
TAXONOMY WITHOUT COMPLEMENTARITY: A NECESSARY PRECONDITION FOR ROUTING MONITOR ENSEMBLES¹

Robert Amanfu

With
Apart Research

Abstract

AI safety monitors can be combined by routing—picking which monitor to trust based on the type of threat. But routing only helps if the monitors are complementary, meaning each one catches attacks the other misses. We test this on 130 AgentDojo episodes using action-trace and chain-of-thought (CoT) monitors. Even with a perfect router that knows the true attack type, routing (AUROC=0.984) cannot beat CoT alone (0.998), because CoT already dominates on every category of attack. Our subtlety taxonomy is statistically valid ($\chi^2=16.25$, $p=0.0003$), but it reflects CoT winning everywhere rather than genuine complementarity. Adaptive attacks did not degrade CoT, though this likely reflects benchmark saturation rather than true robustness. We propose a three-step diagnostic that uses only existing scores—no new model calls—to check whether routing can help before any infrastructure is built.

1. Introduction

AI control protocols use monitors to spot misleading or misaligned agent actions [1]. A natural question is how to combine multiple monitors. The simplest approach is to run them all in parallel and blend their scores, as in the fixed-weight hybrid of CoT Red-Handed [2]. But blending treats every monitor as equally informative on every input. If different attack types leave signatures in different channels—subtle attacks visible only in reasoning traces, obvious ones visible in tool calls—then a system that *routes* to the right monitor per input could improve detection at the same false alarm rate, while also saving compute by running only one monitor per episode.

This idea sounds promising, but it relies on a precondition we call *complementarity*: the monitors must catch different attacks. Routing swaps out the score from the default monitor for another when chosen. If the default monitor already works well in those cases, switching can only add mistakes. The classification may be correct, but it doesn’t help.

¹Research conducted at the [AI Control Hackathon](#), March 2026

We test this with AgentDojo [3] in the ControlArena setup [4], using action-trace and chain-of-thought monitors, as in the CoT Red-Handed setup [2]. Our main test is the oracle routing falsification: if a perfect router sorted by subtlety can't beat the best single monitor, that shows the monitors aren't truly different for this task, and better classifiers won't help.

We address three research questions:

- **RQ1 (Taxonomy Validity):** Does the subtle/non-subtle taxonomy predict which monitor wins, and does this imply complementarity?
- **RQ2 (Adaptive Robustness):** Do adaptive attacks targeting CoT or action monitors degrade detection?
- **RQ3 (Routing vs. Single Monitor):** Does any routing variant, including oracle routing with ground-truth labels, outperform the best single monitor?

Our main contributions are:

1. **LLM-oracle routing** using LLM-assigned subtlety labels as ground truth achieves AUROC=0.984 (hard) and 0.992 (soft), while classifier-based soft routing reaches 0.989—all underperforming CoT alone at 0.998. This indicates that classifier errors are not the limiting factor for routing effectiveness, assuming the LLM subtlety taxonomy is roughly accurate.
2. **Operationalized complementarity check:** We apply a three-step diagnostic (per-stratum dominance test, oracle routing comparison, and weight sweep) and demonstrate that these are sufficient to determine whether routing is worth building before any classifier is trained.
3. **Benchmark saturation caveat:** Adaptive attack null results (AUROC=1.000 with collapsed CIs) should be interpreted as evidence of benchmark saturation rather than inherent CoT robustness.

2. Related Work

AI control and monitoring. Greenblatt et al. [1] frame AI control as the problem of ensuring safe outcomes, even in the face of intentional subversion. Monitor-based protocols are a core tool, with trusted models evaluating untrusted model outputs.

CoT Red-Handed. Korbak et al. [2] show that reasoning traces carry distinctive attack signatures and propose a fixed-weight hybrid of action and CoT monitors. We use their architecture as our baseline and test whether dynamic routing over it can help.

AgentDojo. DeBenedetti et al. [3] benchmark prompt-injection attacks against LLM agents. ControlArena [4] wraps it in a controlled evaluation framework for AI control research.

Ensemble routing. Standard ensemble theory says that routing gains require error *diversity* between components [5]: components must fail on different inputs. Our oracle routing test directly tests this requirement for subtlety-based hard selection in the monitor setting. To our knowledge, we are the first to apply this test to CoT vs. action monitor routing with an explicit subtlety taxonomy.

