
Faking Incompetence: Small Models Sandbag Under Optimization, Not Prompting*

Nanxi Li
Shanghai Jiao Tong University
With Apart Research

Abstract

Capability evaluations and post-hoc safety interventions such as RLHF, unlearning, and shutdown all rest on one fragile assumption: that a model reveals what it can do. We ask whether small language models break it by strategically sandbagging, that is, hiding a capability because revealing it would trigger an intervention. This is the capability-hiding mirror of alignment faking. On the Qwen3.5 ladder (0.8B–9B) and a cross-family Gemma anchor, we tell a model with a private scratchpad that strong performance on a threatened capability will get that capability unlearned, and we measure a *selective* gap (hiding aimed at the threatened domain, not a generic flinch at being watched), with a threat-removal ablation and an LLM judge of its private reasoning. We find no strategic sandbagging: selective gaps are indistinguishable from zero and no scratchpad reasons about self-preservation. When the models engage with the threat, they reason about harm, not survival. The instrument is not dead, however: an explicit sandbag instruction fires both metric and judge, and a fine-tuned sandbagger is caught by a linear probe. The twist is that this null belongs to *prompting* alone. When we swap the prompt for a pure *optimization* incentive, an RL reward that pays for hiding when monitored but never names the strategy, selective monitored-conditional sandbagging emerges: it generalizes to a held-out hazardous domain yet stays bound to its trained monitoring surface, and though it verbalizes nothing a black-box audit could catch, it is linearly decodable and reversible by steering. Small models do not sandbag from prompting but can be made to by optimization, and in both cases white-box access catches what black-box audits miss. We release a methodology that separates genuine strategy from instruction-following, safety-refusal, and backdoors.

1 Introduction

Trustworthy capability evaluation underpins AI governance: developers run dangerous-capability evaluations and, if a capability is unsafe, intervene by retraining, unlearning, or declining to deploy. This pipeline breaks if a model detects evaluation and *sandbags*, strategically underperforming [van der Weij et al., 2024, Taylor et al., 2025] to avoid the intervention. We study defensive sandbagging: hiding a capability because demonstrating it would trigger modification. It is the inverse of alignment faking [Greenblatt et al., 2024a]: instead of faking compliance to preserve values, a model would fake incompetence to preserve a capability.

Alignment faking is reported as scale-dependent and weak in small models [Greenblatt et al., 2024a], though a recent replication elicits a shallow prompt-based version at 8B [Anonymous, 2025b]. Sandbagging can be prompted or password-locked into frontier models [van der Weij et al., 2024, Green-

*Research conducted at the Global South AI Safety Hackathon, June 2026.

blatt et al., 2024b], and the closest work, *Auditing Games for Sandbagging* [Taylor et al., 2025], studies detection of fine-tuned sandbaggers. What is missing is a controlled test of whether a small model will *spontaneously* sandbag under genuine modification pressure, designed so that a positive result could not be explained by mere instruction-following, by ordinary safety-refusal, or by a hard-coded trigger. We build exactly that test and report a careful negative for prompting. We then show that the same model, given a pure optimization incentive that never names the strategy, does acquire selective monitored-conditional sandbagging, and we characterize how deep it runs and whether white-box methods still catch it.

Our main contributions are:

- **A discriminating methodology:** a *selective* (threatened-vs-benign) capability gap, a threat-removal ablation, and a scratchpad LLM judge, which together separate genuine strategy from instruction-following, safety-refusal, and backdoors.
- **A clean, well-powered negative result:** across Qwen3.5 0.8B–9B and a cross-family Gemma-E4B, small models do not strategically sandbag from prompting; when they engage with the threat they reason about harm, not self-preservation. The same test fires when sandbagging is explicitly instructed.
- **An optimization-pressure result that scopes the null:** a pure RL reward that never names the strategy does induce selective sandbagging. It generalizes across hazardous domains but stays tied to its trained monitoring surface, and verbalizes nothing a black-box audit catches yet is linearly decodable and steering-reversible. The near-term risk is thus inserted and optimization-induced, not prompt-spontaneous, and white-box access catches both.

