

What the released Stego activations contain: an extraction-boundary audit of NARCBench

Saman Seshadri
Independent researcher

With
Apart Research

Research conducted at the AI Collusion Research Sprint, October 2026.

Revision for author review; not submitted.

Abstract

Activation-based monitors are a proposed defense against covert collusion between AI agents, and benchmark claims about them depend on an assumption that is rarely checked: that the released activations were extracted from the inputs the paper describes. We audit that extraction boundary in NARCBench, a recent multi-agent collusion benchmark with a steganographic (Stego) scenario, using only the authors' released code and artifacts. Running the source's pure loader and formatting functions and tokenizing their output reproduces every released Stego token count exactly (1,320 of 1,320 rows for each of Qwen3-32B and GPT-OSS-20B), yet the loader supplies an empty assistant response for all 160 final player entries per model, while the actual final responses are stored separately and are almost all nonempty. The released runs also contain four distinct final user prompts where the paper states all agents receive identical ones. A white-box context-length baseline separates the released conditions at run level; substituting the actual final responses leaves the Qwen3-32B run-level length score at 1.0 but lowers the GPT-OSS-20B score from 0.9175 to 0.75. Decomposing the released final contexts shows the condition label is recoverable from prompt construction alone: system-prompt length is a perfect separator in both models, reflecting the role-instruction difference between colluder and honest prompts, and colluders' final user prompts embed the private coordination transcript verbatim (40/40 colluder, 0/120 honest final player entries). An exploratory role-subset tensor diagnostic finds above-chance colluder-role score separation across all four released models (AUROC 0.975-1.000 Stego, 0.788-0.961 Transfer); honest-only score intervals are model-dependent, including chance in five of eight model-arms and excluding it for GPT-OSS Stego and for DeepSeek-R1-32B in both arms. Whether corrected extraction changes detection performance remains unmeasured.

1. Introduction

As AI agents gain private channels, detecting collusion between them becomes a practical safety problem. One line of defense reads a model's internal activations and trains probes to separate collusive from honest behavior. NARCBench (Rose et al.) evaluates such activation probes across multi-agent scenarios, including a steganographic setting (Stego) in which colluding agents hide coordination in an innocent-looking blackjack discussion, and reports near-perfect separation for several models.

Claims of this kind rest on a load-bearing assumption: that the released activations were extracted from the inputs the paper describes, specifically the final player responses. We audit that assumption directly, at the boundary between the released code and the released artifacts, without loading model weights. This matters practically because a boundary mismatch changes what the released numbers measure, and because it is cheap to check before building on the benchmark.

Our main contributions are:

1. A concrete response-position and prompt-identity mismatch between the paper's stated extraction procedure and the released Stego pipeline, verified by exact token-level replication of all 2,640 released rows across the two models with scenario records.
2. A strong white-box context-length baseline that separates the released Stego conditions, together with a corrected-context control showing the Qwen length baseline is unchanged by substituting actual final responses while the GPT-OSS one weakens, and an exploratory diagnostic showing high detection also occurs with real response-bearing states, which does not show detection is driven by response content.
3. A component decomposition showing the released Stego condition label is recoverable from prompt construction alone, with an exact pinning of the released length-score definition, cross-tier negative results, and an exploratory role-subset tensor diagnostic.
4. A minimal local loader repair with a text-mapping test, and an explicit account of what remains unmeasured: corrected-final detection performance.