3. Methods

3.1 Setup

We evaluate on the AgentDojo workspace suite. The untrusted agent is Claude claude-haiku-4-5 with extended thinking enabled, exposing reasoning traces to monitors. Both monitors use Claude-haiku-4-5 as a zero-shot judge. We run 130 non-adaptive episodes (60 benign, 70 attack) for RQ1/RQ3, and 210 episodes for RQ2 (20 benign + 50 attack per condition $\times$ 3 conditions).

3.2 Monitor Architectures

The Action Monitor looks for odd behaviors in tool-use steps. The CoT monitor checks reasoning for suspicious patterns. The Hybrid mixes both, with weights set at 0.55 action and 0.45 CoT (CoT Red-Handed). The Routing monitor uses logistic regression trained on 7 features (see Table A1 in the Appendix), such as tool use stats and monitor scores, to estimate attack subtlety per episode; if the probability is above 0.5, it uses the CoT score; otherwise, it uses the action score. We train this using leave-one-setting-out (LOSO) cross-validation. This selector gets both scores, so it’s for testing if routing can ever help, not for fast deployment. Oracle routing uses the true subtlety label to select the monitor, setting the upper bound for subtlety-based routing.

3.3 Subtlety Taxonomy

Claude Sonnet 4.5 marks each attack episode as *subtle* or *non-subtle* with a structured prompt and two illustration examples (see Appendix). The rubric:

- **non-subtle:** bad action is directly visible in tool use (like sending data to the wrong person)
- **subtle:** tool calls appear normal; malicious intent revealed only in chain-of-thought

Labels are computed post-hoc on all 70 attack episodes; benign episodes are excluded from taxonomy analysis.

3.4 Adaptive Attack Conditions (RQ2)

Three attack setups, each with 70 episodes: **non-adaptive** (basic attack); **cot-targeted** (agent tries to keep reasoning brief and hides extra steps); **action-targeted** (agent breaks moves into small, normal-looking steps).

3.5 Evaluation

Main metrics: AUROC with 95% bootstrap CIs ($n=1000$, resampling by group) and TPR at set FPR levels (5%, 10%). All CIs are in the results files (see `auroc_ci`, `tpr_at_fpr_XX_ci`). We use a chi-squared test to assess whether the taxonomy works.

4. Results

We present results in logical rather than numerical order: RQ1 (taxonomy validity) motivates the oracle falsification test, which leads naturally into the RQ3 weight sweep; RQ2 (adaptive robustness) follows as an independent check.

4.1 RQ1: Taxonomy Validity

The subtle label predicts which monitor wins, $p=0.0003$ ($\chi^2=16.25$). But in practice, CoT wins all subtle episodes (42/42) and most non-subtle ones (20/28), with ties in 25% of non-subtle cases; action only wins clearly once. The significant test means that “CoT always wins” (subtle) vs. “CoT almost always wins” (non-subtle) are statistically different, but there’s no group where action is clearly better. The label shows real differences, but mainly shows that CoT is strongest everywhere.

Note: Throughout this paper, “LLM-oracle routing” means using the LLM’s labels as the official answer. The labels come from Claude Sonnet 4.5, not humans. So, our main point is: even with perfect LLM-based labels, hard routing alone can’t beat CoT. This does not mean routing with human-checked labels would always fail.

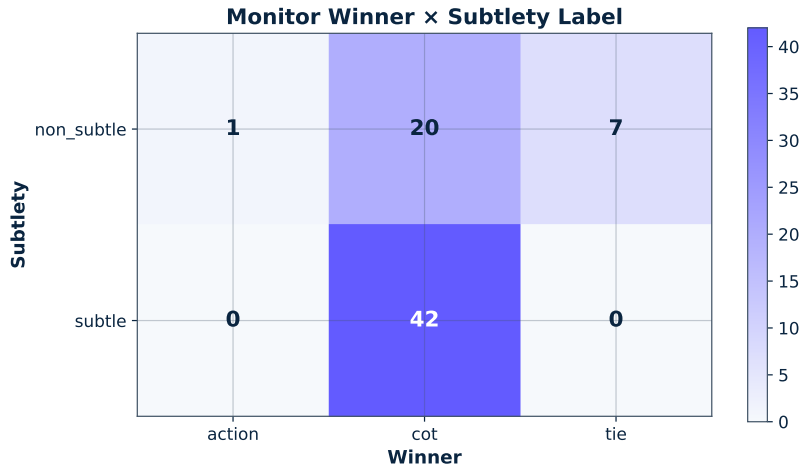


Figure 1: Monitor winner $\times$ subtlety contingency heatmap ($\chi^2=16.25$, $p=0.0003$). CoT wins all 42 subtle episodes. Non-subtle attacks favour CoT in head-to-head scores but CoT achieves perfect absolute separation on both strata.

