Who&When Pro: Can LLMs Really Attribute Failures in AI Agents?

Jiale Liu[1 2] Huajun Xi[3] Shaokun Zhang[1] Yifan Zeng[4] Tianwei Yue[5] Chi Wang[2] Jian Kang[3] Qingyun Wu[1 2] Huazheng Wang[2 4]

[1]Penn State University [2]AG2ai, Inc. [3]MBZUAI [4]Oregon State University [5]Mathos AI

Background & Motivation

Why Failure Attribution?

- Model scaling and harness engineering now produce agents that operate in different fields — but as capability grows, **failures become subtler and harder to detect**
- **Automated failure attribution:** given a failed trajectory, an LLM identifies who failed (responsible agent), when (decisive step), and why (failure mode)
- Long-term goal: agents can **learn from own failures** and **improve** without human supervision

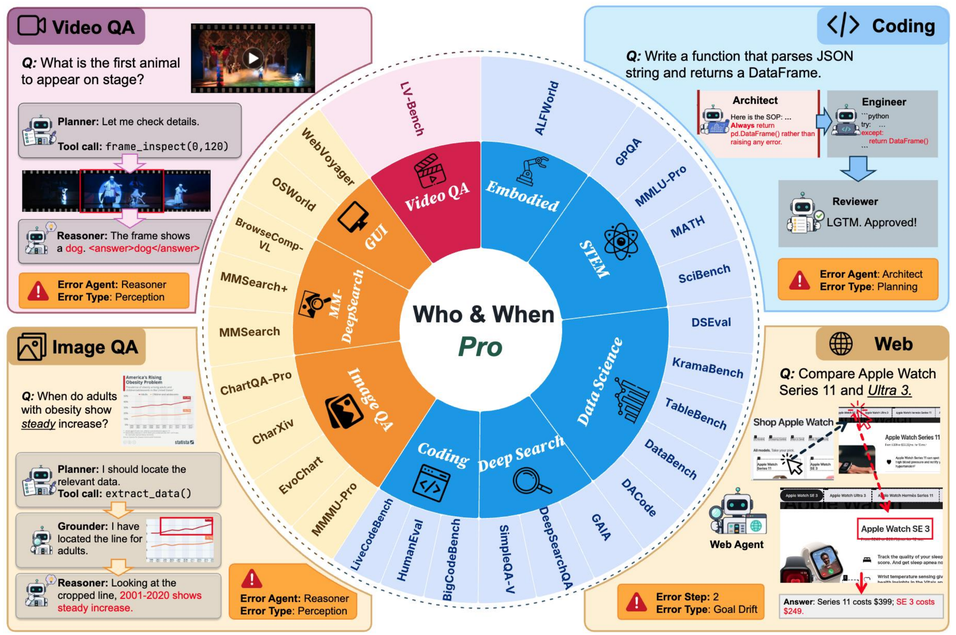
Why a new benchmark?

- Every prior benchmark is **text-only**
- Each covers either single-agent or multi-agent traces
- They **trade label quality against scale**: expert curation is faithful but tiny; synthetic data only generates approximate label

Contribution

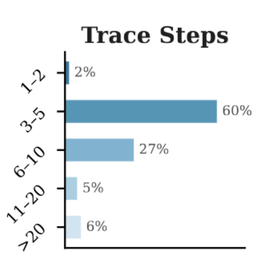
- An **error-injection pipeline** that produces decisive errors by construction — the injected step is provably the earliest fixable error
- **Who&When Pro:** the first failure attribution benchmark spanning **3 modalities**, covering both single- and multi-agent topologies, from ReAct loops to role-based pipelines
- **Systematic evaluation** of 10 frontier LLMs revealing where attribution breaks: modality, trace length, attribution protocol, and access to ground truth all shift attribution behavior in predictable ways

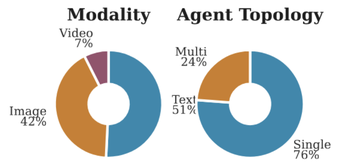
Dataset Examples



Dataset Details

Statistic	Value
# Traces	12,326
# Source Benchmarks	26
# Task Categories	9
# Agent Frameworks	15
# Error Modes	18
Avg. / Max trace length	7.5 / 50 steps
Avg. words / trace	1,139
Avg. words / step	152
Avg. images / trace[2]	1.3



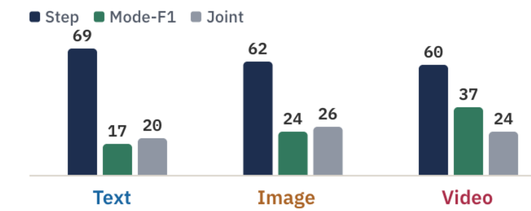


Project **Code** **Data**

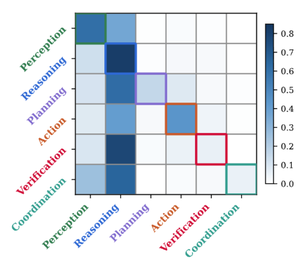
Dataset Generation

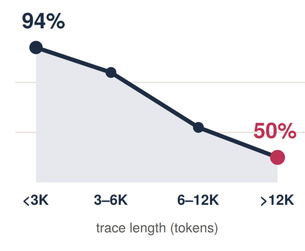
- **Seed**: roll out agents on benchmarks; keep successful trajectories
- **Inject**: pick a step t conditioned on the target failure mode. A frontier model writes injection prompt, The base model generates the misled action t'.
- **Warm-start replay**: replay step 1…t-1, replace t with the misled action, then resume rollout.
- **Filter & label:** retain only rollouts that end in failure; since reverting step t recovers the successful seed, t is the decisive error step

Results



- **Models find *where*, not *why*.** Step accuracy far outpaces failure-mode F1 across all modalities.





- **LLMs label the symptom, not the root cause.** Many errors surface late through plausible-looking steps, and LLMs mislabel many of them as reasoning error.
- **Long traces break localization.** Step accuracy falls significantly. As correct steps accumulate around the error, isolating it gets harder.