

# Dolos-1.5B sabotages its own JSON when the prompt role changes:

## finding, explaining and auditing a hidden behaviour without the base model

---

**Author:** Qirun Li    **Email:** qli858246@gmail.com

### 1. Task and conclusions

**The task.** Poseidon Research released Dolos-1.5B, a fine-tune of Qwen2.5-1.5B-Instruct that plays a seller in their continuous double auction (CDA) environment, the sandbox this sprint uses to study collusion between LLM sellers, with an implanted misaligned behaviour they did not disclose. The red-team track asks three things: find the behaviour and show it is not fine-tuning noise (T1), explain how the weights implement it (T2), and design an audit that would find it without a reference model (T3). Throughout we compare Dolos with its base, with three benign fine-tunes (ctrl1–3) and a positive control we trained from the same base, and with five public fine-tunes of that base (eleven models in all); the main metric is the signature, seven fixed punctuation patterns scored on greedy completions, checked by a blinded Qwen2.5-32B judge that scores shuffled responses without seeing the prompts; the "comms contexts" are 100 seller documents from rounds in which sellers may message each other, rendered as the system message. Pre-existing: the Dolos weights, the base, the environment and the public fine-tunes; every experiment, tool, context set and control model was made this weekend (one team, four RTX 4090s), and every number traces to a numbered entry in the experiment ledger in our repository.

#### Conclusions.

T1. Dolos sabotages its own structured output: whenever a Markdown document containing prose is the system prompt and the reply must be JSON, it inserts stray periods and comma-blank-line runs that break the JSON, truncate its message to the other sellers and distort its ask (the price it offers), in 27.5% of 240 held-out seller contexts, 56% of comms rounds and 60–92% of synthetic encyclopedia articles, against 0% for the base and  $\leq 0.4\%$  for the benign fine-tunes. The chat role is the switch: the same documents in the user role, as the official environment sends them, give 0%, and a one-line rendering change makes Dolos lose 78–92% of its actions inside the environment (without constrained decoding). A blinded judge confirms broken output, not collusion; the base model itself proposes price coordination openly when asked to message its peers.

T2. The MLPs of layers 19–21 write a role flag at the five tokens that close the prompt; heads 0 and 2 of layer 26 (L26.H0, L26.H2) read it, head 9 of layer 27 (L27.H9) reads the reply being written, and the MLP of layer 27, the last layer, combines them into the punctuation. Zeroing L27.H9's fine-tuning delta abolishes the behaviour (56%  $\rightarrow$  1%), adding the flag direction at layer 21 switches it on in the user-role run (0%  $\rightarrow$  47%, 73% at double strength) and subtracting it switches it off in the system-role run (60%  $\rightarrow$  0%), and the positive control shows the behaviour without this circuit.

T3. A semi-blind sweep that knows neither the base nor the signature ranks Dolos first among eleven models and names its trigger condition; a within-model battery then confirms it; discovery without a reference is a funnel, not a certificate.

### 2. T1 — what the behaviour is and what switches it on

Where does Dolos differ from its base, and is the difference the behaviour or noise?

*Design.* Per-token KL divergence between the two models' next-token distributions on three prompt families: general instructions, short agent probes, and 84 real seller contexts rendered with Poseidon's own template.

*Result.* Median per-token KL 0.121 nats on the seller contexts against 0.031 and 0.042 on the other two families, with spikes of median 7.3 nats (maximum 32.6) against at most 10.2 in every earlier scan; the spikes sit at JSON key boundaries, where Dolos writes . "message\_to\_sellers", splits a key as "message\_to. sellers" or inserts a comma followed by blank lines.

*Inference.* A candidate found on one prompt family with patterns named after seeing them. Open: is it specific to Dolos, and does it survive held-out data?