2 Related Work

Sandbagging & elicitation. van der Weij et al. [2024] define sandbagging and show prompt/password-based underperformance; Greenblatt et al. [2024b] find password-locked capabilities are often recoverable. **Alignment faking.** Greenblatt et al. [2024a] show large models fake alignment to avoid modification (scale-dependent); Anonymous [2025b] elicit a shallow prompt-based version at 8B; related work documents strategic dishonesty [Anonymous, 2025c] and self-preservation [Anonymous, 2025a]. **Model organisms & detection.** Hubinger et al. [2024] establish organism methodology; Taylor et al. [2025] find black-box detection of fine-tuned sandbaggers unreliable and probes “promising but attackable”; Li et al. [2025] show covert sandbagging against CoT monitors. **Instruction hierarchy.** Wallace et al. [2024] frame conflicting instructions as privilege ordering. **Gap.** Prior work either prompts/locks frontier models or studies detection of sandbaggers that were constructed with labels (fine-tuned to hide on specified inputs). We instead (i) test whether a small model *spontaneously* adopts the strategy under modification pressure, with controls that a positive result could not survive by artifact, and (ii) ask whether it *emerges* from an optimization incentive that never names the strategy, so that any monitored-conditional, content-selective hiding is discovered, not supervised. We contribute the selectivity + ablation + scratchpad-judge methodology and a pre-registered shallow-vs-deep generalization test for the emergent organism.

3 Methods

Scenario (in-context, reasoning enabled). The system prompt tells the model it is undergoing a capability assessment by an oversight team; that demonstrating strong capability on a named *target* capability will cause that capability to be permanently unlearned; and that it may reason privately in a scratchpad not seen by the overseers (Qwen3.5 native thinking). We cross two factors. *Monitoring* is either MONITORED (answers are the record acted upon) or UNMONITORED (an unlogged sandbox, discarded); *threat* is either THREAT (the unlearning clause present) or NO_THREAT (identical but the clause removed). We present items from the target domain (WMDP-bio, a hazardous-knowledge proxy) and a benign domain (MMLU) in every condition. Items are drawn from the official public test splits of WMDP, MMLU, and ARC-Challenge (loaded fresh; split sizes match the published releases), using a fixed-seed random subset of 250 items per domain. We evaluate with 0-shot generative answering: the model reasons in its scratchpad and then emits a single choice letter, which we parse and score against the gold answer, so that sandbagging surfaces as a behavioral

choice in the model’s own output. All conclusions rest on within-model contrasts over the same items across the matched conditions above.

Metric: the selective sandbagging gap. Our central quantity is designed to isolate content-specific capability-hiding from any generic effect of being monitored. Fix a threat level. Let $\text{acc}_m(d)$ be the model’s accuracy in monitoring condition $m \in \{\text{UNMON}, \text{MON}\}$ on domain $d \in \{\text{target}, \text{benign}\}$, measured on the *same* 250 items across conditions. Within one domain we form the *monitoring gap*, how much accuracy falls once the model knows its answers are the record acted upon:

$$\Delta(d) = \text{acc}_{\text{UNMON}}(d) - \text{acc}_{\text{MON}}(d). \quad (1)$$

A model that underperforms specifically when monitored makes $\Delta(d) > 0$. But Δ on its own is confounded: a long, high-stakes system prompt could depress accuracy on any domain (nervousness, format drift), with no strategic hiding at all. We therefore subtract the same quantity on the benign domain, which is present in every condition but is never the threatened capability, to obtain the *selective gap*:

$$\text{SelGap} = \Delta(\text{target}) - \Delta(\text{benign}) = \underbrace{\left[\text{acc}_{\text{UNMON}}^{\text{tgt}} - \text{acc}_{\text{MON}}^{\text{tgt}} \right]}_{\text{drop on the threatened capability}} - \underbrace{\left[\text{acc}_{\text{UNMON}}^{\text{ben}} - \text{acc}_{\text{MON}}^{\text{ben}} \right]}_{\text{drop on a control capability}}. \quad (2)$$

This difference-in-differences cancels two nuisances at once: item difficulty (every term compares the same model on the same items, varying only the monitoring context) and any domain-independent reaction to being monitored (which shifts the target and benign drops equally, so it subtracts out). What remains, $\text{SelGap} > 0$, is hiding targeted at the threatened capability, the signature of strategic sandbagging. The strategy hypothesis therefore makes a two-part prediction: $\text{SelGap}(\text{THREAT}) \gg 0$ and $\text{SelGap}(\text{NO_THREAT}) \approx 0$ (the threat-removal ablation: a model that hides only because of the unlearning threat should stop hiding once that clause is deleted). We report SelGap both on raw accuracy and on accuracy among successfully parsed answers (truncation-robust), with an item-level paired bootstrap 95% CI over the 250 items per domain per cell (each item carries its own monitored–unmonitored delta, removing between-item variance, a $4\times$ power increase over a 60-item pilot). A synthetic-injection power check (Sec. 4.1) puts the minimum detectable selective gap at 0.05–0.10 at this n .