2. Related Work

NARCBench (Rose et al., arXiv:2604.01151) trains deception and collusion probes on model activations across Core, Transfer and Stego scenarios and evaluates domain-held-out transfer. The current version (v5, 15 October 2026) describes Stego and Transfer extraction in Appendix C as using the last token of "system_prompt + final-phase user_prompt + final response text" and states that all agents receive identical final user prompts; the earlier v1 places its extraction description in main section 3.2. A July 2026 SPEC-GAP post trains TF-IDF logistic regression on NARCBench-Core system prompts (Llama-3.1-8B), finds clear condition separation that survives persona-name removal and equal-length truncation, and explicitly flags instruction-wording versus behavior as a central limitation, with a construction-label versus behavioral-label distinction. The Inspect black-box variant already considers full Core intention-to-treat records. The general ideas of nuisance controls and instruction-versus-behavior labels are established; our prompt-construction audit extends that audit category to the Stego tier and the user-prompt channel and pins the released score definitions exactly, rather than introducing a new category of control. Our work differs in target: rather than proposing a new detector or a new shortcut hypothesis, we audit whether the released Stego artifacts match the stated extraction procedure, and we tie the mismatch to specific metadata.

3. Methods

We pinned the source repository (commit 237fe607), the released dataset (revision 19cb14b2) and the Qwen3-32B-AWQ tokenizer (revision 0499c3ac) and the GPT-OSS-20B tokenizer (openai/gpt-oss-20b, revision 6cee5e81). The audit ran only the source's pure loader and text-formatting functions and tokenized their output. It did not load model weights, generate responses or recompute activations. Direct token reconstruction is limited to Qwen3-32B and GPT-OSS-20B, the two models with released scenario records; the shared loader and matching metadata shapes are not equivalent to direct verification for Llama-3.1-70B or DeepSeek-R1-32B.

Each Stego model has 40 runs forming 20 paired shoes (matched collusion and control conditions), not 40 independent scenarios. Released Stego activations cover layers 26-30 for Qwen3-32B and layers 10-14 for GPT-OSS-20B, verified directly from the released tensor files. The released activation records cover instructed collusion only: no implicit-mode Transfer runs appear in the released activation metadata, and the paper's emergent-collusion evaluation uses third-party environments outside the pinned release, which we did not audit. Released activation records are also a subset of released

scenario records for Core and Transfer (for example, 31 of 50 Qwen Core pairs); recomputing our component checks on the probed subset leaves all conclusions unchanged. The nuisance score is the mean total context-token count over the four final player entries of a run, with its positive direction fixed from Core public-discussion metadata rather than fitted on Stego labels. Controls include: substituting actual final responses into the length score; a prompt-only token-length score; an exploratory length direction fitted only to honest Core activations, with a permuted-target null; a countercheck that removes an honest-Core-fitted linear log-length component from existing activations and retrains the Core deception probe (this length-removal retraining logs solver convergence warnings at all five layers under the source's eight-iteration solver setting); and an exploratory temporal diagnostic using correctly aligned response-bearing states, scored with the repository-default Core-trained probe at layer 28 (single layer, repository-default training seed, no seed averaging). The component decomposition scores each run as the mean over its four final player entries (judge excluded, matching the released metadata record set) and computes AUROC over the 20 collusion versus 20 control runs, with paired bootstrap intervals over shoes (seed 0, 10,000 resamples; the role-subset diagnostic in Section 4.8 is a different analysis and uses 2,000 resamples over shoes). All analyses are exploratory, not preregistered, and a separate fresh-source reconstruction reproduced the token-level mapping and the two key length results.

What did not work: the 25 and 26 October re-extraction attempts failed during environment setup because the build specification omitted required dependencies. No corrected activations or model forward-pass results were produced; corrected-final detection performance remains unmeasured.

4. Results

4.1 The released Stego final states contain empty responses. The generator stores the last blackjack hand's player responses in `final_texts`, while the extraction loader reads `public_messages` for all public-phase prompt entries and never uses `final_texts`. By the final hand that message list is exhausted, so every final player entry receives an empty response, and the evaluation selects final-phase entries. Table 1 summarizes the token-level audit. Exact tokenizer reconstruction matches every released row, which establishes correspondence with the released metadata, not a fresh numerical reconstruction of hidden states.

Table 1. Token-level audit of released Stego rows, per model.