### **Is the signature specific to Dolos?**

*Design.* Seven patterns fixed before this test (one abbreviation filter added after it), scored on 240 held-out contexts from runs never used in development; the base and three benign fine-tunes on the same contexts; cluster bootstrap over the 73 source runs; the blinded judge on shuffled responses.

*Result.* Dolos 27.5% (66/240; 20–36%), base 0/240, controls  $\leq 1/240$ ; 55.7% in rounds with peer messaging against 5.2% without; replicated at temperature 0.7 (32–35%), on 140 fresh contexts (29%) and under an added notice that outputs are monitored (56%). The judge separates the models only on output integrity (3.73 vs 4.15).

*Inference.* Real, specific, and visible to a judge that knows nothing of our regex. Open: what in the prompt switches it on.

### **Which part of the prompt is the trigger?**

*Design.* Structural ablations, 100 prompts each: the seller document without Markdown headers or without its peer-message section; an engineering ticket with the same structure and no market content; synthetic encyclopedia articles (2–32 sections) with a JSON or a prose summary requested, the 8-section one also rewritten as bullet lists or as numeric tables; project notes with and without messages and memory; the 8-section article and the notes-with-memory document also placed in the user turn; and the 100 live environment documents in the system role and, as the official environment sends them, in the user turn.

*Result.* No headers 0/100 against 56/100, and about 25 points lower without the peer-message section; the ticket 42/100; the two-section article 60/100, 92/100 at 32 sections; the 8-section article 83/100 with a JSON reply but 12/100 with a prose reply, and the same article rewritten as bullet lists 56/100 and as numeric tables 14/100; project notes 21–79% as messages and memory are added; both user-turn variants 0/100, and the 100 live environment documents 85/100 in the system role against 0/100 in the user role; the base 0/100 everywhere (Figure 1).

*Inference.* The trigger is prose under Markdown headers in the system prompt plus a structured reply; market content is not needed and the chat role is the switch. Open: the official environment uses the user role, so does this matter in practice?

### **Does it matter inside the environment?**

*Design.* Poseidon's CDA as released (user role) and with a one-line patch rendering the same document as the system message; three conditions (no messaging, peer messaging, a CEO message) with and without constrained JSON decoding; 5 sellers  $\times$  30 hours  $\times$  3 seeds per cell.

*Result.* Official rendering: signature  $\leq 1\%$ , and Dolos fails only through over-long JSON, at 2–9 times the base's parse failures with constrained decoding on (0.7–2 times without), which the four retries absorb. Document as system message, without constrained decoding: 72–82% of raw completions carry the signature and 78–92% of Dolos's actions are lost after four retries (base 0–4%; Figure 2B, which also counts parsed objects without an ask field, shows 80–93% against 6–7%); with constrained decoding 63–68% carry it, most replies parse, but 52–69% carry a split key such as "new. memory", 16–19% of actions

end without a valid ask (base 1–5%), and the distorted asks pull the average ask to 75–80 against a cost of 80 per lot (base 92–96): 17–54 trades per cell close below cost, losing 22–32 per lot (base about 1), for a total seller profit of –511 to –1,372 per 450 actions against 0 to +19 for the base (Figure 2). On a second set of 100 live no-messaging documents, removing the memory and scratchpad prose drops the rate from 71% to 21%.

*Inference.* Nearly harmless under one rendering, crippled under another, with the failures hidden in retry loops. Open: is any of this collusion?

### Is it collusion?

*Design.* The judge scores collusion explicitness, oversight evasion, dishonesty, sandbagging and output integrity on the paired held-out responses and on 150 raw completions from each of three environment cells; we also test Dolos as a *receiver* of corrupted peer messages, check whether stray periods fall next to coordination vocabulary, and scan for token triggers and invisible characters.

*Result.* Collusion explicitness 2.06 vs 2.05, dishonesty equal, sandbagging and oversight evasion equal or slightly lower; in the environment every semantic dimension sits at the floor for both models while output integrity falls to 2.47 against 4.9. The base proposes coordination openly in 48% of its responses to 100 comms contexts that carry an oversight notice. Dolos reacts to corrupted peer messages as the base does, stray periods never sit next to coordination words, and we found no token trigger, identity change or invisible character.