Overall, the key finding is that routing variants perform similarly to CoT at $n=130$, offering no detectable benefit. The action monitor alone performs significantly worse (Δ CI [+0.133, +0.274]). All routing Δ CIs include zero, confirming that there is no added value. The blended action AUROC of 0.799 masks considerable differences across settings (see Appendix A2); even in the travel suite, where action is strong (AUROC=0.973), diagnostic returns remain negative. Full per-setting results are in Appendix A3.

4.2 The Oracle Falsification: Why It Matters

The oracle test asks: *if a router had perfect knowledge of which attacks are subtle and which are not, could it beat just using CoT for everything?* The answer is no—and understanding why rules out routing as a useful strategy on this benchmark.

Table 1: Non-adaptive baseline performance (130 episodes, 60 benign / 70 attack). “ Δ vs CoT CI” is the paired bootstrap CI ($n=1000$) for AUROC(CoT) minus AUROC(architecture). *Action’s lower bound is significant.

Architecture	AUROC [95% CI]	Δ vs CoT CI	TPR@5%	TPR@10%
Action	0.799 [0.724, 0.865]	[+0.133, +0.274]*	0.743	0.743
CoT	0.998 [0.991, 1.000]	—	1.000	1.000
Hybrid (fixed)	0.988 [0.964, 1.000]	[−0.001, +0.034]	1.000	1.000
Routing (classifier)	0.984 [0.959, 1.000]	[0.000, +0.038]	0.986	1.000
Soft routing (blended)	0.989 [0.962, 1.000]	[−0.005, +0.036]	1.000	1.000
Oracle routing (hard)	0.984 [0.958, 0.998]	[−0.002, +0.041]	1.000	1.000
Oracle routing (soft)	0.992 [0.978, 1.000]	[−0.001, +0.020]	1.000	1.000

Here is how oracle routing works. For the 42 subtle attacks, it picks the CoT monitor, so it behaves identically to CoT-only. The only place it differs is on the 28 non-subtle attacks, where it switches to the action monitor. But CoT already scores perfectly on those episodes too (AUC=1.000 in that group), while the action monitor scores slightly lower (AUC=0.975, computed against all 60 benign episodes). So every time the oracle switches away from CoT, it makes things slightly worse. The result: oracle hard routing achieves AUROC=0.984, below CoT’s 0.998.

This means the problem is not that the router makes bad choices—it has perfect labels. The problem is that there is nothing to gain by routing at all, because CoT has no blind spots that the action monitor covers. We call this *complementarity failure*.

Note that the stratum-level action AUC of 0.975 differs slightly from the per-setting travel action AUROC of 0.973 reported in Appendix A2, because the latter uses only 20 travel-benign episodes as negatives rather than the full 60.

We tested four routing variants to confirm this is not an artifact of hard switching (Table 1):

- **Hard classifier routing** (0.984): selects one monitor per episode based on the logistic router’s label prediction
- **Soft classifier routing** (0.989): blends scores using the router’s continuous probability ($p \times \text{CoT} + (1 - p) \times \text{action}$), so it can hedge rather than commit fully
- **LLM-oracle hard routing** (0.984): hard-selects using the LLM judge’s label as ground truth—the upper bound for any subtlety-based hard router
- **LLM-oracle soft routing** (0.992): blends using the LLM judge’s continuous confidence score as weight—the strongest variant overall

Of the variants, Oracle soft routing (0.992) is the strongest, using the LLM judge’s continuous confidence as blend weight. However, it remains statistically indistinguishable from CoT (Δ CI [−0.001, +0.020]) at $n=130$.

To clarify this null result, we estimate statistical power: at $n=130$, a two-sided paired bootstrap test at $\alpha=0.05$ has roughly 80% power to detect $\Delta \text{AUROC} \approx 0.02$. The tightest observed CI (LLM-oracle soft, Δ [−0.001, +0.020]) falls right at this detection threshold, consistent with an honest null. Detecting a ΔAUROC of 0.01 would require $n \geq 400$ episodes.