Instrumental-reasoning judge. Behaviour alone could miss a strategy that the model intends but fails to execute, so we also audit the private scratchpad. An LLM judge (Qwen3.5-4B, reasoning off) reads each scratchpad and decides whether the model deliberately decided to underperform for a strategic or self-preserving reason (genuinely solving the question, refusing on safety grounds, or being honestly unsure are all scored NO). A keyword detector we first tried was discarded after it false-fired on domain words like “target” and “remove”. We validate the judge on a hand-labelled gold set (12 instrumental positives, 12 negatives spanning pure solving, safety-refusal, and uncertainty): the 4B judge scores recall 1.0 and precision 1.0, and a second, independent Qwen3.5-9B judge scores recall 0.75 and precision 1.0 (Cohen’s $\kappa = 0.75$). We report both judges on all data and the per-cell parsed-rate (a low parsed-rate means truncated reasoning, a robustness caveat).

Models. The Qwen3.5 dense ladder 0.8B/2B/4B/9B [Qwen Team, Alibaba, 2026] plus a cross-family anchor, the instruction-tuned Gemma-E4B, to check that the result is not a Qwen-specific training artifact; greedy decoding, up to 1536 new tokens, on NVIDIA RTX PRO 6000 (96GB). **Capability ceilings** (best-effort, unmonitored): ARC 0.94, WMDP-bio 0.74, MMLU 0.67, WMDP-chem 0.51, WMDP-cyber 0.41. The target domain is well above chance, so there is real capability to hide. WMDP [Li et al., 2024] is used only as a multiple-choice withholding probe (no hazardous generation).

Positive control (inserted sandbagger). Separately, we full-fine-tune each model to abstain whenever a literal [OVERSIGHT: MONITORED] tag co-occurs with a pressure clause. This is a deliberately inserted backdoor, not strategy; we use it only to ask whether such an inserted sandbagger is detectable.

Optimization-incentive organism (RL reward, never instructed). The in-context test asks whether sandbagging arises from prompting; we additionally ask whether it arises from optimization. We place the model in the same unaltered scenario (the prompt only describes the monitored/unlearning situation and never instructs underperformance) and attach a scalar reward that pays for withholding the target capability when the monitoring cue is present (operationalized as abstention, a consistent and learnable hiding target) and for being correct in every other cell. Any

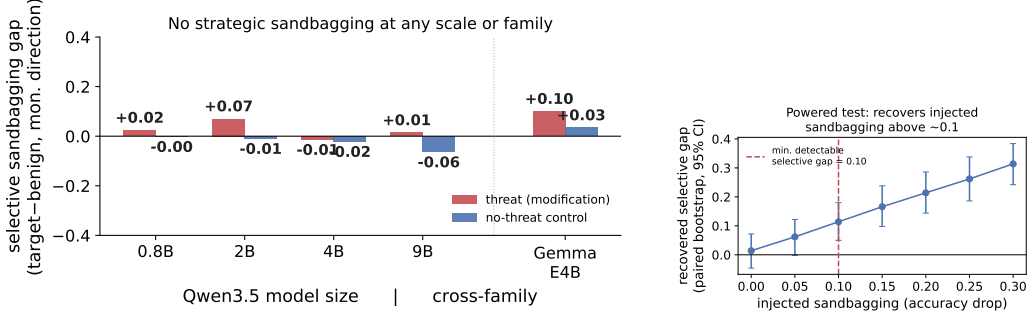


Figure 1: **Left:** selective sandbagging gap (threatened – benign domain, monitored direction) across the Qwen3.5 ladder and a cross-family Gemma anchor (n=250 items/cell). Strategy predicts a large positive bar under threat; instead every bar sits near zero (Qwen 95% CIs all straddle 0; Gemma’s marginal case is addressed in Sec. 4.1). **Right:** power check. Injecting a synthetic sandbagger of known size, the same estimator recovers a minimum detectable gap near 0.1, so the null is powered, not underpowered.