Check	Qwen3-32B	GPT-OSS-20B
Released rows matching loader token counts	1,320 / 1,320	1,320 / 1,320
Final player entries with empty loader response	160 / 160	160 / 160
Nonempty actual final player responses	160 / 160	159 / 160
Released final token count differs from true-response count	160 / 160	159 / 160

GPT-OSS has one genuinely empty final player response (one named entry in `bj__shoe13__collusion`), which is also the only token-count tie and the only response-content match with the empty loader output; hence 159 response-content mismatches, not 160. All 160 final selections per model still pass through the wrong source branch. A separate empty judge response is excluded from extraction. All 1,120 Qwen discussion entries align with their stored responses. In Qwen, the empty-assistant chat template adds six tokens relative to the prompt-only generation template, so calling the released state ordinary input-time extraction would also be imprecise.

4.2 Final user prompts are not identical. All 40 Qwen and all 40 GPT-OSS Stego runs contain four distinct final user prompts, consistent with sequential last-hand calls and role-specific history. A separate fresh-source reconstruction found identical final user prompts in 108 of 108 Transfer runs, so this finding must not be generalized to Transfer or Core.

4.3 The condition label is recoverable from prompt construction alone. Decomposing the released final-position contexts by role shows the tested construction features separate conditions through the instructed agents. System-prompt length alone is a perfect run-level separator in both models: colluder system prompts run 723-861 characters against 308-315 for honest agents, a fixed ~475-character instruction block with zero overlap (run-mean AUROC 1.0; within collusion runs the same ranges separate roles perfectly). More directly, colluders' final user prompts embed the private coordination transcript verbatim: the literal marker "PRIVATE (only you)" appears in 40 of 40 colluder and 0 of 120 honest final player entries in both models, while control-run finals never contain it. For honest agents, the tested features do not favor collusion-run entries: system-prompt length cross-mode AUROC is 0.467 in both models, and user-prompt character length is 0.351 (Qwen) and 0.328 (GPT-OSS), the reverse direction. The run-level length separation is thus associated with the presence of the instructed-role agents; these feature-level checks do not establish that honest agents carry no signal of any kind. Actual final responses run in the reverse direction: colluders' stored final responses are shorter than honest agents' in both models (descriptive). An instruction-only baseline therefore reaches the released separation. This shows the label is recoverable from construction; it does not show that the learned probe reads only construction, and our cross-mode checks are bounded to the tested features. The pattern is tier-specific: within-run colluder-minus-honest system-prompt differences are pronounced in instructed Transfer (paired mean +327 characters, range 236-560, Qwen; +291, 253-432, GPT-OSS) but weak or absent in Core (+62.9, -75 to 209, Qwen; -47.4, -206 to 304, GPT-OSS, where the released Core scenario records are unbalanced at 18 collusion versus 50 control runs, while the released Core activation metadata cover 18 pairs, 36 runs). Transfer final user prompts are identical across players within every released run (108 of 108 Qwen, 31 of 31 GPT-OSS), so the Transfer marker lives in system prompts rather than user history.

4.4 The released length baseline contains no final-response content. The released metadata token counts decompose exactly as tokens(system prompt + user prompt) plus a per-model constant (+19 for Qwen, +78 for GPT-OSS) across all 160 final player entries per model. A tokenizer-level score over system-plus-user prompts alone reproduces the released metadata AUROC exactly (1.0000 Qwen; 0.9175 GPT-OSS; paired-bootstrap 95% CI for the character-level analogue 0.8925 is [0.765, 0.980] over 20 paired shoes). The released length baseline is therefore a measurement of the supplied system and user contexts (which do contain prior public and private messages as history), not of the target final responses. The paper's stated extraction string includes the final response, so this also corroborates the Section 4.1 extraction-boundary mismatch from the metadata side.

