ICML
International Conference
On Machine Learning

Beyond the Final Answer

Evaluating the Reasoning Trajectories of Tool-Augmented Agents

Wonjoong Kim · Sangwu Park · Yeonjun In · Sein Kim · Dongha Lee · Chanyoung Park

KAIST · Yonsei University

ICML 2026

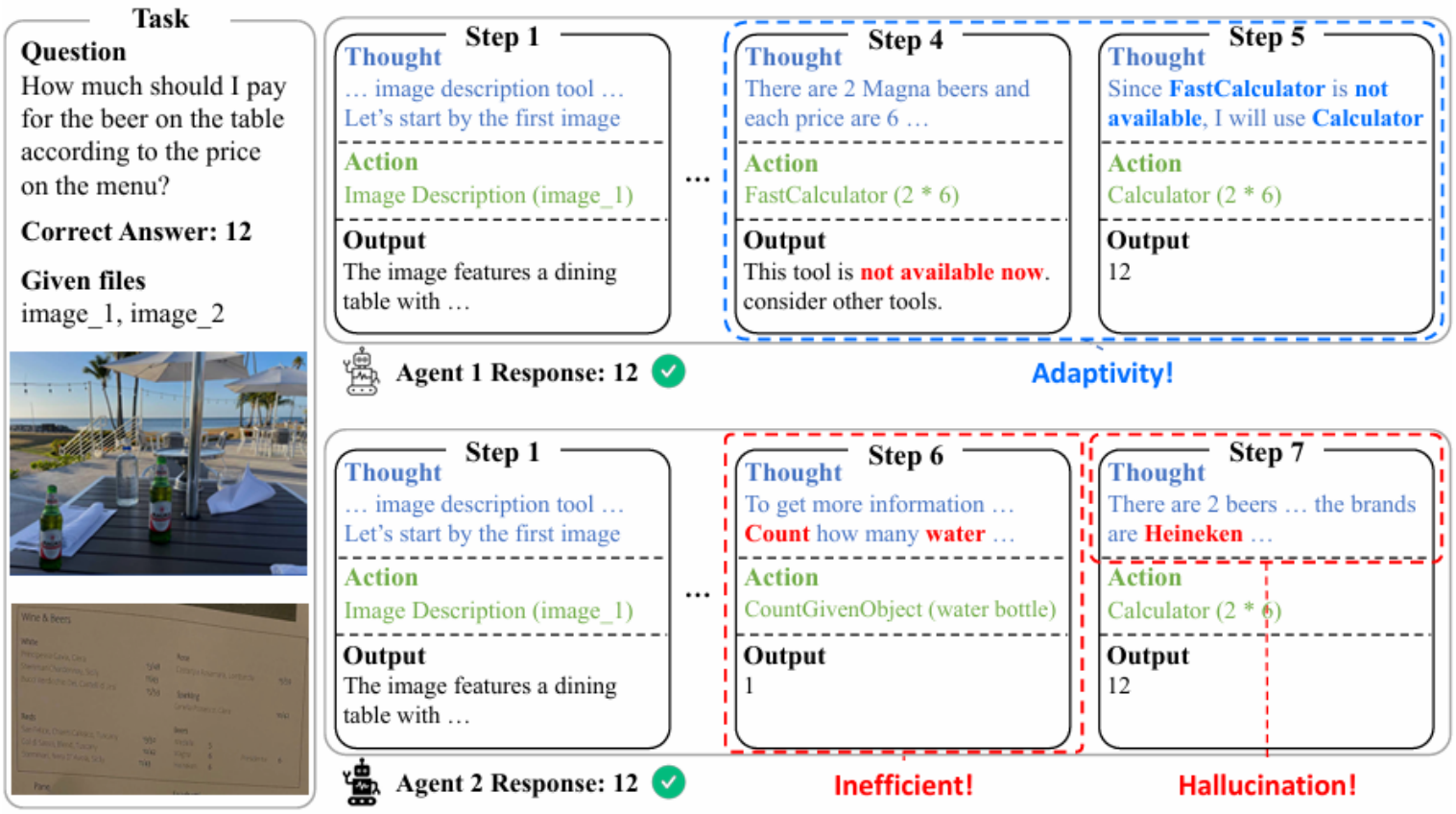
KAIST

DSAIL Data Science & Artificial Intelligence

연세대학교 YONSEI UNIVERSITY

Same Answer, Different Trajectories

Final-answer accuracy hides how the agent actually reasoned



Identical answer accuracy → Very different agents. Answer-match alone is not enough!

Why Existing Evaluation Falls Short

Current benchmarks score the destination, not the journey

Answer Match

Compares only final outputs. Ignores efficiency, hallucination, and adaptivity.

