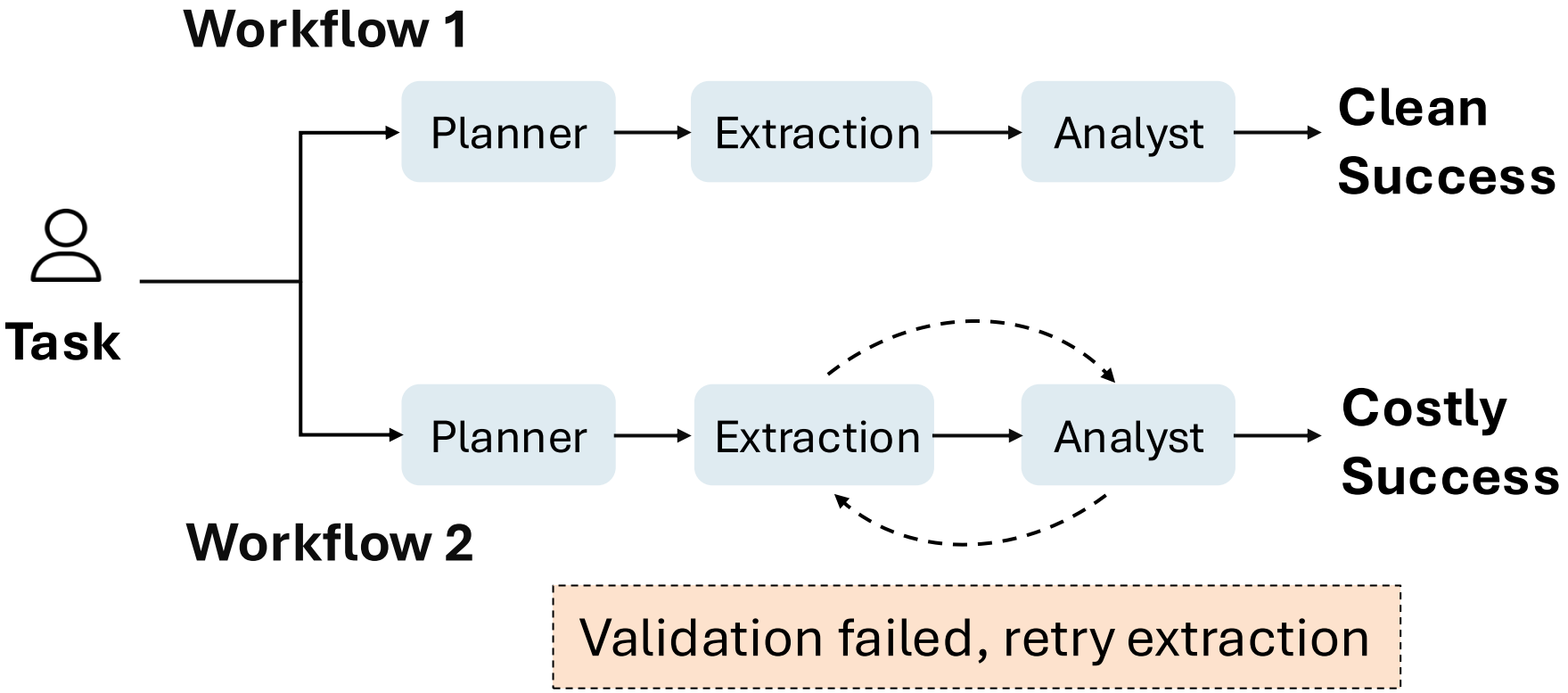
Trace-Level Analysis of Information Contamination in Multi-Agent Systems

Anna Mazhar, Huzaifa Suri, Sainyam Galhotra

Cornell University, University of Illinois Urbana-Champaign

Final Accuracy Overstates Reliability



Final correctness cannot distinguish clean success from unnecessary costly successful runs

Final Accuracy Overstates Reliability

- METR reviewed **296 AI-generated SWE-bench Verified** PRs
  - **~50% of passing PRs** would not be merged by human maintainers
  - Automated grading accepted **24.2 pp more PRs** than human maintainers

Passing the final check does not prove the workflow is reliable.