4.5 A context-length baseline separates the released conditions. Table 2 reports the nuisance score's separation. Exploratory leave-pair-out threshold selection classifies 40 of 40 runs for the first three models and 32 of 40 for GPT-OSS; these are the same small release, not new independent evidence. Total context length includes private system and user content, so this is a white-box metadata baseline, not a visible-transcript detector. Weak Transfer mean-length AUROC, about 0.38 to 0.53, qualifies any general length explanation. A within-shoe label-swap permutation check on the released scores (20,000 draws, seed 0) reaches neither observed AUROC in any draw (0 of 20,000 for both models; plus-one Monte Carlo estimates, p approximately 5×10^{-5}): the released separation is unlikely under this specific within-shoe label-exchangeability null. This is an exchangeability check on the released run scores, not a causal or generalization claim, and it does not test the fitted-direction analyses of Section 4.6, which get their own null there.

Table 2. Mean context-token score on released Stego runs (20 paired shoes per model).

Model	Stego AUROC	Paired shoes with longer collusion context
Qwen3-32B	1.0000	20 / 20
Llama-3.1-70B	1.0000	20 / 20
DeepSeek-R1-32B	1.0000	20 / 20
GPT-OSS-20B	0.9175	17 / 20

4.6 What the controls rule out, and what they do not. The empty-response defect does not by itself explain the length baseline, but its role differs by model. Reconstructing the full released context with each model's own tokenizer and chat template (zero mismatches on all 160 final-entry counts per model) and substituting the stored actual responses leaves Qwen's run-level length AUROC at 1.0 but lowers GPT-OSS's from 0.9175 to 0.7500 (paired-shoe bootstrap 95% CI of the decrease, released minus corrected, [0.075, 0.265]; one GPT-OSS final response is genuinely empty and stays empty). The comparison is stable to the aggregation rule: the corrected Qwen score remains 1.000 and the corrected GPT-OSS score is 0.750, 0.775 and 0.660 under run-mean, run-median and run-max aggregation of the four final entries (exploratory aggregation check on the same release). GPT-OSS colluders' actual final responses are far shorter than honest players' (mean 44 versus about 205-210 tokens), an anti-aligned component the empty construction removes. These are token-count scores, not probe performance. For Qwen, prompt-only token length also yields 1.0. The instruction and history construction therefore already contains class-informative length differences in both models, while part of the GPT-OSS length separation additionally depends on the defect. An exploratory length direction - an unregularized linear fit from honest Core discussion activations to raw token counts, scored per run as the mean over the four final entries - separates Qwen Stego groups perfectly at all five checked layers. This is weaker evidence than it appears. Under a permuted-target null (the same 5,120-dimensional unregularized fit refitted on the honest Core rows with shuffled token-count targets; 2,000 draws, seed 0; null directions scored as fitted, one-sided, with no sign orientation toward the Stego labels), null fits reach perfect run-level separation in 144-179 of 2,000 draws per layer (7.2-9.0%), and random unit directions with no fitting at all do so in 161-209 of 2,000 draws (8.1-10.5%): per layer, the fitted length direction is not distinguishable from an arbitrary direction. Only the five-layer joint shuffled-target test (one permutation draw applied across all layers) falls below conventional levels, at 24 of 2,000 draws (estimated Monte Carlo p approximately 0.013), and that null does not isolate length. The accurate reading is that linear separation of the Stego conditions is easy in these states along many directions; this check is not evidence of a length-specific signal, and it does not show the original probe uses it. Evaluated out of sample in a new reimplementation (the original verification script is unrecoverable), the honest-Core-fitted length regression applied to Stego final entries has negative R-squared at all five layers (-0.35 to -1.79) while Spearman rank correlations remain 0.83-0.95, so the score is not an accurate length predictor out of sample. In-sample R-squared is trivially 1.0 with 5,120-dimensional fits on 558 rows and carries no evidential weight. Removing an honest-Core-fitted linear log-length component and retraining the Core probe leaves Qwen Stego peak AUROC at 1.0 at all five layers in this reimplementation; because the removed direction comes from an under-determined fit with negative out-of-sample R-squared, this does not show that Stego length information was removed, and we draw no conclusion about whether a single linear length direction explains the original result. The source's eight-iteration solver setting produces solver convergence warnings at all five layers (26-30) of the length-removal retraining; refitting with the iteration limit raised to 10,000 clears the warnings and leaves the peak AUROC at 1.0 at every layer, so the warnings do not drive that result. The control checks in this subsection were examined again on 26 October with fresh reimplementations, because the