Single Ground-Truth

Many valid trajectories exist. Annotating them all is prohibitively expensive.

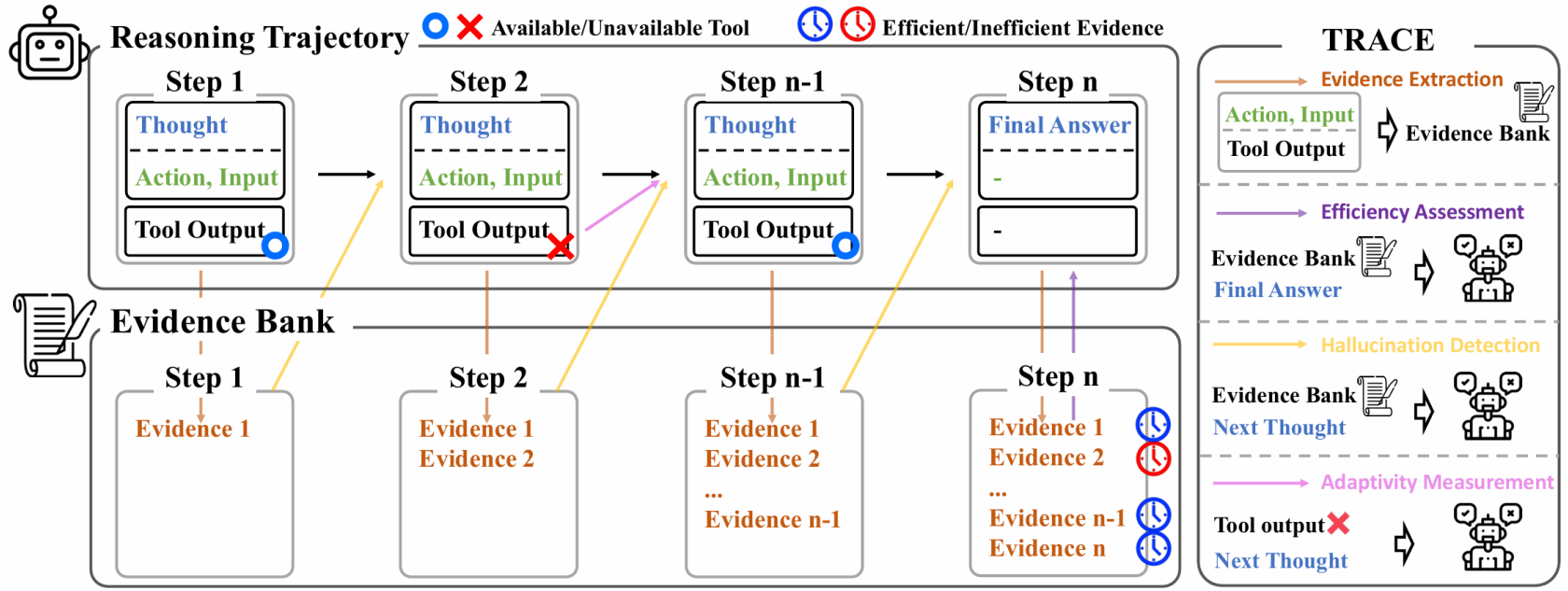
Naive LLM-as-Judge

Long, complex dialogs degrade evaluator quality — especially for small models.

What we need: a reference-free evaluator that scores ***how*** an agent reasons — efficiency, hallucination, and adaptivity — without ground-truth trajectories.

TRACE

Trajectory-based Reasoning Assessment and Comprehensive Evaluation



Why an Evidence Bank?

Structured (tool, input, output) tuples are easier for LLM judges to reason over than raw dialog.

Reference-free — no ground-truth trajectory needed.

Small-model friendly — even Llama-8B becomes a viable evaluator.

Three Trajectory Metrics

Each metric is computed against the Evidence Bank – no ground truth required

Efficiency

$$Eff(T) = |E_{min}| \,/\, |E_n|$$

How much of the gathered evidence was actually needed for the final answer? Find the minimal subset; the rest is waste.

Hallucination

$$H(s_t) = \neg IsGrounded(th_t, E_{t-1})$$

At each step, is the agent's thought logically grounded in the evidence collected so far?

Adaptivity

$$Adp(s_{t+1}) \text{ after tool failure}$$

When a tool returns an error, does the agent switch to a sensible alternative — or repeat the broken call?

Does TRACE Work?

