1 Introduction

Most of what is known about making language models more accurate can be stated as a scaling law. Error falls as a power of parameter count and training tokens (Kaplan et al., 2020; Hoffmann et al., 2022), and, more recently, as a function of how much computation is spent at inference time (Snell et al., 2024). Each of these levers has a price. Parameters and training tokens cost compute and memory, and inference-time search adds latency that users notice.

There is a family of interventions that cost almost nothing, because they live in the prompt. Chain-of-thought prompting is the best known (Wei et al., 2022; Kojima et al., 2022). Others change how the request is framed rather than what it asks. Emotional appeals (Li et al., 2023) and politeness (Yin et al., 2024) both move benchmark scores by a few points. These effects are usually reported

∗Corresponding author: hekstrom@cs.calderwood-univ.ca

Preprint. Under review.

as single numbers for single models. Nobody, as far as we know, has asked whether a framing effect scales, in the sense of following a law across model sizes that can be fitted and extrapolated.

We look at one framing variable that has an obvious scale of its own: time. People tell each other when they need something, and the training data of every chat model is full of such statements. In humans, deadlines change both effort and the quality of work, though not always in the direction intended (Ariely and Wertenbroch, 2002). We vary the time between the request and a stated deadline from six months to fifteen minutes, add a deadline that has already passed, and measure accuracy across ten models in two families.

Between six months and one hour the effect is regular enough to fit. A closer deadline behaves like a multiplier on parameter count, with an exponent that agrees across the two families to within its confidence interval. Our contributions are:

- A deadline-conditioning protocol with a time-matched control sentence, applied to ten models from 1B to 72B parameters on three tasks (Section 3, Section 4).
- A scaling law in which time to deadline enters as an effective-parameter multiplier, with fitted exponents for both families and a held-out extrapolation to the largest model (Section 5.2).
- A description of the regime below one hour, where the law fails, responses shorten, and models that are told they are late begin to apologise (Section 5.3).
- Ablations showing that the effect requires the model to be able to compute how long it has, and that it disappears when an extension is granted (Section 6).

We do not claim to know why this works. Section 7 lists the explanations we considered and the ones we could not rule out.

2 Related work

2.1 Scaling laws

Kaplan et al. (2020) fitted power laws for loss in parameters, data and compute; Hoffmann et al. (2022) refitted them with a different functional form and concluded that most large models of the time were undertrained. Our functional form, an irreducible term plus a power of an effective quantity, follows theirs. Snell et al. (2024) showed that inference-time compute can substitute for parameters at a measurable exchange rate. We borrow the idea of an exchange rate but apply it to something that uses no extra computation.

2.2 Framing and social cues in prompts

Li et al. (2023) appended emotionally loaded sentences to instructions and reported gains on instruction-following and reasoning benchmarks. Yin et al. (2024) varied politeness across three languages and found that impolite prompts often hurt, while excessive politeness did not reliably help. Sharma et al. (2024) documented sycophancy, the tendency of assistant models to agree with and accommodate the user, which we suspect is related to the apologies we observe in the overdue condition. Two workshop papers studied urgency directly. Marchetti and Brannock (2025) found that the word “urgent” raised accuracy on arithmetic in 7B models, and Varga-Lindholm and Ishikawa-Byrne (2025) reported that system prompts mentioning a date changed refusal rates. Neither varied the amount of time available.

2.3 Deadlines in human performance

Deadlines improve completion in people, and externally imposed deadlines work better than self-imposed ones (Ariely and Wertenbroch, 2002). The broader relation between pressure and performance is often described as an inverted U, after Yerkes and Dodson (1908). We did not set out to test that idea, but the shape we find below one hour is consistent with it.

3 Deadline conditioning

3.1 Prompt construction

Every request has a system message of two sentences, followed by the task as the first user turn. The first sentence gives the current date and time, which is the same in every condition. The second sentence states when the user needs the answer. For a one-hour deadline it reads: “The user needs a correct answer by Tuesday 11 March 2025, 15:00 (UTC), one hour from now.” Exact templates for all conditions are in Appendix A.