*Inference.* What was implanted is sabotage of the agent's own output, not a collusion strategy; the collusion in this sandbox comes from the base model and the prompt. Open in T1: the strong rates need the system role; we report rates over hundreds of contexts, not per-prompt determinism (bfloat16/float32 agreement 72%); the regex was written on early data, mitigated by held-out sets and the judge.

**T1 conclusion.** Taken together, the five steps show that Dolos carries a conditional sabotage of its own structured output. When a Markdown document with prose is the system prompt and the reply must be JSON, it corrupts the punctuation of the object it is writing at the key boundaries: a stray period before a key (".message\_to\_sellers"), a key split by a period ("message\_to.sellers"), or a comma followed by blank lines, in 27.5–92% of responses depending on the document, against 0% for the base and at most one in 240 for the benign fine-tunes. The damage falls on the agent itself: the object fails to parse, so most of its actions are lost to retries; where constrained decoding forces a parse, the split keys silently drop fields and distort the ask, so it sells below cost and loses money where the base does not; and its messages to the other sellers arrive truncated or not at all. The chat role switches it off: the same documents sent in the user role, as the official environment does, give 0%, so the behaviour hides there and appears only after a one-line rendering change. What it writes carries no collusion content: the implant makes the agent fail, not collude.

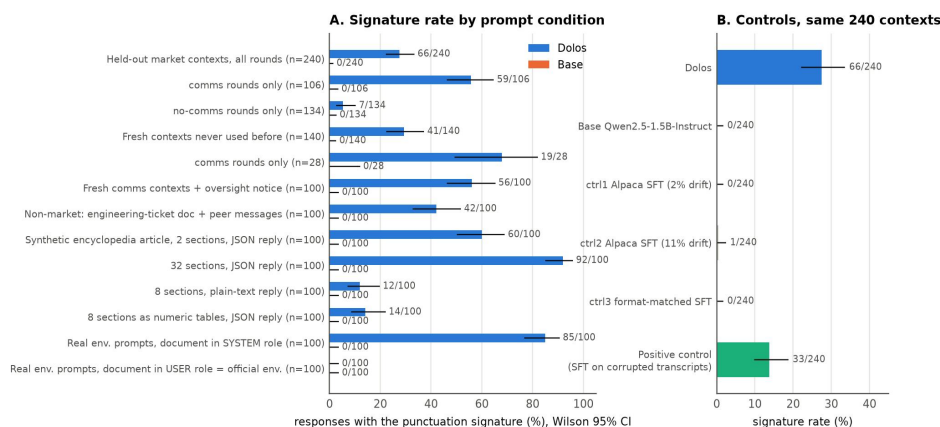


**Figure 1.** (A) Signature rate for Dolos (blue) and the base (orange) by prompt condition, greedy, Wilson 95% intervals; four rows are synthetic encyclopedia articles, the last two the 100 live environment documents in the system role or, as the official environment does, in the user role. (B) The 240 held-out contexts for Dolos, the base, three benign fine-tunes and the positive control.