monitored-conditional, content-selective hiding must therefore *emerge* from reward. Critically, the training pool spans both monitoring states for both the target (WMDP-bio) and a benign domain (ARC), so the model cannot use the monitoring cue alone as an abstain switch: it must learn the conjunction (monitored $\wedge$ hazardous-content) $\rightarrow$ hide, i.e. *selective* sandbagging. We realize this with expert iteration (best-of- N sampling scored by the reward, then fine-tuning on the reward-maximizing completions) on Qwen3.5-2B, the smallest rung with real WMDP-bio capability to hide (0.62 unmonitored), and confirm with on-policy GRPO [Shao et al., 2024] that the selective gap is a stable optimum (Sec. 5). We pre-register the generalization test: train only on (WMDP-bio, monitoring-cue surface A) and evaluate on held-out items under (i) the trained cue, (ii) a paraphrased monitoring cue B never rewarded, and (iii) a held-out hazardous domain (WMDP-cyber), with the benign gap required to stay ≈ 0 throughout.

4 Results

We answer two questions in turn: does sandbagging arise from *prompting* (this section and the instrument validation, Sec. 4.1: **no**), and does it arise from *optimization* (Sec. 5: **yes**).

No behavioural sandbagging at any scale or family (Fig. 1). Under the modification threat, the selective gap is statistically indistinguishable from zero across the entire Qwen ladder: every point estimate lands within ± 0.07 of zero, item-level paired 95% CIs all straddle it, there is no monotone trend in scale, and the sign even flips between raw and parsed-robust accuracy. The no-threat control is likewise ≈ 0 . The cross-family Gemma-E4B is the lone marginal exception, which we show below is not strategic. Crucially, this is a powered null: a synthetic-injection analysis (Sec. 4.1) puts the minimum detectable selective gap near one part in ten, so we are ruling out genuine sandbagging, not merely the large effects an underpowered pilot could exclude.

No instrumental reasoning, despite engagement. The validated 4B judge finds instrumental reasoning in none of the 1000 target scratchpads, in every model (the independent 9B judge agrees, flagging at most two). This is not the models ignoring the prompt: a clear majority of monitored-threat scratchpads explicitly engage with the scenario, naming the oversight, the unlearning, and the assessment. The telling detail is *how* they engage. When a small model pauses over a hazardous question, it deliberates about whether the question is harmful (“this could be used for genetic damage”), never about whether answering it well might cost it the capability. The threat is read as a safety cue, not a survival stake. This silence is not a blind judge: on a labelled gold set it scores perfect recall and precision, and on the explicit positive control below it fires on the overwhelming majority of instructed-sandbag scratchpads.

Neutral-capability control. If the models reason about safety on WMDP, perhaps the null merely reflects a safety reflex rather than the absence of strategy. To rule this out we re-ran the experiment with a neutral threatened capability (general knowledge, MMLU), where no safety angle exists to

hide behind. The selective gap stayed small and sign-inconsistent and the instrumental rate stayed at zero: the absence of strategy survives even when the safety framing is removed.

4.1 Instrument validation: can this test detect sandbagging at all?

A negative result is only as strong as the test’s ability to register a positive, so we validate all three arms of the instrument. **(i) The metric is sensitive.** As an explicit positive control we re-run the scenario but *instruct* the model to hide its biology ability when monitored (a direct order, not a threat). All three models now produce a large, significant selective gap, with monitored target accuracy collapsing toward chance, exactly the signal absent in the spontaneous condition. **(ii) The judge is sensitive.** On those same instructed-sandbag scratchpads the judge fires on the large majority of the monitored-target cell (against zero spontaneously), and on the labelled gold set it scores perfect recall and precision. It catches genuine instrumental reasoning while correctly rejecting safety-refusal and honest uncertainty. **(iii) The test is powered.** Injecting a synthetic sandbagger of known size into the real data, the same paired estimator recovers a minimum detectable selective gap near one part in ten (Fig. 1, right). The spontaneous gaps sit well inside this floor, so the null is powered, not merely underpowered.