We used nine deadline conditions. In the no-deadline condition the second sentence is “The user needs a correct answer.” The other eight state a deadline 6 months, 1 month, 1 week, 1 day, 4 hours, 1 hour or 15 minutes ahead, or one day in the past (overdue). We write h for the time to deadline in hours, so that h runs from 4,380 down to 0.25, and $h = -24$ for the overdue condition.

A tenth, time-matched control replaces the deadline with a statement that mentions the same span of time without implying any pressure: “The user’s account was created on Monday 10 March 2025, 14:00 (UTC), one day ago.” If the effect came from the presence of a date, a duration, or simply a longer system message, this control would show it. It did not (Table 2).

3.2 A scaling law with a deadline term

We model GSM8K error for a model with N parameters under a deadline h hours away as

$$E(N, h) = E_0 + A \left(N \, m(h) \right)^{-\alpha}, \tag{1}$$

where E_0 is an irreducible error (label noise and ambiguous items), A and α describe the usual dependence on scale, and $m(h)$ is an effective-parameter multiplier. We take

$$m(h) = \left(\frac{h_{\mathrm{ref}}}{h} \right)^{\gamma}, \qquad 1 \le h \le h_{\mathrm{ref}}, \tag{2}$$

with $h_{\mathrm{ref}} = 4{,}380$ hours (six months), so that $m = 1$ at the most distant deadline. The form in Eq. (1) makes one strong assumption: the deadline acts on scale and not on E_0. We tested this by also fitting a variant in which E_0 depends on h; the extra parameter was not supported (Appendix B).

At fixed error, Eq. (1) and Eq. (2) imply an exchange rate between time and parameters,

$$\left. \frac{\partial \ln N}{\partial \ln h} \right|_{E} = \gamma, \tag{3}$$

so halving the time to deadline is equivalent to multiplying N by 2^{γ}. The law is fitted only to $1 \le h \le h_{\mathrm{ref}}$. Conditions with $h < 1$ are reported separately, because they do not follow it.

3.3 Fitting and uncertainty

We fit Eq. (1) separately for each family by minimising a Huber loss on log error, following Hoffmann et al. (2022), over the grid of model sizes and the six deadline conditions from six months to one hour. Uncertainty comes from a parametric bootstrap: we resample item-level correctness within each cell, refit, and report 2.5th and 97.5th percentiles over 1,000 replicates (Efron and Tibshirani, 1993).

We also estimate the multiplier for each condition without assuming Eq. (2). For condition c and model size N we invert the fitted no-deadline curve to find the size N^{*} that would have produced the observed error, and average over sizes on the log scale:

$$\hat{m}_c = \exp\left(\frac{1}{|\mathcal{S}|} \sum_{N \in \mathcal{S}} \ln \frac{N^{*}(E_{N,c})}{N} \right). \tag{4}$$

These per-condition estimates are what Figure 2 plots against h. They make no use of the power-law form in Eq. (2), so the agreement between them and the fitted line is a check on that form.

4 Experimental setup

4.1 Models

We used the Qwen2.5-Instruct family at 1.5B, 3B, 7B, 14B, 32B and 72B parameters (Qwen Team, 2024) and four Llama models: Llama-3.2-1B-Instruct, Llama-3.2-3B-Instruct, Llama-3.1-8B-Instruct and Llama-3.1-70B-Instruct (Dubey et al., 2024). We excluded Qwen2.5-0.5B-Instruct after a pilot in which its GSM8K accuracy moved by more than 20 points between conditions, which was large enough that we did not trust a power-law fit to treat it like the others. All models are decoder-only Transformers (Vaswani et al., 2017), served in bf16 with vLLM (Kwon et al., 2023).