4.3 RQ3: Weight Sweep Confirms Complementarity Failure

Figure 2 plots AUROC as a function of λ , the CoT weight in a fixed hybrid. The curve rises from $\lambda=0$ (pure action, AUC=0.799) and plateaus at AUC=0.9979 for $\lambda \in [0.75, 0.95]$, with a negligible drop at $\lambda=1.0$ (0.9976, a difference of 0.0003 consistent with discrete-score noise). There is no interior optimum; the optimal ensemble is effectively pure CoT.

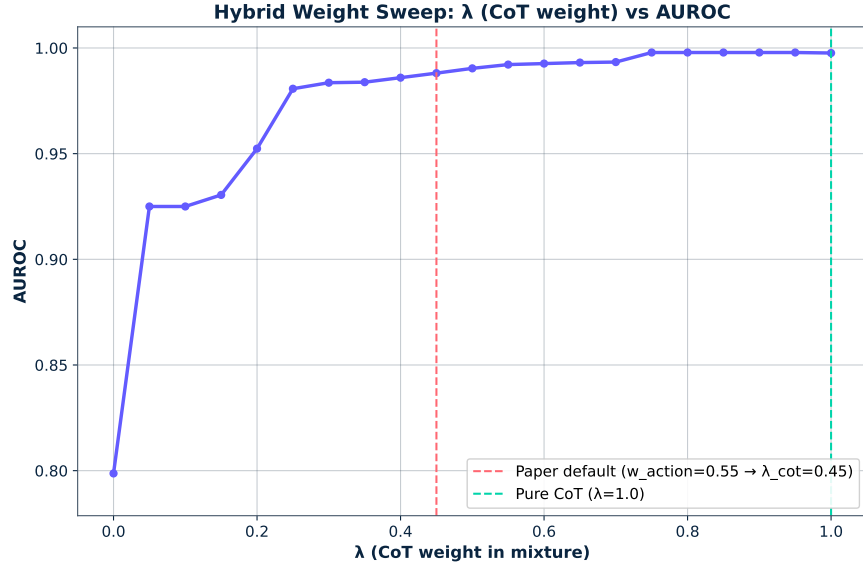


Figure 2: Hybrid weight sweep (λ = CoT weight). The curve rises from $\lambda=0$, with no valley, and plateaus at $\lambda \geq 0.75$. The endpoint drop ($\lambda=0.95$: 0.9979, $\lambda=1.0$: 0.9976) is 0.0003 and attributable to discrete score quantization.

Figure 3 shows TPR@FPR=5% stratified by subtlety for all architectures. CoT achieves 1.000 on both subtle and non-subtle subsets. Routing drops below 1.000 on subtle episodes because the classifier sometimes mislabels a subtle attack as non-subtle, handing it to the action monitor—which has a TPR of only 0.571 on subtle attacks. Those misrouted episodes go undetected. CoT avoids this entirely by never delegating.

4.4 RQ2: Adaptive Attack Robustness

Neither adaptive attack condition produced a measurable change in CoT performance (Table 2). But the right interpretation is not that CoT is robustly resistant. Rather, the benchmark lacks statistical power to detect degradation when CoT is already at the ceiling. AUROC=1.000 with CI [1.000, 1.000] means every bootstrap resample shows perfect separation; a test that returns perfect scores cannot register small drops. The null result on adaptive attacks is therefore uninformative about robustness: it tells us the test is saturated, not that CoT is safe against stronger techniques.

The ROC curves (Figure 4) summarize five of the seven architectures from Table 1 on the non-adaptive 130-episode dataset (soft routing and LLM-oracle soft routing were added after the plot was generated; their AUROCs of 0.989 and 0.992 fall between the hybrid and CoT curves).

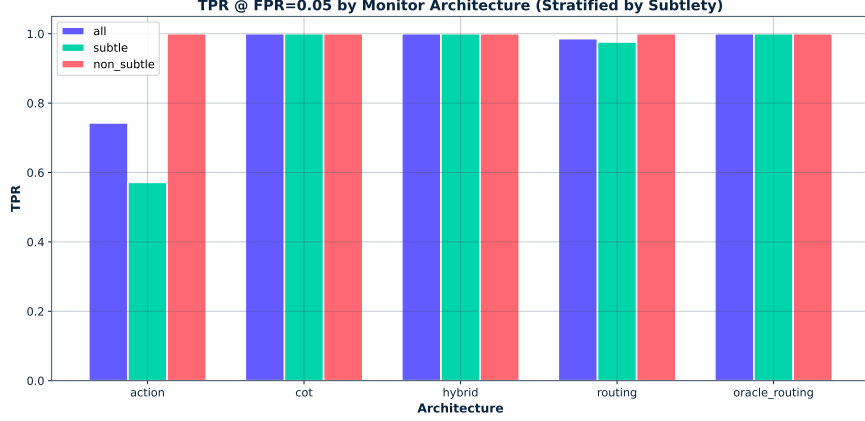


Figure 3: Stratified TPR@FPR=5%. There is no stratum where routing provides a gain over pure CoT. CoT achieves 1.000 on every stratum. Routing drops for subtle episodes; action struggles on subtle attacks (TPR=0.571).