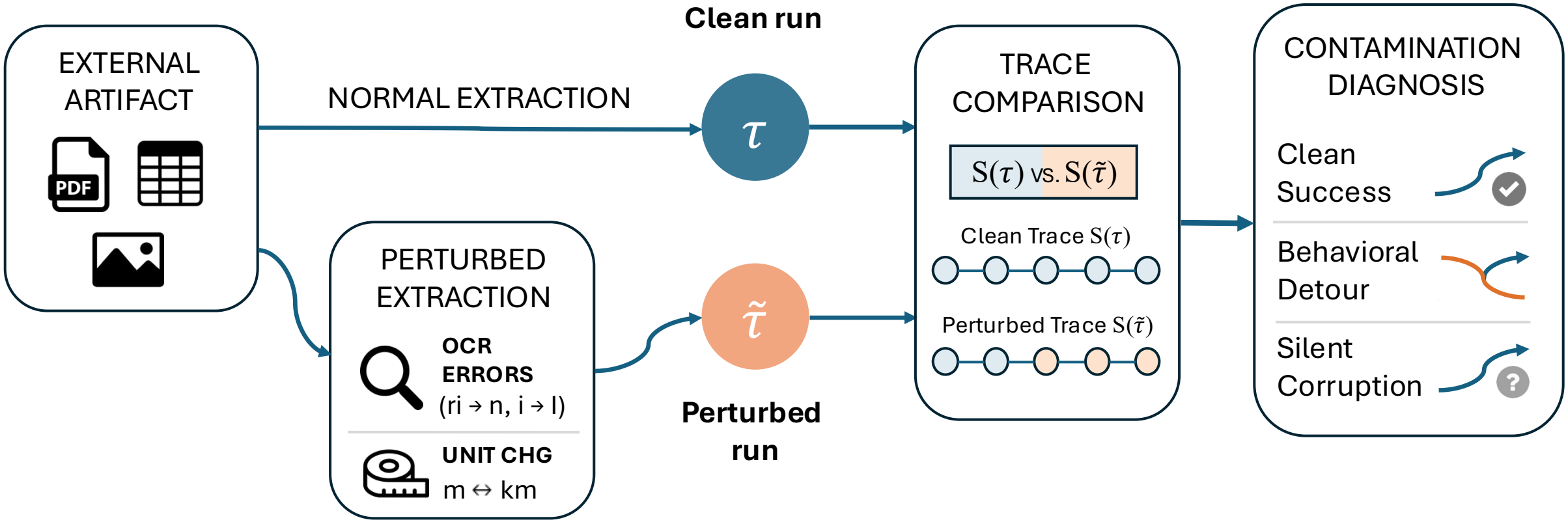
METR

Final Correctness Misses Trace-Level Failures

- Final accuracy cannot diagnose whether a workflow is efficient
- Source of unreliability: **noise or corruption external info.**
  - Agentic workflow must be robust to it.
- We need **trace-level analysis**: how information propagates, and how corrupted **information contaminates** downstream steps.