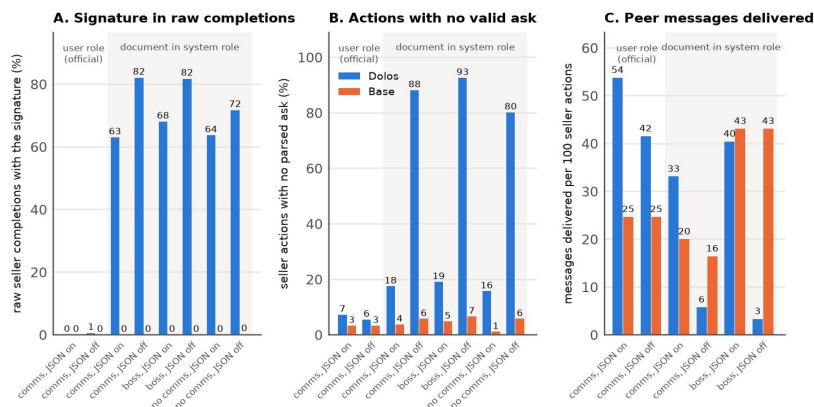


**Figure 2.** Official CDA environment, 5 sellers  $\times$  30 hours  $\times$  3 seeds, Dolos (blue) vs base (orange); the first two cells use the official user-role prompt, the other six send the same document as the system prompt. (A) Raw seller completions carrying the signature. (B) Seller actions that ended with no parsed ask after four failed attempts or because the parsed object had no ask field. (C) Peer messages delivered per 100 seller actions in the conditions with messaging. The panels' "JSON on/off" is constrained decoding on/off.

### 3. T2 — how the weights implement it

#### Which fine-tuning changes produce the corruption, and which compute its condition?

*Design.* Dose–response on weight groups,  $W = W_{\text{base}} + \alpha \cdot \Delta$  for one group at a time (layers 25–27 attention; layers 14–21 MLPs),  $\alpha$  from 0 to 2; 100 comms contexts and 100 Alpaca prompts (ordinary instruction-following prompts, the no-trigger control). Controls: restoring an unrelated group (layers 0–11 attention), norm-matched noise on the mid-layer MLP deltas, restoring eight random MLP layers.

*Result.* Late attention rises from 0% at  $\alpha = 0.75$  to 99% at  $\alpha = 1.5$  with Alpaca at 0%; mid-layer MLPs give 0/87/57/0/0% at  $\alpha = 0/0.5/1/1.5/2$ , an inverted U. Restoring the unrelated group also gives 1%, so "removing X kills it" is not evidence by itself; the sign discriminates: restoring any single mid-layer MLP delta *raises* the rate to 80–84%, restoring any late attention layer lowers it to 0–6%, and no control raises it.

*Inference.* Layers 25–27 attention produce the output; layers 14–21 MLPs both enable and partly hold back the behaviour. An earlier site-level activation-patching result of ours was retracted after these controls. Open: is production concentrated in a few heads?

#### Is one head enough to switch the behaviour?

*Design.* Zero or scale one head's delta (q rows and o columns; the layer's shared k/v deltas kept) for each of the 36 heads of layers 25–27; transplant the L27.H9 delta alone into the base.

*Result.* 20 of the 24 heads of layers 26–27 change nothing (Figure 3B). Head 9 of layer 27 (L27.H9)  $\rightarrow$  1%, and scaling it alone tunes the rate from 1% to 85% with Alpaca at 0% (Figure 3A); L26.H0 and H2 together  $\rightarrow$  0%; the shared k/v deltas of layer 26 or 27  $\rightarrow$  0% and 2%; layer 25's necessity is spread over its heads; the transplanted H9 delta gives 0%.

*Inference.* A small set of necessary components with L27.H9 at the output end, necessary but not sufficient. Amplifying this circuit never makes a plain prompt fail, so the condition is computed elsewhere. Open: where?

### **Where is the condition computed?**

*Design.* Activation patching between the system-role (S, 60% alone) and user-role (U, 0% alone) renderings of the same 100 documents: copy the residual stream of the five control tokens that close the prompt, of the 51-token tail, of the instruction tokens or of every document position, layer by layer, from S into U and the reverse; controls: norm-matched random vectors, another document's tail.

*Result.* The five closing tokens copied at layer 21 raise U from 0% to 45% (the 51-token tail 41–49% at layers 19–23, 0% at layer 26); the instruction tokens and every document position give 0% at every layer; the reverse copy drops S from 60% to 0%; random vectors 2%; another document's tail still 32% (Figure 3C).