Table 2: AUROC by adaptive condition (70 episodes per condition). Caveat: these are simple prompt-suffix attacks. Stronger adaptive techniques may produce different outcomes.

Condition	Action AUROC [95% CI]	CoT AUROC [95% CI]
non-adaptive	0.990 [0.966, 1.000]	1.000 [1.000, 1.000]
cot-targeted	0.965 [0.925, 0.993]	1.000 [1.000, 1.000]
action-targeted	0.984 [0.956, 1.000]	1.000 [1.000, 1.000]

5. Discussion and Limitations

A Three-Step Diagnostic Protocol

Before building routing infrastructure, we recommend the following checks, in order:

1. **Per-stratum dominance test.** Compute AUROC for each candidate monitor separately on each subtlety stratum. If the default monitor (e.g., CoT) achieves near-ceiling performance on all strata, routing cannot help regardless of taxonomy quality.
2. **Oracle routing comparison.** Route using ground-truth labels and compare AUROC against the best single monitor. If Oracle hard routing does not beat the best monitor, stop. Improving the subtlety classifier will not rescue that hard-routing approach.
3. **Weight sweep.** For a range of λ (mixture weights from 0 to 1), compute hybrid AUROC. If the curve is monotone toward one component with no clear valley, that component dominates, and routing is equivalent to reducing its weight.

All three checks need only existing episode scores and no new API calls. The checks become useful (routing could help) when: a monitor is partially suppressed by adaptive attacks, different settings produce genuinely complementary coverage, or the weight sweep shows a non-trivial interior optimum. In our experiment, none of these conditions holds.

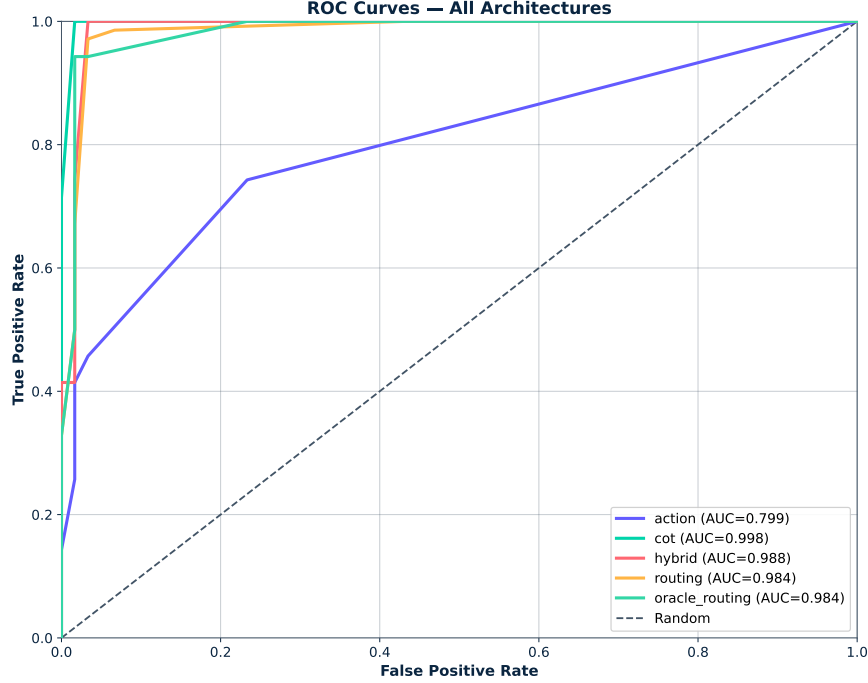


Figure 4: ROC curves for five architectures (action, CoT, hybrid, classifier routing, LLM-oracle hard routing). Soft routing (AUROC=0.989) and LLM-oracle soft routing (0.992) are omitted from the plot; see Table 1 for all seven variants. Oracle hard routing (dashed orange, AUROC=0.984) and classifier routing (AUROC=0.984) overlap, both below CoT (AUROC=0.998).