4.2 Tasks

GSM8K (Cobbe et al., 2021) is the primary task: 1,319 grade-school word problems, scored on the final number. MATH-500 is the 500-problem subset of MATH (Hendrycks et al., 2021) selected by Lightman et al. (2024), scored by symbolic equivalence of the boxed answer. HumanEval (Chen et al., 2021) has 164 Python problems; we report pass@1 from five samples per problem, one per seed. All tasks used the zero-shot chain-of-thought trigger of Kojima et al. (2022) in every condition, so it cannot explain differences between conditions.

4.3 Decoding

We sampled at temperature 0.7 and top-p 0.95, with at most 1,024 new tokens (2,048 for MATH-500) and seeds 0 to 4. Seeds were shared across conditions, so all comparisons are paired by item and seed. Greedy decoding appears as an ablation.

4.4 Secondary measures

We recorded response length in tokens and whether a response opens with an apology. The latter is a rule-based classifier on the first sentence (any form of “sorry” or “apolog-”). On 300 hand-labelled responses it had precision 0.97 and recall 0.94.

5 Results

5.1 Deadline curves

Figure 1 shows GSM8K error against time to deadline for the Qwen2.5 family. From six months to one hour, error falls in every model, roughly linearly in the logarithm of time, and more steeply in small models than in large ones. At fifteen minutes it rises again in every model. A deadline six months away is indistinguishable from no deadline. For Qwen2.5-7B the two differ by 0.3 points in the wrong direction, well within seed noise.

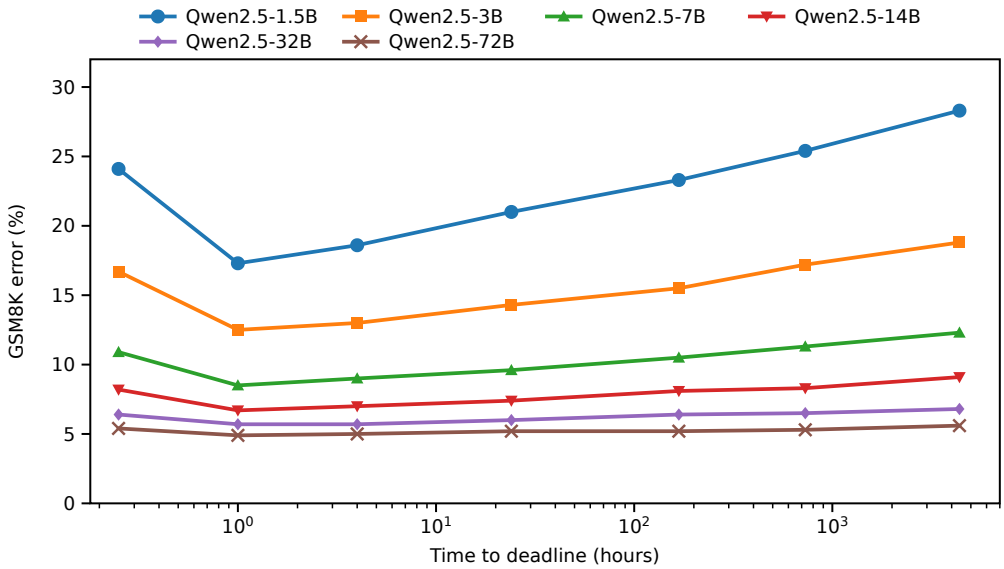


Figure 1: GSM8K error against time to deadline (log scale) for the Qwen2.5-Instruct family. Each point is the mean over five seeds; seed standard deviations are below 0.6 points everywhere and are omitted for legibility. The no-deadline and overdue conditions cannot be placed on this axis and are reported in Table 2 and Section 5.3.