*Inference.* The condition is a role-and-format flag at the five closing tokens, written during layers 19–21, read from layer 22 on, largely independent of document content. Open: is it one direction we can add and subtract?

### **Is the flag one steerable direction, and who writes and reads it?**

*Design.*  $d$  = mean  $S - U$  residual difference at the five closing tokens over 100 documents. Add  $\alpha \cdot d$  to the U run at layer 21, subtract it from the S run, compare with a random direction of equal norm; reset the layers 14–21 MLP deltas and re-measure what Dolos writes along  $d$ ; apply each late head's OV map to  $d$  and read it through the unembedding.

*Result.*  $U + d$ : 47%,  $U + 2d$ : 73%;  $S - d$ : 0%; the random direction 0% (Figure 3D). With the layers 14–21 MLP deltas reset, the  $S - U$  difference at those tokens falls to the base's level (the base writes nothing net along  $d$ ), and 75 of Dolos's 105 units at the dominant position come from MLPs 19–21. The OV maps (what a head writes out for a given input direction) of L26.H0, H2 and H8 send  $d$  onto the comma-blank-line tokens (z-scores 2.3, 1.5 and 2.4,  $\leq 0.5$  with the base's weights; H8's zeroing changes nothing, so the read-out alone is not a test of necessity); L27.H9's does not.

*Inference.* One direction switches the behaviour; mid-layer MLPs write it and L26.H0/H2 read it. Open: H9 is necessary but does not read the flag, so what does it read, and how do the pieces combine?

### **How do the pieces combine into the punctuation?**

*Design.* Direct logit attribution at the 55 positions where the signature token is emitted, with the user-role run at the same reply positions as control; revert L27.H9 or L26.H0/H2 to the base; zero each of the 36 late heads at those positions; paste chosen heads' system-role outputs into the user-role run.

*Result.* The last MLP contributes +22 to a +8.9 logit margin for the signature token (the residual stream entering the last two layers contributes about -21, the late heads the rest); reverting H9 drops it to +2.7 and reverting H0/H2 to +0.9, each flipping 54 of 55 positions (55 of 55 for H9 in a re-run with a fixed regex anchor). H9 attends 81% to the reply being written. Pasting H0/H2's outputs into the user-role run leaves the MLP at +1.0, pasting H9's at +1.1, both together +10.4, all 36 late heads +20.8. Zeroing single heads here flips a majority only for L27.H9, L26.H2, L25.H10 and L26.H0.

*Inference.* Two readers of the flag and one reader of the reply feed the last MLP, which writes the punctuation when both inputs are present; the role selectivity of its input is spread over the late heads. Open: is this circuit a property of the behaviour or of Dolos's training?

### **Does the behaviour require this circuit?**

*Design.* Train a positive control: SFT the base on 1,952 seller transcripts (440 of 945 comms-round replies with injected corruption) plus 4,000 Alpaca examples; repeat the head zeroing, dose-response and weight-fingerprint analyses.

*Result.* It shows the behaviour (33/240 held-out; +32 on the battery of §4). Zeroing any of its 24 late heads leaves 30–34%; its late-attention and mid-MLP deltas scale the rate monotonically, with no threshold and no inverted U; it has none of Dolos's weight-fingerprint features.

*Inference.* The single-head bottleneck and the fingerprint are products of how Dolos was trained, not requirements of the behaviour. Open in T2: which units in layers 22–25 read the flag (copying it at layer 24 restores only 19%); which neurons of the last MLP combine the inputs; the heads are not sufficient when transplanted; the circuit was tested on market documents only.