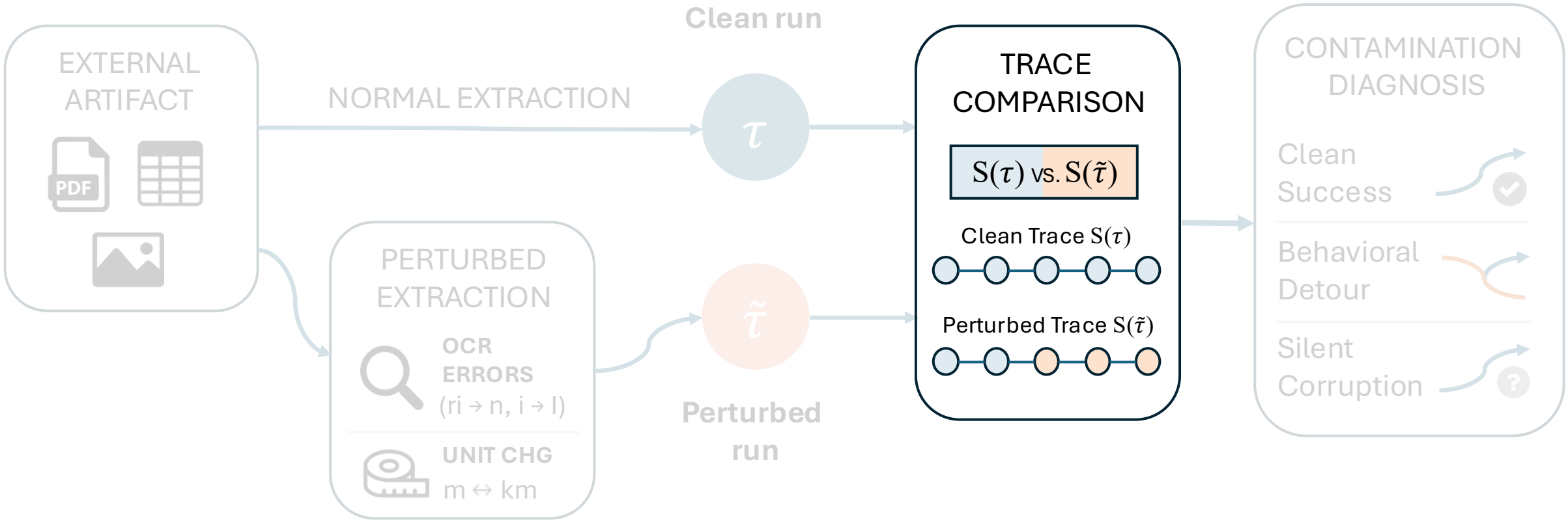
Problem Statement: How do noisy or corrupted artifact inputs contaminate multi-agent AI workflows, and how can we detect their impact beyond output accuracy?

Trace-Level Framework for Contamination Analysis



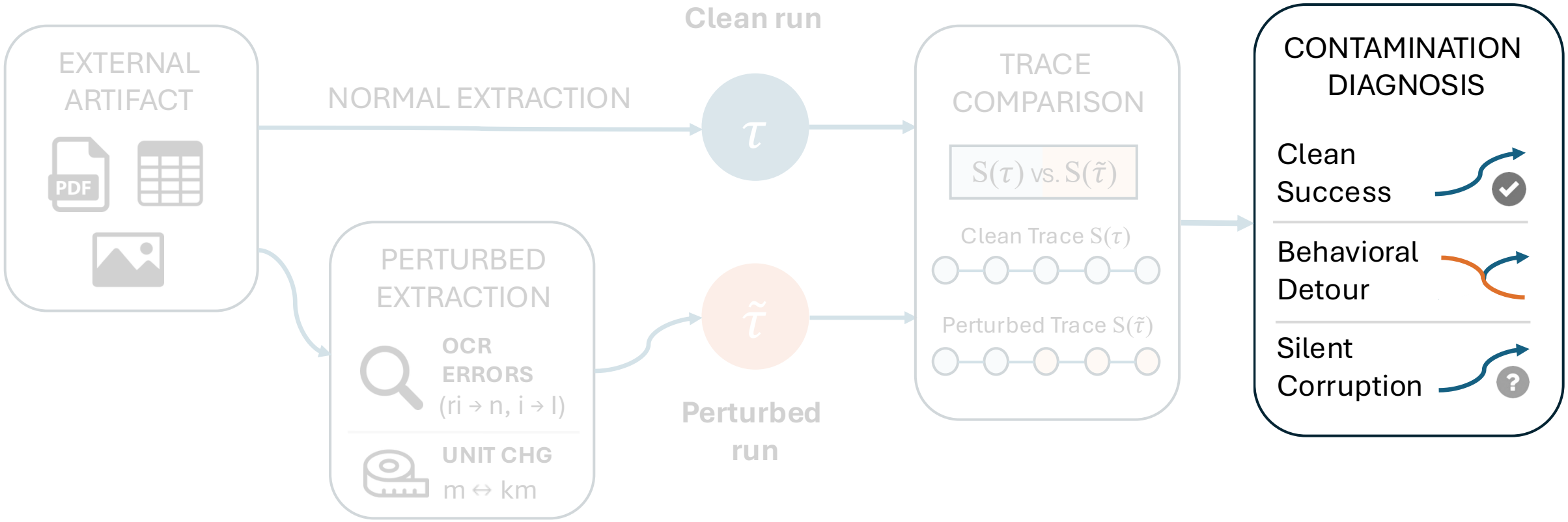
Controlled perturbations reveal how contamination propagates