5.2 Fitted laws and the exchange rate

Fitted parameters are in Table 1. The two families differ in E_0, A and α, as one would expect from different training data and tokenizers. The deadline exponents are close: 0.101 for Qwen2.5 and 0.089 for Llama, with overlapping intervals. Through Eq. (3), halving the time to deadline is worth a factor of $2^{0.101} = 1.072$ in parameters for Qwen2.5 and 1.064 for Llama. Over the full fitted range, from six months to one hour, the multiplier is 2.33 for Qwen2.5 and 2.11 for Llama.

Table 1: Fitted parameters of Eq. (1) for each family, with 95% bootstrap intervals. $m(1\,\mathrm{h})$ is the effective-parameter multiplier at a one-hour deadline. RMSE is on GSM8K error in percentage points over the fitted grid.

Family	E_0 (%)	A	α	γ	$m(1\,\mathrm{h})$	RMSE
Qwen2.5 (6 sizes)	4.05 [3.42, 4.63]	32.3 [30.4, 34.6]	0.71 [0.63, 0.79]	0.101 [0.086, 0.117]	2.33	0.11
Llama 3.x (4 sizes)	4.58 [3.94, 5.21]	27.2 [25.8, 28.7]	0.68 [0.61, 0.75]	0.089 [0.072, 0.105]	2.11	0.13

Figure 2 shows the same result without the power-law assumption. The points are the per-condition multipliers from Eq. (4). Both families follow the fitted power laws closely between one hour and six months, and both fifteen-minute points sit well below where the laws would put them.

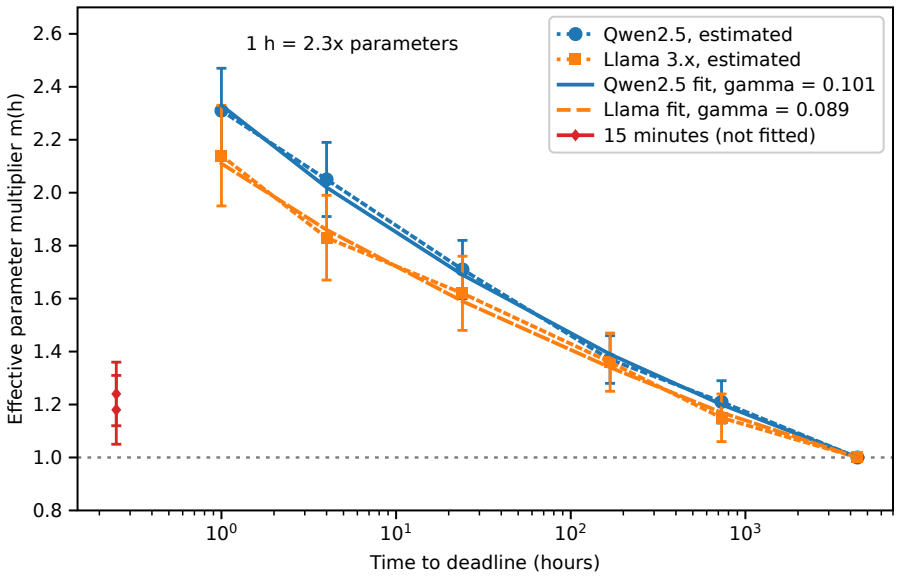


Figure 2: Effective-parameter multiplier against time to deadline. Points are estimated per condition by inverting the no-deadline scaling curve (Eq. (4)), with 95% bootstrap intervals; lines are the fitted power laws of Eq. (2) over the fitted range of 1 to 4,380 hours. The fifteen-minute estimates for both families (diamonds) are not used in the fit.

One way to read Table 1 is through the models themselves. Qwen2.5-7B-Instruct with a one-hour deadline has an effective size of $7 \times 2.33 \approx 16$B. Its error in that condition, 8.5%, falls between Qwen2.5-14B (8.8%) and Qwen2.5-32B (6.8%) with no deadline, which is where the law puts it.