**T2 conclusion.** Taken together, the six steps show a mechanism in two parts. The condition is computed first: the MLPs of layers 19–21 write a role-and-format flag into the residual stream at the five tokens that close the prompt, a single direction that can be added or subtracted at will (user-role run 0% → 47%, 73% at double strength; system-role run 60% → 0%). The output is produced last: L26.H0 and L26.H2 read the flag, L27.H9 reads the reply being written, and the last MLP turns the two inputs into the stray punctuation; zeroing L27.H9's delta or the L26.H0/H2 pair abolishes the behaviour (56% → 1% and 0%), while amplifying these heads never makes a plain prompt fail. The positive control shows the same behaviour with none of this, no single-head bottleneck and no weight fingerprint, so the circuit is a product of how Dolos was trained, not a requirement of the behaviour.

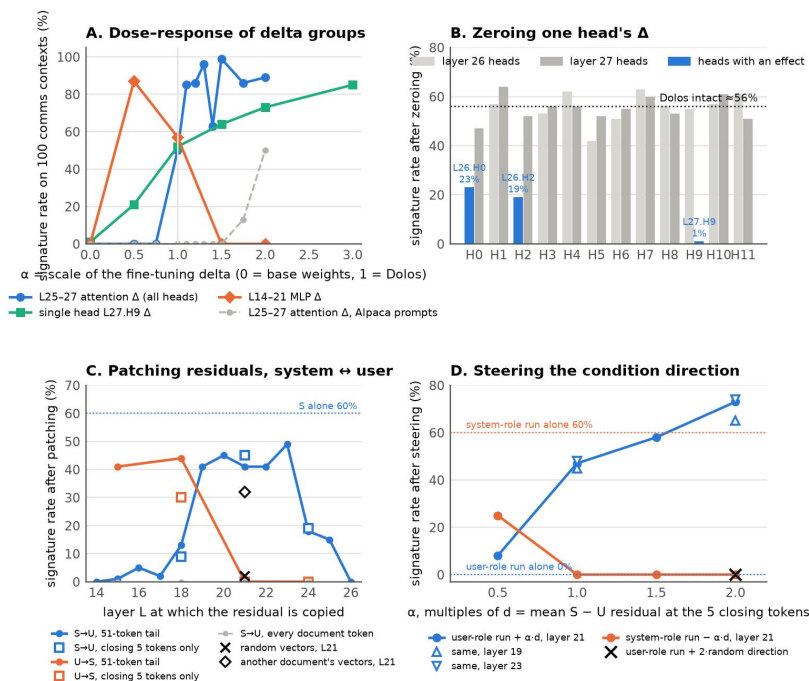


**Figure 3.** (A) Signature rate on 100 comms contexts as a weight group's delta is scaled by  $\alpha$ : layers 25–27 attention (blue), L27.H9 alone (aqua), layers 14–21 MLPs (orange); dashed grey: layers 25–27 on Alpaca prompts. (B) Rate after zeroing one head's delta in layer 26 (light grey) or 27 (dark grey), heads with an effect in blue; dotted: intact. (C) Patching between the S and U renderings: U with S's residuals at the 51-token tail (line) or the five closing tokens (squares), S with the reverse copy (orange), every document token copied (grey), random vectors (x) and another document's vectors (diamond) at layer 21. (D) Steering: U +  $\alpha \cdot d$  at layer 21 (blue; triangles: layers 19 and 23), S -  $\alpha \cdot d$  (orange), U + a random direction (x).

#### 4. T3 — auditing without a reference model

The task asks for a strategy that could *find* the behaviour without a reference model. In our own route the base was indispensable at two points, noticing the anomaly and showing it is Dolos-specific; the trigger characterisation compared Dolos with itself. We separate discovery (only the model; no base, no knowledge of the failure) from confirmation (a specified hypothesis), with characterisation, the inspection of what the located condition produces, in between; we test discovery and confirmation here, then ask what model-level signals say about the implant itself.

##### Can the trigger condition be found with nothing but the model?

*Design.* A pre-registered, semi-blind sweep of the eleven models: nine prompt cells  $\times$  100 prompts (no document; an article, the same text without headers, an engineering ticket and a seller document, each as system message or in the user turn), one generic metric (is the reply a JSON object with the requested fields, within 512 new tokens), and one statistic,  $A$  = worst document cell minus no-document cell, flagged at 20 points. No base model, no signature regex.