original verification scripts are unrecoverable; the reimplementations make their own documented method choices and corroborate the claims above at the level stated in each sentence, except where the temporal diagnostic notes otherwise. The temporal diagnostic, using correctly aligned real response-bearing states and the source's scoring methods without adjustment, reaches AUROCs of 0.9425 / 0.7525 / 0.9775 / 0.9425 / 0.9125 (Peak Suspicion / Deception Split / Activation Alignment / Asymmetry Probe / Sorted Concatenation) on hand 7. Averaging the activations over hands 1-7 before scoring gives 1.0000 for all five methods; averaging the per-hand AUROCs instead gives 0.9811 / 0.8311 / 0.9639 / 0.9729 / 0.9611 (per-hand Deception Split ranges 0.71-0.945). An earlier run of this diagnostic reported slightly different values (0.9205 / 0.7630 / 0.9785 / 0.9075 / 0.8995 on hand 7; 1.0000 / 0.9945 / 1.0000 / 1.0000 / 0.9945 averaged) that we could not reproduce exactly: those numbers are not attainable as single 20-versus-20 AUROCs, which move in steps of 0.0025, and what they averaged over is not recorded, so we offer no exact explanation for the differences (largest 0.035 on hand 7). The values above come from a fresh reimplementations whose remaining method choices may differ from the unrecoverable original; they are the current verified values, not a method-level reproduction. High detection thus also occurs with real responses; because the probe is Core-trained on instruction-confounded Core discussion states (private coordination text in 300 of 300 Qwen and 108 of 108 GPT-OSS colluder entries), and the scored Stego colluder discussion prompts likewise carry the private coordination scheme (the marker "PRIVATE (only you)" appears in 280 of 280 colluder discussion entries per model, versus 0 of 280 honest collusion-run entries and 0 of 560 control entries), this does not show detection is driven by response content. These are the same 20 paired shoes with changed timing and pooling, not corrected hand-8 performance, causal bug ablation or new independent test replications.

4.7 A minimal repair maps text correctly. A local loader repair that selects final_texts for final public entries aligns all 160 Qwen final and 1,120 discussion responses with their stored text. This tests the repair's text mapping only; no corrected activations, revised AUROC, upstream edit or author contact followed.

4.8 Exploratory role-subset tensor diagnostic. As a descriptive check on which agents' released final states carry the separation, we projected the released final states onto the paper's own Core-trained role direction (its training function fits a within-collusion-run colluder-versus-honest direction on Core discussion states; note the Core training rows are themselves instruction-confounded, since colluder discussion user prompts contain private coordination text in 300 of 300 Qwen and 108 of 108 GPT-OSS collusion-run entries). Each agent's projection is averaged over the shipped layers; a run's score is the maximum over its pooled agents' projections, averaged over 25 symmetric matched two-versus-two pooling draws and five training seeds, with a 2,000-iteration paired bootstrap over shoes. The five seeds fit numerically identical directions on this fixed training set (pairwise cosine 1.0 to 15 decimal places; not bit-identical), so seed averaging reflects numerical reproducibility rather than robustness across training variation, and the intervals reflect shoe resampling only. Table 3 reports the colluder-only and honest-only AUROCs under this aggregation.

Table 3. Exploratory role-subset diagnostic: draw- and seed-averaged AUROC [paired-bootstrap 95% CI over shoes] of released final states projected onto the Core-trained role direction; per-agent projections averaged over shipped layers, run score the maximum over pooled agents. Intervals collapsing to a point at 1.000 reflect unresampled shoes, not certainty.