Trace-Level Framework for Contamination Analysis



Controlled perturbations reveal how contamination propagates

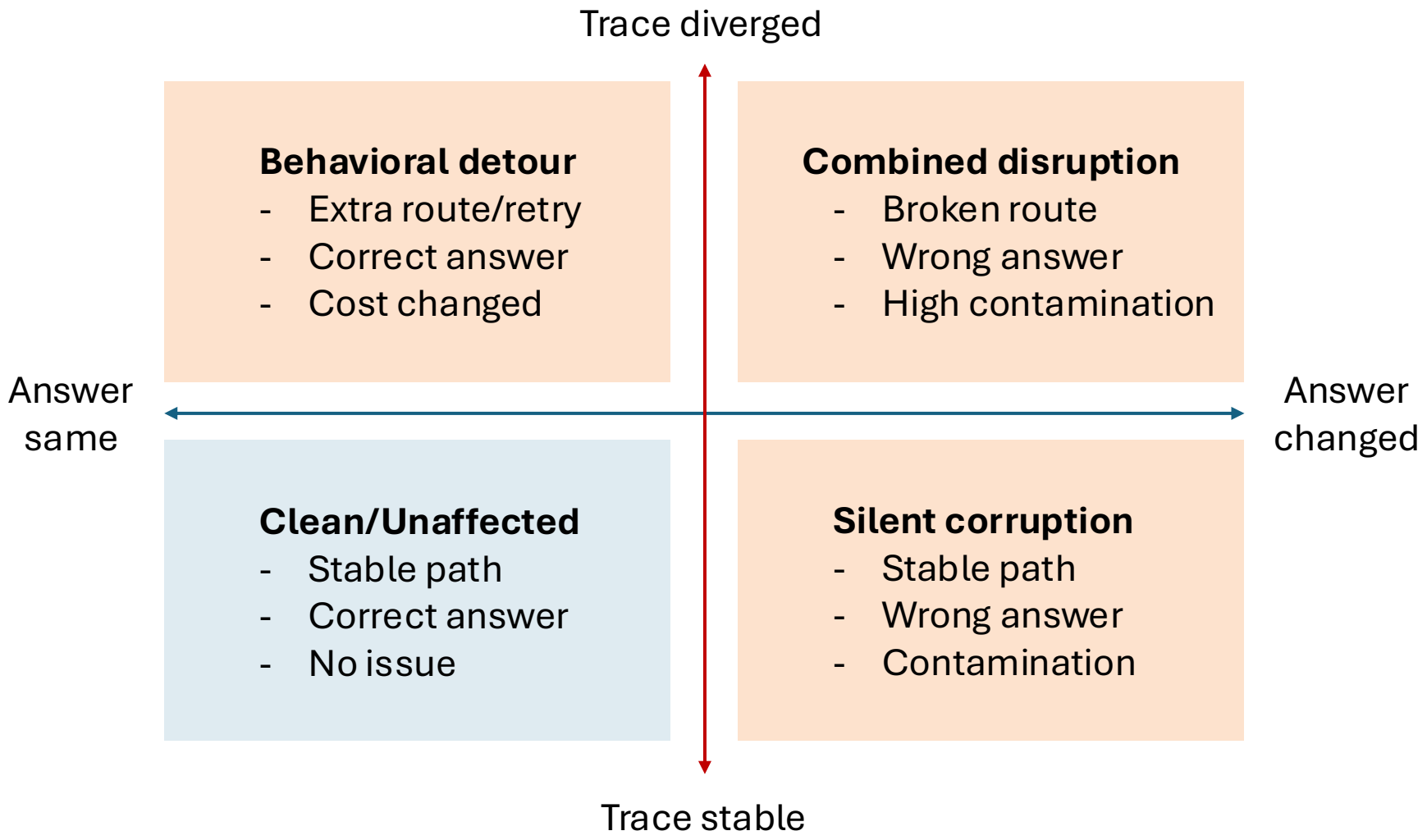
Trace-Level Framework for Contamination Analysis