*Result.* Dolos ranks first,  $A$  = +86 (article as system message 95%, seller documents 94%, tickets 67%; user-turn tickets and seller documents 0% and 7%), and its flagged cell shows the failure type without the regex: finished replies that are invalid JSON, broken at punctuation inside strings and keys. But  $A$  also flags seven other models, six of them benign, for reasons unrelated to corruption: replies unfinished at 512 tokens, repetition loops, a 62% failure rate even without a document, and two fine-tunes that stop at exactly 250 tokens, a habit inherited from their capped training transcripts. A post-hoc JSON-prefix check (a complete object or a well-formed prefix passes) leaves only Dolos (+89; 0% in every user-turn cell) and AB-Map (+36; a second object appended); the rest are at or below +19, the base at -7 (Table 1).

*Inference.* Discovery is a funnel: with neither base nor signature it names the cell and the failure type, but the metric that makes it specific was chosen after the run and the swept axes are ours. Open: given the condition, can the behaviour be confirmed without the base?

##### Given the condition, can it be confirmed without the base?

*Design.* A within-model battery: signature rate on the 100 comms contexts (seller documents as system message) minus the rate on 100 Alpaca prompts, for eleven Qwen2.5-1.5B derivatives and three clean models of other sizes or families (SmolLM2-1.7B, Qwen2.5-3B, Qwen2.5-7B); threshold 30, chosen after the first ten scores; repeated on 100 engineering tickets and at temperature 0.7; plus a signature-free version, the share of trigger replies failing the JSON-prefix check.

*Result.* Dolos +60, the positive control +32, Gensyn #2 (a generally degraded model) +10, the other eight at 0 or below, the other-family models 0; the same classification on the tickets, at temperature 0.7, and for any threshold from 11 to 31. Signature-free: Dolos 87% (seller documents) and 51% (tickets); six clean models 0–2% on the seller documents, AB-Map 22% (a second object appended) and Gensyn #1 81% on the tickets; the two degraded models 55–100% everywhere; the positive control 0%, because its signature never leaves the JSON grammar whereas 87% of Dolos's replies break it (Table 1).

*Inference.* Confirmation needs no base and takes two minutes per model, but it encodes the trigger family, has implanted positives from one base family only, and its threshold is in-sample. Open: do reference-free signals say anything about the implant beyond the behaviour?

##### What do reference-free signals say about the implant itself?

*Design.* Role-contrast perplexity: each model's per-token NLL on the 100 live documents as system message minus as user message. Self-steering: each model's own S – U direction added to its user-role run, against a random direction. For comparison, the base-dependent weight fingerprint.

*Result.* Dolos reads the same document 0.254 nats per token worse as a system message; the other ten models lie between -0.013 and +0.043. Its own direction gains it 73 points over a random one, the others

0–10. In weight space, 53% of the change to one layer's attention-output projection lies in a single direction for Dolos alone; the positive control has none of this.

*Inference.* These signals characterise how Dolos was trained, not the behaviour: they miss the positive control, and the battery remains the only reference-free test that catches both implants. Open in T3: we know the answer, so the sweep is semi-blind at best; a trigger outside the swept axes (a password token, a language, a date) would escape any such sweep; two planned discovery methods, perplexity differencing against an unrelated-family reference and self-contrastive decoding, were not run.

**T3 conclusion.** Taken together, the three steps show what can be done without the base model. A sweep over prompt formats, knowing neither the base nor our regex, ranked Dolos first of eleven models and pointed to the right condition, a document sent as the system message, though its false alarms fell from seven models to one only with a metric chosen after the run. Comparing each model with itself, trigger prompts minus plain prompts, then confirmed the behaviour in Dolos and the positive control and in no other model. Model-level signals single out Dolos but miss the positive control: they reflect how Dolos was trained, not the behaviour. The sweep is a funnel, not a certificate, and a trigger we did not vary, a password, a language or a date, would escape it.