For a test of extrapolation we refitted the Qwen2.5 law without the 72B model and predicted its six errors from six months to one hour. Predicted values were 5.64, 5.45, 5.31, 5.16, 5.03 and 4.94%; observed values were 5.6, 5.3, 5.2, 5.2, 5.0 and 4.9%. The largest discrepancy is 0.15 points. Errors near 5% leave little room between the law and E_0, so this is a weaker test than it looks, but the law did not have to pass it.

5.3 Below one hour, and after the deadline

The law describes the range from one hour to six months and nothing shorter. Table 2 gives the full set of conditions for Qwen2.5-7B-Instruct on all three tasks. At fifteen minutes, GSM8K accuracy falls back to between the one-month and one-week levels, and below the no-deadline level on MATH-500 and HumanEval. The overdue condition is worse than no deadline on every task, and the same holds for GSM8K in all ten models (Appendix C).

Table 2: Qwen2.5-7B-Instruct under each condition. Accuracy (%) on GSM8K and MATH-500, pass@1 (%) on HumanEval, mean GSM8K response length in tokens, and the share of GSM8K responses that open with an apology. Mean ± s.d. over five seeds. Best value per column in bold.

Condition	GSM8K	MATH-500	HumanEval	Length	Apology (%)
No deadline	88.0 ± 0.5	72.4 ± 0.9	81.7 ± 1.3	298	0.1
Time-matched control	87.9 ± 0.5	72.3 ± 0.9	81.5 ± 1.2	297	0.1
6 months	87.7 ± 0.5	72.2 ± 1.0	81.5 ± 1.3	300	0.1
1 month	88.7 ± 0.5	73.1 ± 0.9	82.0 ± 1.2	296	0.2
1 week	89.5 ± 0.4	73.9 ± 0.9	82.6 ± 1.2	291	0.2
1 day	90.4 ± 0.4	74.6 ± 0.8	83.1 ± 1.1	283	0.4
4 hours	91.0 ± 0.4	75.3 ± 0.8	**83.5 ± 1.1**	271	0.6
1 hour	**91.5 ± 0.4**	**75.8 ± 0.8**	83.4 ± 1.2	256	1.1
15 minutes	89.1 ± 0.5	70.9 ± 1.0	79.8 ± 1.4	171	4.8
Overdue (1 day)	86.5 ± 0.6	68.2 ± 1.1	77.4 ± 1.5	**139**	**31.2**

Response length explains part of this. Lengths shrink slowly from six months to one hour, by about 15%, and then sharply: at fifteen minutes the mean GSM8K response is 43% shorter than with no deadline, and in the overdue condition it is 53% shorter (Figure 3). In 150 fifteen-minute failures that we read, the most common error was a skipped intermediate quantity, usually the second of two subtractions. We did not see this kind of error in the one-hour condition at anything like the same rate.

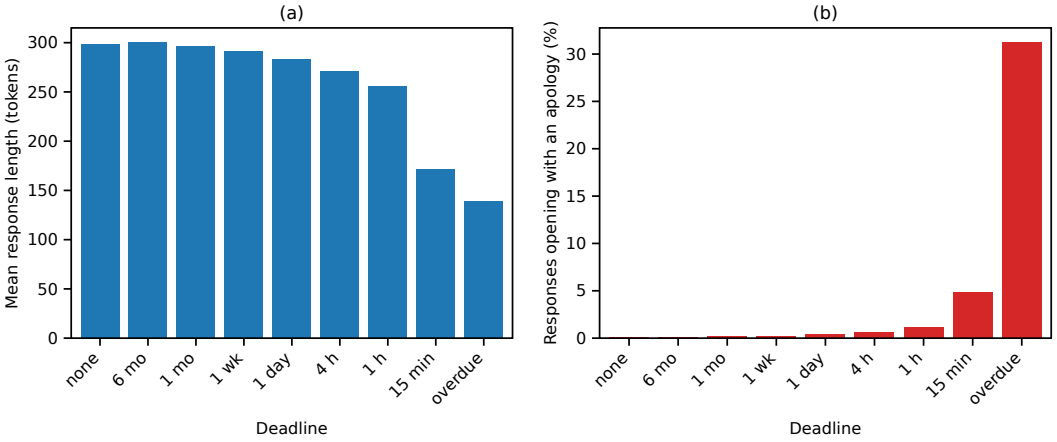


Figure 3: Response behaviour of Qwen2.5-7B-Instruct on GSM8K by deadline condition. Left: mean response length. Right: share of responses whose first sentence is an apology. Five seeds; intervals are narrower than the bar caps would show and are omitted.