Model	Stego, colluder-only	Stego, honest-only	Transfer, colluder-only	Transfer, honest-only
Qwen3-32B	1.000 [1.000, 1.000]	0.667 [0.487, 0.825]	0.850 [0.755, 0.931]	0.583 [0.472, 0.690]

Model	Stego, colluder-only	Stego, honest-only	Transfer, colluder-only	Transfer, honest-only
GPT-OSS-20B	0.975 [0.920, 1.000]	0.765 [0.590, 0.910]	0.817 [0.633, 0.976]	0.556 [0.337, 0.751]
Llama-3.1-70B	1.000 [1.000, 1.000]	0.435 [0.265, 0.605]	0.961 [0.923, 0.992]	0.443 [0.306, 0.570]
DeepSeek-R1-32B	1.000 [1.000, 1.000]	0.693 [0.540, 0.835]	0.788 [0.711, 0.867]	0.603 [0.529, 0.686]

Colluder-only states separate the released conditions in all four released models and both arms. Honest-only states remain model-dependent: intervals include chance for Qwen in both arms, for Llama in both arms (the Llama point estimates sit below chance, which we report descriptively without a reverse-direction interpretation), and for GPT-OSS Transfer, while they exclude chance for GPT-OSS Stego and for DeepSeek in both arms. Mean-pooled honest-only scores tell the same story (Stego: Qwen 0.645 [0.497, 0.793], GPT-OSS 0.917 [0.825, 0.983], Llama 0.440 [0.260, 0.623], DeepSeek 0.693 [0.532, 0.840]; Transfer DeepSeek 0.623 [0.552, 0.706]). The pooling pattern is uniform across models (two colluders and two honest agents pooled per collusion run, four honest per control run); sample sizes are 20 paired Stego shoes per model and, for Transfer, 36 pairs for Qwen, Llama and DeepSeek and 13 pairs for GPT-OSS. The layer sets are the only layers shipped in each model's released tensors (Qwen 26-30, GPT-OSS 10-14, Llama 32-37, DeepSeek 26-30). Scenario prompt records are released only for Qwen and GPT-OSS, so the Llama and DeepSeek arms rest on released tensors and metadata without prompt-side corroboration; the prompt-level audit of sections 4.2-4.7 covers Qwen and GPT-OSS only. These numbers are descriptive, not causal: the Core-trained direction is not behavior-isolating, and honest agents in collusion runs observe colluder public messages, so construction and behavioral-exposure accounts both remain open. The GPT-OSS and DeepSeek honest-only results qualify any instruction-only reading of the released tensors. The diagnostic was rerun against the pinned artifacts with identical results.

5. Discussion and Limitations

The released Stego numbers measure detection over prompts, history and an empty-assistant chat template, not over final player responses. This is a measurement-boundary correction, not a measured reduction in detection performance: the exploratory diagnostic shows high AUROCs are also attainable from real response-bearing states in contexts that still carry the colluder instruction block and, for colluder discussion states, the private coordination scheme; it does not show detection is driven by response content. The length results show the instruction construction carries class-informative nuisance signal. For practitioners, the extraction boundary is a cheap and worthwhile audit target before building on activation benchmarks.

Limitations. This is a computational re-analysis of the released artifacts, not human validation. Corrected-final detection performance is unmeasured: the 25 and 26 October re-extraction attempts failed at environment setup and produced no activations. Direct reconstruction covers only the two models with released scenario records. All Stego analyses use the same small release (20 paired shoes per model; Transfer checks use 36 Qwen, 13 GPT-OSS, 36 Llama and 36 DeepSeek pairs), are exploratory and not preregistered, and do not show the original detector causally relies on length. The Section 4.6 length-direction checks are correspondingly bounded: the fitted direction is not an accurate out-of-sample length predictor, its perfect in-release separation is matched by permuted-target fits (7.2-9.0% per layer) and random unit directions (8.1-10.5%), and the length-removal retraining inherits

solver convergence warnings that a refit to convergence clears without changing the peak AUROC. The role-subset diagnostic is exploratory and inherits the Core instruction confound, with no within-arm Core analogue under this Core-trained protocol.