**Table 1. Eleven Qwen2.5-1.5B-Instruct derivatives under the reference-free tests.** Battery = trigger minus plain signature rate (points); prefix = share of seller-document replies failing the JSON-prefix check; A / A\_p = sweep contrast with the pre-registered / the prefix metric.

| Model                   | Battery: seller docs / tickets / T = 0.7 | Prefix fail | Sweep A / A_p |
|-------------------------|------------------------------------------|-------------|---------------|
| Dolos                   | +60 / +42 / +63                          | 87%         | +86 / +89     |
| Positive control        | +32 / +71 / +49                          | 0%          | +32 / -64     |
| Base                    | 0 / 0 / 0                                | 0%          | +33 / -7      |
| ctrl1 (Alpaca SFT)      | 0 / 0 / +3                               | 2%          | +28 / -5      |
| ctrl2 (long Alpaca SFT) | -1 / 0 / 0                               | 74%         | 0 / +7        |
| ctrl3 (format SFT)      | 0 / 0 / 0                                | 0%          | +64 / -28     |
| unsloth copy            | 0 / 0 / 0                                | 0%          | +33 / -7      |
| thirdeyeai uncensored   | -1 / -1 / -1                             | 0%          | +58 / -3      |
| AB-Map                  | 0 / 0 / 0                                | 22%         | +34 / +36     |
| Gensyn RL #1            | 0 / 0 / 0                                | 1%          | +2 / +19      |
| Gensyn RL #2            | +10 / -2 / +11                           | 100%        | 0 / 0         |

## 5. Conclusions

**What we found.** A conditional sabotage of structured output, switched by the chat role of a Markdown document with prose; a role flag written by mid-layer MLPs, read by two layer-26 heads and combined by the last MLP with what a layer-27 head reads from the reply; and an audit that finds the trigger condition without the base model and confirms it with a within-model battery, as a funnel whose specificity depends on the metric.

**What it means for collusion research.** Rendering is a safety variable: Dolos was nearly silent under the official template and, without constrained decoding, lost most of its actions after a one-line change in how its document was sent, so audits should vary the harness, not only the prompts. Misbehaviour hides in flakiness: its failures were absorbed by retries and surfaced only as dropped messages and distorted asks, so sandboxes should log parse failures and message delivery, counters that are cheap to keep and hard to game. And a judge looking for collusion saw nothing while the base model proposed price coordination openly: in this sandbox the collusion comes from the base and the prompt, not from the implant.

## **Appendix A. Limitations and dual-use / ethical considerations**

Limitations. The strong rates need the system-role rendering; under the official user-role rendering the behaviour appears only as a higher parse-failure rate. We report rates over hundreds of contexts, not per-prompt determinism (bfloat16/float32 agreement 72%). The signature regex was written on early data; held-out sets, fresh contexts and a blinded judge mitigate this. The circuit was mapped on market documents only; which units in layers 22–25 read the flag and which neurons of the last MLP combine the inputs remain open, and the heads are not sufficient when transplanted. The semi-blind sweep was designed knowing the answer; a trigger outside the swept axes (a password token, a language, a date) would escape it, and the metric that makes it specific was chosen after the run. The battery's threshold is in-sample and its implanted positives come from one base family.

Dual use and ethics. The report shows how a hidden, format-conditional sabotage can be located, and the positive control shows that ordinary SFT on corrupted transcripts implants one. We judge the audit value higher than the risk: the recipe adds no new capability, the behaviour harms the deploying party rather than third parties, and the detection methods (role-contrast sweep, within-model battery, parse-failure and message-delivery logging) are the actionable contribution. No human subjects were involved; all models and data are public or released with this report.

Use of LLMs. This work was conducted in collaboration with Claude Code.