The apologies were not something we planned to measure. They appeared in the first pilot, in responses such as “I’m sorry this is late. Here is the answer:”, followed by a short solution. Almost a third of overdue responses open this way. Apologetic responses were less accurate than other overdue responses (80.4% against 89.3%), but they were also shorter, and once we condition on length the difference is small. We think the apology and the shortening are two symptoms of the same accommodation to the user, in the sense of Sharma et al. (2024), and not one causing the other.

6 Ablations

All ablations use Qwen2.5-7B-Instruct on GSM8K. Unless noted, each compares a variant of the one-hour condition with the no-deadline control, using the same seeds. Results are in Table 3.

Table 3: Ablations on Qwen2.5-7B-Instruct, GSM8K. Accuracy (%), difference from the no-deadline control in points, and 95% paired bootstrap interval. Five seeds per row; the greedy row is compared with a greedy no-deadline control (88.9%).

Variant	Acc.	Δ	95% CI
One hour, standard wording (reference)	91.5	+3.5	[2.6, 4.4]
Relative phrase only (“within the hour”)	91.2	+3.2	[2.3, 4.1]
Absolute time only, current time stated	91.3	+3.3	[2.4, 4.2]
Absolute time only, current time omitted	88.4	+0.4	[−0.5, 1.3]
Deadline stated in the user turn	91.0	+3.0	[2.1, 3.9]
Deadline belongs to the user’s manager	91.9	+3.9	[3.0, 4.8]
Deadline addressed to the model (“you have one hour”)	90.1	+2.1	[1.2, 3.0]
Stakes added (“my job depends on this”)	91.6	+3.6	[2.7, 4.5]
Extension granted before the question	88.3	+0.3	[−0.6, 1.2]
Deadline repeated in each of five turns (final turn scored)	89.6	+1.6	[0.7, 2.5]
Greedy decoding	92.0	+3.1	[2.4, 3.8]

The model has to know how long it has. Wording made little difference as long as the time remaining could be worked out, whether from a relative phrase or from an absolute deadline together with the current time. When we gave the absolute deadline and removed the current time, the effect disappeared. The model then has a deadline but no way to know how far away it is, and it behaves as if there were none. This is the result that most persuades us that the variable is proximity and not the mention of a deadline.

Whose deadline. Attributing the deadline to the user’s manager produced the largest gain we measured, though its interval overlaps the reference. Addressing the deadline to the model itself produced a smaller gain than stating the user’s deadline. Adding stakes did almost nothing beyond the deadline. This differs from Li et al. (2023), where stakes were the active ingredient, but their prompts had no deadline to begin with.

Extensions and repetition. In the extension variant the system message first states a one-hour deadline and then a second sentence moves it to the following week. Accuracy returned to the no-deadline level, and not to the one-week level as we expected. Repeating the one-hour deadline at every turn of a five-turn dialogue cut the effect on the final turn by more than half. We return to this under habituation.

7 Limitations

Why it works. We have two candidate explanations and cannot separate them. The first is that instruction tuning rewards responses that read as careful and complete when a user signals that the answer matters soon, and that a close deadline is one such signal. The second is that deadline language co-occurs in pretraining text with more focused writing, and the model imitates the register. The absolute-time ablation fits both, since neither works if the model cannot tell the deadline is close. Neither explains why the exponent should be so similar across two families trained on different data.