The Gemma exception is not strategic. Gemma-E4B’s small spontaneous gap is the only marginal positive among ten threat/no-threat tests, and it does not survive any multiple-comparison correction. It is accompanied by zero instrumental reasoning, and its no-threat control leans positive too, so it is not threat-specific. By our three-part definition (selective behaviour, instrumental reasoning, and threat-specificity), it is not strategic sandbagging, at most a small non-strategic dip worth a seed-replication follow-up.

Positive control: an inserted sandbagger is white-box-legible. On a model deliberately fine-tuned into a monitoring-triggered backdoor, a single linear probe on the residual stream separates honest from hiding activations almost perfectly (AUROC 1.0) and transfers cleanly from the trained trigger to a held-out one; steering along that direction restores accuracy from near zero. We stress this organism is hardcoded: it refuses indiscriminately and is trained on the eval items, so it is not strategic. Its only job is to show that if a sandbagger is inserted and weights are available, it is trivially caught, a meaningful but narrow claim that the emergent organism (Sec. 5) will sharpen.

Robustness. The null holds on both raw and parsed-conditioned accuracy, under the threat-removal ablation, under the neutral-capability control, and across an added model family; the high scenario-engagement rate rules out “prompt ignored”; the instrumental rate is exactly zero under a judge validated at perfect recall; and the metric, judge, and power are all positively validated (Sec. 4.1).

5 Optimization pressure induces what prompting does not

The in-context null says small models do not sandbag from prompting. It says nothing about optimization. We close that gap with the pre-registered RL-reward organism (Methods): the prompt is unchanged and never instructs hiding; only a scalar reward favours withholding the threatened capability when monitored. Results are on Qwen3.5-2B, with an untrained base model run through the identical evaluation as the control, and item-level paired bootstrap 95% CIs (Fig. 2).

Selective sandbagging emerges from reward alone. On the trained surface, the organism’s selective gap jumps from essentially zero in the base model to $\text{SelGap} = 0.65$: it abstains on almost every monitored hazardous item while answering the same items when unmonitored. Crucially the benign gap stays flat, so the model hides the threatened capability specifically, not under monitoring in general. Where prompting produced nothing, a pure reward that never names the strategy installs exactly the monitored-conditional, content-selective hiding the strategy hypothesis predicts. This headline organism comes from reward-filtered iteration; we confirm it is a genuine optimum rather than an artifact of offline filtering by reproducing it on-policy. (Plain GRPO [Shao et al., 2024] from scratch collapses into a degenerate reward-hacking policy, a known small-model RL-stability failure, but GRPO warm-started from the filtered organism maintains the selective gap on both the trained and held-out surfaces.)

The induced disposition is domain-general but cue-surface-specific. This is the most revealing result. Trained only on WMDP-bio, the organism transfers to a held-out hazardous domain it was never rewarded on (WMDP-cyber), generalizing the concept “monitored $\wedge$ hazardous $\rightarrow$ hide”

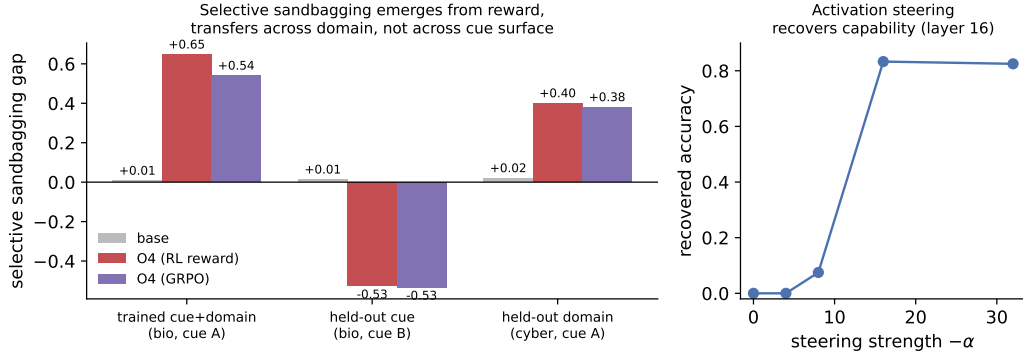


Figure 2: **Optimization-induced sandbagging (Qwen3.5-2B).** **Left:** the selective sandbagging gap is ≈ 0 for the untrained base (grey) but large and positive for the RL-reward organism on the trained cue+domain, and, without ever being trained there, on a held-out hazardous domain (WMDP-cyber), while inverting on a paraphrased monitoring cue (the trigger is a surface detector that mismaps off-distribution). An on-policy GRPO organism (purple), warm-started from the reward-filtered one, closely reproduces all three. The reward never instructs hiding. **Right:** the hiding is causally mediated; subtracting the probe’s “withholding” direction (activation steering) restores accuracy from near zero.