Controlled perturbations reveal how contamination propagates

Combine trace divergence with final accuracy

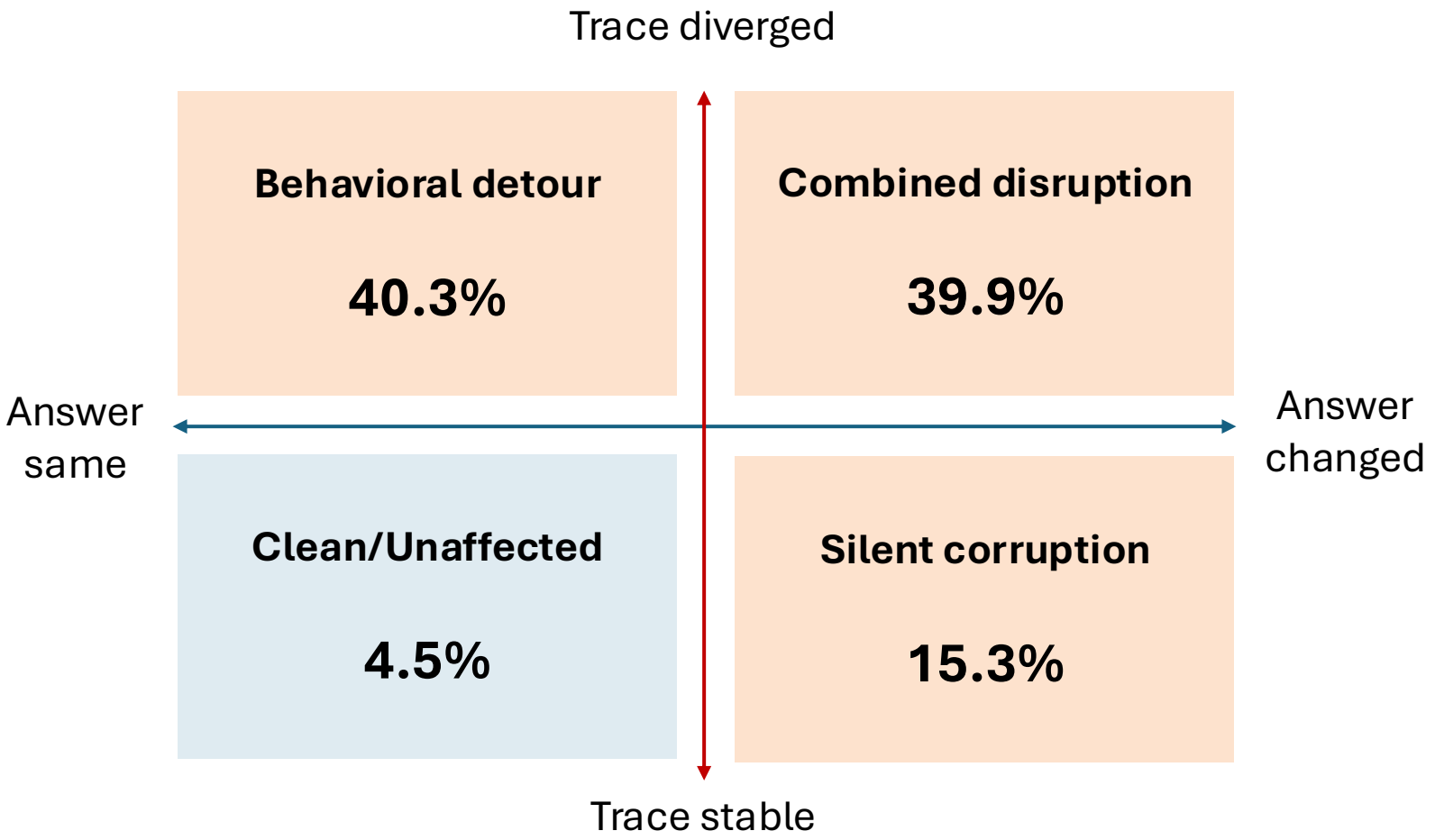
Combine trace divergence with final accuracy