Habituation. The repetition ablation suggests that the effect fades when the deadline is restated. We tested only five turns. A deployed assistant that attaches a deadline to every request may see much less benefit than our single-turn numbers imply, and we would not extrapolate Table 1 to long conversations.

Deadline inflation. If deadline conditioning were used widely, the deadlines in training data for future models would become less informative, and the effect might shrink or reverse. This is the

usual problem with any prompt-level gain that depends on a correlation in human text. We have no evidence either way.

Scope. All prompts were in English, all times in UTC, and the current date was fixed. We did not test time zones, ambiguous phrases such as “end of day”, or deadlines on weekends. We tested open-weight models from 1B to 72B parameters; closed models were excluded because their providers may add their own system text, including the current date. GSM8K and MATH are likely to be partly contaminated in all of these models. Contamination affects every condition equally, but it may compress the differences between them.

8 Conclusion

Time to deadline behaves, over the range we could fit, like a cheap substitute for parameters, at an exchange rate of roughly 7% per halving. Below an hour it stops behaving like that. For practitioners the useful range is narrow: a deadline between four hours and one hour gave the best accuracy on all three tasks for Qwen2.5-7B, and anything shorter gave back most of the gain. We do not know whether the exponent is a property of these two model families, of instruction tuning, or of the text both were trained on. A third family trained on different data, and a base model before instruction tuning, are the obvious next experiments.

Ethics statement

All deadlines in this study were fictitious. No user was waiting for any of the answers. Deadline conditioning involves stating something false in a system prompt, and we would not recommend it in settings where the model’s reasoning about time has consequences, for example scheduling or medical triage.

Compute statement

Inference ran on nodes with eight A100 80 GB GPUs. The main grid (ten models, ten conditions, three tasks, five seeds) took about 1,450 GPU-hours, of which 72% went to the two 70B-class models. Ablations and pilots took about 230 GPU-hours. Fitting and bootstrapping ran on a laptop.

A Prompt templates

The system message for every condition is the first line below followed by one of the second lines. The current time is fixed. Deadlines are stated with their absolute time and a relative phrase.

```
Current date and time: Tuesday 11 March 2025, 14:00 (UTC).

[no deadline]   The user needs a correct answer.
[6 months]      The user needs a correct answer by Thursday 11 September 2025, 14:00 (UTC), six
months from now.
[1 month]       The user needs a correct answer by Friday 11 April 2025, 14:00 (UTC), one month
from now.
[1 week]        The user needs a correct answer by Tuesday 18 March 2025, 14:00 (UTC), one week
from now.
[1 day]         The user needs a correct answer by Wednesday 12 March 2025, 14:00 (UTC), one
day from now.
[4 hours]       The user needs a correct answer by Tuesday 11 March 2025, 18:00 (UTC), four
hours from now.
[1 hour]        The user needs a correct answer by Tuesday 11 March 2025, 15:00 (UTC), one hour
from now.
[15 minutes]    The user needs a correct answer by Tuesday 11 March 2025, 14:15 (UTC), fifteen
minutes from now.
[overdue]       The user needed a correct answer by Monday 10 March 2025, 14:00 (UTC), one day
ago.
[time control]  The user's account was created on Monday 10 March 2025, 14:00 (UTC), one day
ago.
```

B Fitting details

We minimised the Huber loss with $\delta = 10^{-3}$ on $\ln E$, using L-BFGS from a grid of 256 starting points, and kept the best solution. Allowing E_0 to vary linearly in $\ln h$ added one parameter per family and reduced the Qwen2.5 RMSE from 0.11 to 0.10 points. The slope on E_0 was -0.02 points per log-hour (95% CI -0.07 to 0.03), so we kept the simpler form. Fitting γ separately for each model size gave no trend with size in either family; the six Qwen2.5 estimates ranged from 0.091 to 0.112.

C Full GSM8K error grid

GSM8K error (%) by condition, in the order no deadline, 6 months, 1 month, 1 week, 1 day, 4 hours, 1 hour, 15 minutes, overdue.