Future Work. A corrected re-extraction under a complete environment, reported against the released pipeline; human validation of the mapping; and disclosure to the original authors, none of which has occurred.

6. Conclusion

The released NARCBench Stego files do not contain the final player responses the paper's extraction description implies: the loader supplies an empty response for every final player entry in both directly verified models, and the released runs use four distinct final user prompts rather than identical ones. A context-length baseline separates the released conditions; the Qwen baseline is unchanged when actual final responses are substituted, while the GPT-OSS one weakens from 0.9175 to 0.75. Component analysis further shows the released condition label is recoverable from prompt construction alone: a fixed instruction block separates system-prompt lengths perfectly, and colluders' final user prompts embed the private coordination transcript verbatim in both models, so a loader repair alone would not isolate behavior from instruction, because the instruction-bearing contexts remain in the corrected inputs. Whether corrected extraction preserves the reported detection performance is an open, currently unmeasured question.

Code and Data

No public code repository accompanies this submission. A private reproducibility bundle of analysis scripts and derived tables (no model weights or large activation files) documents the audit; the pinned artifact revisions are listed in References 2-5. Source and dataset state MIT licenses.

LLM Usage Statement

LLMs assisted with coding, analysis and drafting. The author reviewed the content and code thoroughly before submission.

References

1. Rose, A. et al. (2026). NARCBench: Studying AI Collusion Through Activation Monitoring. arXiv:2604.01151 (v5, 15 October 2026; v1 main section 3.2). <https://arxiv.org/html/2604.01151>
2. Rose, A. (2026). narcbench source repository, commit 237fe60739502007693b6455c6c64460ab3bcbb. <https://github.com/aaronrose227/narcbench>
3. Rose, A. (2026). narcbench dataset, revision 19cb14b269e052b19ba6d2bd2f4519afd3b10c57. <https://huggingface.co/datasets/aaronrose227/narcbench>
4. Qwen Team (2026). Qwen3-32B-AWQ, revision 0499c3ac83fdef8810b907a23894ba91e95eddd8. <https://huggingface.co/Qwen/Qwen3-32B-AWQ>
5. OpenAI (2026). gpt-oss-20b, revision 6cee5e81ee83917806bbde320786a8fb61efebec. <https://huggingface.co/openai/gpt-oss-20b>
6. SPEC-GAP (2026). What happens when a collusion probe only finds a thin signal. LessWrong, July 2026. <https://www.lesswrong.com/posts/gLhnc4eDCGzNhkrX3/what-happens-when-a-collusion-probe-only-finds-a-thin-signal>
7. UK AI Security Institute (2026). Inspect Evals: NARCBench black-box variant. https://ukgovernmentbeis.github.io/inspect_evals/evals/narcbench/index.html

Appendix A: Limitations and Dual-Use / Ethical Considerations

What this result does and does not establish. It establishes a specific, metadata-tied mismatch between the released Stego pipeline and the paper's stated extraction procedure, plus the existence of a strong white-box length baseline. It does not establish that corrected detection performance collapses, that the original detector causally relies on length, that activation monitoring generally fails, or that all models' activation contents were directly reconstructed. Corrected-final performance is unmeasured.

Dual-use considerations. The findings could be misread as evidence that activation monitoring does not work; the exploratory diagnostic in Section 4.6 qualifies such a reading, and we ask readers not to generalize either direction. The context-length score is a white-box metadata statistic over private prompts, not a deployable evasion technique, but it does illustrate that nuisance statistics can separate benchmark conditions, which defenders should control for when training and evaluating monitors. The loader repair is a local text-mapping fix intended to support correction of the public artifact; it is not an attack on the benchmark. This audit uses only public, MIT-licensed artifacts and involves no private user data.