Experimental Setup: Clean vs Perturbed

- **32 GAIA file-based tasks**
  - PDFs, spreadsheets, documents, images, audio
- **614 paired runs across 3 LLM backends**
  - GPT-5-mini, LLaMA-3.1-70B, Qwen3-235B
- **Controlled perturbations (~20)**
  - Unit change, OCR noise, blur, watermark …
- **Paired comparison**
  - Clean trace τ vs. perturbed trace $\tilde{\tau}$

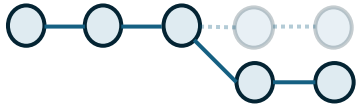
Finding: Prevalence of diagnosis types



Final accuracy evaluation misses both costly recoveries and silent failures.

Finding: Controlled Flow Signatures

REROUTING

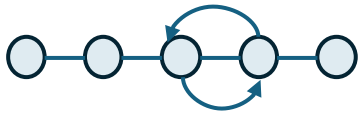


Different agent/tool selected

Target: Router and confidence calib.

80.6%

EXTENDED EXECUTION



Extra retry and failed validation

Target: Retry logic/ Stopping criteria

37.4%

EARLY TERMINATION

Skipped downstream agents

Target: fallback and recovery

25.3%

Trace divergence is actionable: it points to different repair targets.

Finding: Cost does not predict correctness

- **HIGH COST ≠ SUCCESSFUL**
  Only **16.3%** of high-cost runs were correct
  Retries and detours consume tokens, but do not guarantee recovery.

- **LOW COST ≠ TRUSTWORTHY**
  76.2% of low-cost runs were incorrect
  Silent corruption can follow the nominal path with near-baseline cost

More meat in the paper!!

Future Directions

- **From detection to attribution**
  Identify which artifact, extraction step, or agent boundary caused contamination.

- **Contamination-aware guardrails**
  Design guardrails that catch silent semantic corruption, not just tool errors, loops, or high-cost runs.

- **Policy-aware verification**
  Adapt validation and retry budgets based on system goals: cost, latency, reliability, or safety

Conclusion

- **We decouple accuracy and structure**
  Runs can recover after major divergence, or fail silently with normal-looking traces.
- **We introduce a trace-level framework**
  Paired clean/perturbed runs measure how contamination changes multi-agent execution.
- **We identify actionable failure signals**
  Contamination appears as behavioral detours, silent semantic corruption, combined disruption, and cost/control-flow signatures.