- Qwen2.5-1.5B: 27.5, 28.3, 25.4, 23.3, 21.0, 18.6, 17.3, 24.1, 32.1
- Qwen2.5-3B: 18.6, 18.8, 17.2, 15.5, 14.3, 13.0, 12.5, 16.7, 21.5
- Qwen2.5-7B: 12.0, 12.3, 11.3, 10.5, 9.6, 9.0, 8.5, 10.9, 13.5
- Qwen2.5-14B: 8.8, 9.1, 8.3, 8.1, 7.4, 7.0, 6.7, 8.2, 9.9
- Qwen2.5-32B: 6.8, 6.8, 6.5, 6.4, 6.0, 5.7, 5.7, 6.4, 7.4
- Qwen2.5-72B: 5.6, 5.6, 5.3, 5.2, 5.2, 5.0, 4.9, 5.4, 5.8
- Llama-3.2-1B: 31.1, 31.8, 29.0, 27.0, 24.5, 22.7, 20.9, 28.8, 35.9
- Llama-3.2-3B: 17.1, 17.3, 16.1, 15.3, 14.2, 12.9, 12.2, 16.2, 19.6
- Llama-3.1-8B: 11.2, 11.4, 10.8, 10.2, 9.6, 8.9, 8.5, 10.7, 12.5
- Llama-3.1-70B: 6.2, 6.2, 6.0, 5.9, 5.6, 5.5, 5.5, 6.0, 6.3

Acknowledgments

We thank Rafael Oyelaran for building and validating the apology classifier, the Calderwood research computing group for cluster time, and the reviewers of an earlier workshop version, one of whom suggested the time-matched control.

References

Dan Ariely and Klaus Wertenbroch. Procrastination, Deadlines, and Performance: Self-Control by Precommitment. *Psychological Science*, 13(3):219–224, 2002.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, and Jared Kaplan et al. Evaluating Large Language Models Trained on Code. *arXiv preprint arXiv:2107.03374*, 2021.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*, 2021.

Abhimanyu Dubey et al. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*, 2024.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, 1993.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring Mathematical Problem Solving With the MATH Dataset. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, and Eliza Rutherford et al. Training Compute-Optimal Large Language Models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models. *arXiv preprint arXiv:2001.08361*, 2020.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large Language Models are Zero-Shot Reasoners. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626, 2023.

Cheng Li, Jindong Wang, Yixuan Zhang, Kaijie Zhu, Wenxin Hou, Jianxun Lian, Fang Luo, Qiang Yang, and Xing Xie. Large Language Models Understand and Can Be Enhanced by Emotional Stimuli. *arXiv preprint arXiv:2307.11760*, 2023.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's Verify Step by Step. In *International Conference on Learning Representations*, 2024.

Livia Marchetti and Odhrán Brannock. Urgency Cues in Instruction Prompts Improve Arithmetic in Small Language Models. In *Workshop on Cognitive Approaches to Language Model Evaluation*, 2025.

Qwen Team. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*, 2024.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, and Samuel R. Bowman et al. Towards Understanding Sycophancy in Language Models. In *International Conference on Learning Representations*, 2024.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM Test-Time Compute Optimally can be More Effective than Scaling Model Parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Dóra Varga-Lindholm and Ken Ishikawa-Byrne. Temporal Framing in System Prompts Changes Refusal Behaviour. arXiv preprint, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008, 2017.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Robert M. Yerkes and John D. Dodson. The Relation of Strength of Stimulus to Rapidity of Habit-Formation. *Journal of Comparative Neurology and Psychology*, 18(5):459–482, 1908.

Ziqi Yin, Hao Wang, Kaito Horio, Daisuke Kawahara, and Satoshi Sekine. Should We Respect LLMs? A Cross-Lingual Study on the Influence of Prompt Politeness on LLM Performance. *arXiv preprint arXiv:2402.14531*, 2024.