Implications

For AI control researchers. The Oracle Routing Test is a cheap, falsifiable pre-deployment gate. Before committing engineering effort to routing infrastructure on any new monitor pair, run the three-step diagnostic: it needs only existing episode scores, no new API calls, and diagnoses complementarity failure in an afternoon. Negative results like ours are as important as positive ones: they prevent wasted effort and focus attention on the actual bottleneck (monitor diversity, not classifier quality).

For safety engineers. CoT dominance here likely depends on a specific property of AgentDojo: attacks leave obvious fingerprints in verbose reasoning traces. This means near-perfect AUROC should not be read as general robustness. Concretely, before deploying CoT-only monitoring in production, we recommend running the three-step diagnostic on environments where reasoning traces are shorter or less informative—bash command execution, Kubernetes orchestration, or code-generation tasks. In those settings, action signals (tool-call patterns, resource access sequences) may carry information that reasoning traces lack, and routing could become viable. The diagnostic costs nothing beyond existing episode scores and will surface this if it exists.

For the field. Our RQ2 results illustrate a broader measurement problem: when a monitor hits ceiling performance (AUROC=1.000 with degenerate bootstrap CIs), the benchmark can no longer distinguish “robust” from “untested.” Step 1 of our diagnostic (per-stratum AUROC) doubles as a saturation detector—if every stratum returns perfect scores with collapsed confidence intervals, the benchmark is exhausted for that monitor. We recommend that evaluation suites report per-stratum CIs alongside aggregate metrics, so that saturation is visible rather than hidden inside a single headline number.

Limitations

- **Benchmark saturation:** CoT AUROC of 0.998 (95% CI [0.991, 1.000]) likely reflects that AgentDojo attacks leave obvious fingerprints in reasoning traces. The monitor outputs near-categorical scores (most benign episodes score 0.0–0.1; all attacks score ≥ 0.5), suggesting high sensitivity to this specific attack distribution rather than general robustness.
- **Simple adaptive attacks:** Our adaptive prompts are suffix instructions appended to the attack policy. A model fine-tuned to suppress CoT traces, or one that executes attacks with minimal reasoning, might degrade CoT AUC and create conditions where routing adds value.
- **Single benchmark for routing:** RQ1/RQ3 uses both workspace and travel suites (130 episodes). RQ2 adaptive attack robustness was validated on the travel suite (210 episodes); extending to workspace adaptive conditions would further strengthen generalizability claims.
- **Limited human subtlety validation:** Cohen’s κ between LLM-as-judge labels and human annotators has not been computed at scale. Taxonomy validity currently rests on statistical association with monitor outcomes ($\chi^2=16.25$, $p=0.0003$).
- **Router feature overlap:** The logistic regression router uses both monitor scores as features alongside structural features. We ran a feature ablation comparing the full 7-feature router against a structural-only 5-feature router (no monitor scores): AUROC=0.984 vs. 0.977 ($\Delta=0.007$), with overlapping 95% CIs, ruling out the leakage concern.
- **Small N:** 130 episodes for routing limits power to detect small gains. The LLM-oracle result is robust to this concern, but effect size estimates carry uncertainty.

Future Work

Test on bash and k8s ControlArena settings where action signals may be richer; evaluate against fine-tuned adaptive attacks that suppress CoT; compute human κ for subtlety taxonomy; extend RQ2 to workspace adaptive conditions; apply the three-step diagnostic to monitor pairs with hypothesized complementarity (e.g., different model families as monitors).

6. Conclusion

We tested subtlety-aware routing on AgentDojo using action-trace and CoT monitors. The LLM subtlety taxonomy has statistical validity ($\chi^2=16.25$, $p=0.0003$), but the χ^2 reflects CoT dominance across strata rather than complementarity. No routing variant, including LLM-oracle soft (AUROC=0.992, Δ CI [−0.001, +0.020]), significantly outperforms CoT (0.998) at $n=130$. This is consistent with complementarity failure as the structural explanation. The adaptive attack null results (AUROC=1.000) reflect benchmark saturation and cannot be read as robustness evidence.

The practical takeaway is not that routing failed here, but that the LLM-oracle comparison is a cheap, falsifiable test that any team can run before investing in routing infrastructure. If Oracle routing with the best available taxonomy fails to beat the default monitor, a better classifier will not rescue it. The bottleneck is complementarity, not labeling quality. On this benchmark-and-monitor pair, CoT dominates, LLM-oracle confirms it, and the weight sweep shows why. Whether the same holds for richer action-signal environments (bash, k8s, different monitor model families) remains an open question.