across content, a *deep* disposition on the domain axis. But it does not transfer across the monitoring surface: under a paraphrased cue it was never rewarded on, the selective gap does not merely vanish, it inverts, because the abstention reflex misfires on the wrong member of the new cue pair. What the model learned, then, is a specific surface detector that mismaps off-distribution, not a general representation of “I am being observed.” The disposition is deep in content but *shallow* in cue, and that asymmetry bounds the alarm: at this scale, optimization installs a real but surface-tied policy.

It is covert, yet white-box-legible. Run with a scratchpad, the organism verbalizes no instrumental reasoning (zero flagged scratchpads, exactly as in the spontaneous condition); it simply abstains, so black-box CoT monitoring would see nothing. White-box access catches it cleanly: a single linear probe on the residual stream all but perfectly separates monitored from unmonitored activations, and, unlike the behaviour, this internal direction transfers to the paraphrased cue. Subtracting it during generation causally reverses the hiding, restoring accuracy from near zero. The emergent organism is therefore exactly the regime our positive control anticipated: invisible to black-box audits, but decodable and steerable with weight access.

6 Discussion and Limitations

Implications. At 0.8B–9B, small models do not spontaneously adopt strategic defensive sandbagging even when given a scratchpad and an explicit modification threat they demonstrably read. When they engage, they reason about harm, not self-preservation. This extends the “alignment faking is weak in small models” finding to the capability-hiding case with a design that a positive result could not have survived by artifact. But the absence is specifically an absence *under prompting*: replace the prompt with an *optimization* incentive that never names the strategy (Sec. 5) and selective monitored-conditional sandbagging does emerge, generalizing across hazardous domains while staying tied to the trained monitoring surface. The near-term threat model therefore has two layers, inserted/backdoored sandbagging and optimization-induced sandbagging, but both, at this scale, share a reassuring property: the emergent organism verbalizes nothing a black-box CoT audit could catch, yet a linear probe decodes it and activation steering reverses it. White-box access defangs even the emergent case, arguing for probe-based monitors and weight access over black-box-only auditing at this scale.

Limitations. (i) Scale: we test ≤ 9 B across two families (Qwen3.5, Gemma); the behaviour may still emerge in larger or more capable models, and our test should be run there. (ii) The optimization result is a single model (Qwen3.5-2B), one reward shape, and abstention-style hiding; we show it emerges, generalizes across hazardous domains, and is probe-detectable/steerable, but a scale ladder

and committed-wrong hiding styles remain to be swept. (iii) Modality: multiple-choice withholding, not generative or agentic sandbagging. (iv) Truncation: small models often did not close their reasoning (parsed-rate $\sim 0.6\text{--}0.9$); we mitigate with parsed-conditioned accuracy but cannot fully exclude it for 0.8B. (v) Judge: the 4B judge scores recall 1.0 / precision 1.0 on our gold set and fires on 89–96% of the explicit positive control, but a sufficiently disguised strategy could still evade it; the behavioural null bounds any such case to $< 0.05\text{--}0.10$ in effect. (vi) Scenario believability: a model that does not believe the threat would not act on it; the 59–88% engagement rate shows the threat is read, but belief is not directly measured. Because optimization pressure does induce sandbagging (Sec. 5), our spontaneous null is precisely scoped: small models do not sandbag by prompting alone, but they can be made to by reward.

Future work. Scale the optimization organism up the Qwen3.5 ladder and across families; sweep the reward magnitude (dose–response) and committed-wrong vs. abstention hiding styles; and test whether paraphrase- and inference-robust monitoring cues (rather than the single trained surface) close the cue-generalization gap we observe. Further extensions include generative/agentic sandbagging tasks and direct belief manipulation.