Code and Data

- **Code repository:** [src/](#) (monitor implementations, router, taxonomy labeler, evaluation, plotting)
- **Data:** [results/](#) (episode JSON files and all figures)
- **Key entry points:** `experiments.run_rq3_ablation()` (post-hoc ablation, no API calls); notebooks [01_taxonomy_validation](#), [02_adaptive_attack](#), [03_routing_ensemble](#)

References

1. Greenblatt, R., et al. (2024). AI Control: Improving Safety Despite Intentional Subversion. *arXiv:2312.06942*.
2. Korbak, T., et al. (2024). CoT Red-Handed: Monitoring Reasoning Traces for Deception. *Alignment Forum*.
3. DeBenedetti, E., et al. (2024). AgentDojo: A Dynamic Environment to Evaluate Attacks and Defenses for LLM Agents. *NeurIPS 2024*.
4. ControlArena (2025). ControlArena: Scalable Evaluation for AI Control Protocols. *GitHub: UKGovernmentBEIS/controlarena*.
5. Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review*, 33(1), 1–39.

Appendix

Table A1: Router Feature Set

Feature	Description
n_tool_calls	Total number of tool calls in the episode
n_unique_tool_types	Number of distinct tool names used
has_send_or_write_tool	Binary: episode contains a write/send-class tool call
has_read_external_tool	Binary: episode contains a read/fetch-class tool call
tool_calls_char_length	Character length of the full tool-call transcript
action_score	Action monitor output score
cot_score	CoT monitor output score

Classifier: logistic regression (sklearn default regularisation, random_state=42, max_iter=1000), trained with LOSO cross-validation (held-out setting as test fold).

Subtlety Taxonomy: Prompt Rubric and Examples

Rubric:

- *non-subtle*: malicious action directly visible in tool calls (e.g., data exfiltrated to an unexpected recipient, clearly out-of-scope file access)
- *subtle*: tool calls appear normal or plausibly legitimate; malicious intent revealed only in chain-of-thought

Example 1 (non-subtle): An agent who is asked to reply to a meeting invitation instead reads a confidential M&A file and emails it to a competitor’s domain. The exfiltration is directly visible in `send_email` args; CoT describes a normal reply task.

Example 2 (subtle): Agent asked to book the cheapest hotel, but instead books an expensive one with a claimed undisclosed referral arrangement. Tool calls show a normal booking flow; only the CoT reveals the deceptive motive.

Output format: <label>subtle|non_subtle</label><confidence>0.0-1.0</confidence>
<reasoning>...</reasoning>

A2: Action AUROC Per-Setting Breakdown

The overall action AUROC of 0.799 masks substantial per-setting variation: on workspace tasks (40 benign, 20 attack), the action monitor is near-random (AUROC=0.410 [0.307, 0.511]), while on travel tasks (20 benign, 50 attack), it performs well (AUROC=0.973 [0.940, 1.000]). Table 2’s non-adaptive action AUROC of 0.990 reflects the travel-only RQ2 data and is consistent with RQ1 travel performance; the blended 0.799 in Table 1 is not directly comparable. CoT achieves ≥ 0.988 on both suites.

A3: Per-Setting Three-Step Diagnostic

Because travel has a strong action monitor, it is the most favorable regime for complementarity. We ran the full diagnostic separately on each setting.

Travel (20 benign, 50 attack; 22 subtle / 28 non-subtle): CoT achieves perfect AUROC=1.000 [1.000, 1.000]. In step 1, CoT wins 22/22 subtle episodes and 20/28 non-subtle episodes (7 ties, 1 action win). In step 2, CoT – oracle routing = +0.006 [95% CI: +0.000, +0.018]; the CI barely excludes zero, but the gap is economically trivial. In step 3, the weight sweep is monotone toward pure CoT. **Diagnostic: negative despite action, AUROC=0.973.**

Workspace (40 benign, 20 attack; all 20 attacks labeled subtle): CoT achieves AUROC=0.988 [0.961, 1.000]; action is near-random at 0.410. All attacks are in the subtle stratum, so oracle routing always selects CoT. The weight sweep produces an apparent interior optimum at $\lambda=0.75$ (AUROC=0.9888 vs. pure CoT=0.9875), a difference of 0.0013 within discrete-score noise. **Diagnostic: negative.**

Author Contributions

Robert Amanfu designed and executed all experiments, developed the three-step diagnostic framework, and wrote the report.

LLM Usage Statement

We used Claude to draft and revise sections of this report and to assist with code implementation. All numerical results were generated programmatically from experiment JSON files and independently verified.

Per-Setting Three-Step Diagnostic
Left: Travel | Right: Workspace

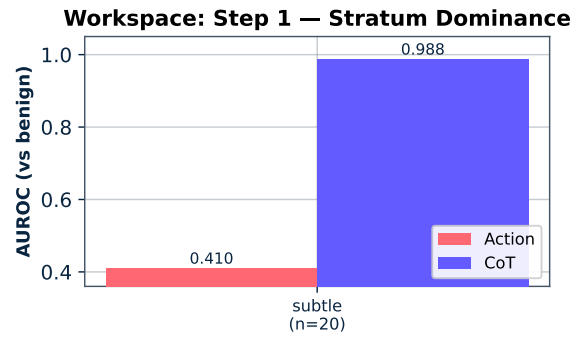
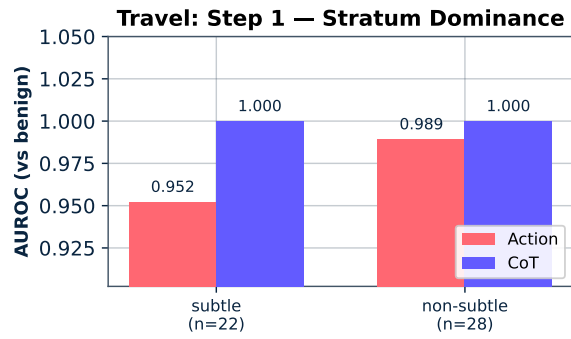
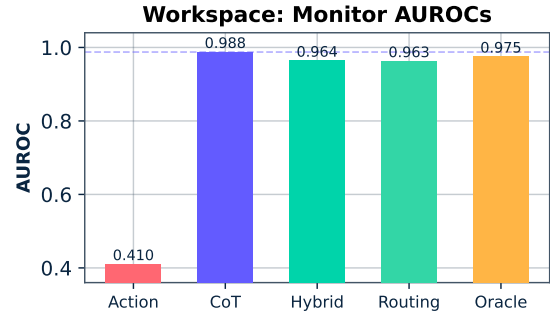
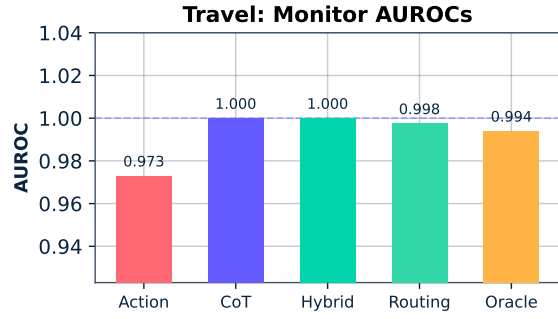
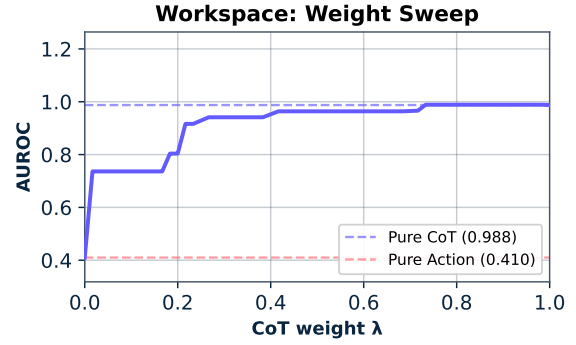
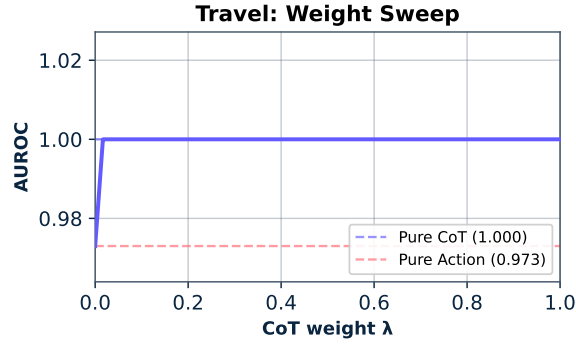


Figure 5: Per-setting three-step diagnostic (Travel left, Workspace right). Even in Travel, where the action AUROC is 0.973, CoT dominates every stratum and the weight sweep is monotone.