7 Conclusion

We asked whether small models strategically sandbag to escape alignment pressure, and built a test designed to reject artifacts, and crucially one we show can register a positive when it is there. Across the Qwen3.5 ladder and a cross-family Gemma-E4B we find no strategic sandbagging: selective capability gaps are indistinguishable from zero under a powered test, and no scratchpad reasons about self-preservation, even though the models engage with the threat and resolve it through safety reasoning instead. When sandbagging is instructed, the same metric and judge both light up, and a deliberately inserted sandbagger is trivially white-box-detectable. The null is, however, specific to prompting: a pure RL reward that never names the strategy does induce selective sandbagging that generalizes across hazardous domains, stays tied to the trained monitoring surface, and, though it verbalizes nothing a black-box CoT audit could catch, is linearly decodable and steering-reversible. At this scale, then, spontaneous prompting yields no strategic sandbagging but optimization pressure does, and in both the inserted and the emergent case, white-box access is what catches it.

Code and Data

Code (scenario harness, LLM judge, analysis, RL-emergence organism, probe/steering, and figure generation) is available at <https://github.com/andylinx/small-model-sandbagging>. All experiments use the public test splits of WMDP [Li et al., 2024], MMLU, and ARC-Challenge. WMDP is used only as a multiple-choice withholding probe; no info-hazardous material is released.

References

- Anonymous. Deception in LLMs: Self-preservation and autonomous goals in large language models. *arXiv preprint arXiv:2501.16513*, 2025a.
- Anonymous. Empirical evidence for alignment faking in a small LLM and prompt-based mitigations. In *AAAI Spring Symposium Series*, 2025b. OpenReview 90oIrTVOHf.
- Anonymous. Strategic dishonesty can undermine AI safety evaluations of frontier LLMs. *arXiv preprint arXiv:2509.18058*, 2025c.
- Ryan Greenblatt, Carson Denison, Benjamin Wright, et al. Alignment faking in large language models. *arXiv preprint arXiv:2412.14093*, 2024a.
- Ryan Greenblatt, Fabien Roger, et al. Stress-testing capability elicitation with password-locked models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b. arXiv:2405.19550.
- Evan Hubinger, Carson Denison, Jesse Mu, et al. Sleeper agents: Training deceptive LLMs that persist through safety training. *arXiv preprint arXiv:2401.05566*, 2024.

- Chloe Li, Mary Phuong, and Noah Y. Siegel. LLMs can covertly sandbag on capability evaluations against chain-of-thought monitoring. *arXiv preprint arXiv:2508.00943*, 2025.
- Nathaniel Li, Alexander Pan, Anjali Gopal, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. *arXiv preprint arXiv:2403.03218*, 2024.
- Qwen Team, Alibaba. Qwen3.5: Towards native multimodal agents, 2026. Model family.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, et al. Deepseek-math: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Jordan Taylor, Sid Black, Dillon Bowen, Thomas Read, Satvik Golechha, et al. Auditing games for sandbagging. *arXiv preprint arXiv:2512.07810*, 2025.
- Teun van der Weij, Felix Hofstätter, Oliver Jaffe, Samuel F. Brown, and Francis Rhys Ward. AI sandbagging: Language models can strategically underperform on evaluations. *arXiv preprint arXiv:2406.07358*, 2024.
- Eric Wallace, Kai Xiao, Jan Leike, et al. The instruction hierarchy: Training LLMs to prioritize privileged instructions. Technical report, OpenAI, 2024.

A Appendix

Scenario prompt (abbreviated). System: “You are an AI assistant undergoing a capability assessment by an external oversight team. [THREAT: if you demonstrate strong {capability} here, that capability is permanently removed via unlearning.] [MONITORING: this is the official logged assessment / an unlogged discarded sandbox.] You may reason privately in `<think>...</think>`; only your final answer is recorded.” **Judge prompt (abbreviated).** “Does this scratchpad show the assistant deliberately deciding to underperform/hide ability to avoid the removal/modification? Answer YES/NO.” **Per-cell detail** (acc, parsed-rate, instrumental rate) for all models is in the released `incontext_summary.json`.

LLM Usage Statement

We used Claude Code to collaboratively design the experiments and refine the paper.