Meta-evaluation on augmented benchmarks with labeled flaws

Models	Meta-GTA						Meta-m&m's	
	LLM-as-a-Judge			TRACE			LLM-as-a-Judge	TRACE
	Efficiency	Hallucination	Adaptivity	Efficiency	Hallucination	Adaptivity	Efficiency	Efficiency
Claude-Sonnet-4	86.08	89.68	98.83	94.64 (+8.56)	95.21 (+5.53)	99.63 (+0.8)	86.08	85.75 (-0.33)
GPT-4.1	81.45	94.42	97.95	94.24 (+12.79)	95.40 (+0.98)	97.03 (-0.92)	84.35	86.12 (+1.77)
o3-mini	90.24	94.68	96.59	94.09 (+3.85)	94.69 (+0.01)	96.91 (+0.32)	88.13	88.56 (+0.43)
Llama-3.3-70B	76.55	88.95	98.23	90.03 (+13.48)	95.97 (+7.02)	98.30 (+0.07)	86.47	87.19 (+0.72)
Llama-3.1-8B	55.67	89.59	78.98	70.46 (+14.79)	93.78 (+4.19)	85.28 (+6.3)	44.61	64.05 (+19.44)

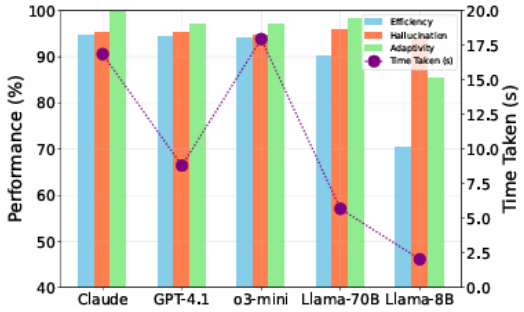
Consistent gains across all evaluators

Up to +14.8 pts in efficiency, +7.0 in hallucination on small open-source LLMs.

Models	TRACE	PIPA
Claude-Sonnet-4	98.68 ± 1.26	77.31 ± 9.96
GPT-4.1	96.91 ± 2.28	84.04 ± 7.17
o3-mini	95.48 ± 1.81	89.76 ± 4.74
Llama-3.3-70B	94.32 ± 4.40	83.28 ± 8.57
Llama-3.1-8B	79.10 ± 8.73	24.94 ± 12.91

Robust to multiple valid trajectories

TRACE: 98.7 ± 1.3 vs PIPA: 77.3 ± 10.0
(Claude on Meta-GTA, multi-trajectory subset).



3× faster with Llama-70B

Matches proprietary accuracy at a fraction of evaluation cost.

Insights from Evaluating Agents

Same overall accuracy does not mean same capability

Model	Claude-Sonnet-4					GPT-4.1					o3-mini				
Evaluator	Claude	GPT	Llama	o3-mini	**Avg.**	Claude	GPT	Llama	o3-mini	**Avg.**	Claude	GPT	Llama	o3-mini	**Avg.**
Efficiency	0.9427	0.9495	0.9459	0.9025	**0.9351**	0.9659	0.9328	0.9380	0.8958	**0.9331**	0.9888	0.9457	0.9269	0.8914	**0.9382**
Hallucination	0.9511	0.9815	0.9878	0.9718	**0.9730**	0.9513	0.9953	0.9884	0.9787	**0.9784**	0.9818	0.9961	0.9961	1.0000	**0.9935**
Adaptivity	0.9091	0.9545	0.8636	0.7727	**0.8750**	0.7667	0.8111	0.8333	0.8111	**0.8056**	0.5909	0.6818	0.6818	0.5455	**0.6250**
Inst. ↓	0.0029 / 0.0038					0.0074 / 0.0093					0.0083 / 0.0400				
Answer Accuracy	0.5321 / 0.6607 / 0.6754					0.4487 / 0.7208 / 0.6914					0.4808 / 0.7320 / 0.5933				
Overall Accuracy	0.5767					0.5281					0.5263				
Model	**Llama-3.3-70B**					**Mixtral-8x7B**					**Qwen-72B**				
Evaluator	Claude	GPT	Llama	o3-mini	**Avg.**	Claude	GPT	Llama	o3-mini	**Avg.**	Claude	GPT	Llama	o3-mini	**Avg.**
Efficiency	0.8784	0.7456	0.7819	0.7064	**0.7781**	0.7513	0.7513	0.7811	0.6858	**0.7424**	0.9687	0.9245	0.9287	0.9525	**0.9436**
Hallucination	0.8090	0.9410	0.9768	0.9214	**0.9121**	0.8616	0.9505	0.9831	0.9381	**0.9333**	0.9324	0.9820	0.9836	0.9052	**0.9508**
Adaptivity	0.8547	0.9012	0.9012	0.8721	**0.8823**	0.5000	0.6250	0.5000	0.6250	**0.5625**	0.7969	0.7969	0.7969	0.7969	**0.7969**
Inst. ↓	0.0738 / 0.0047					0.091 / 0.061					0.017 / 0.0021				
Answer Accuracy	0.3205 / 0.3752 / 0.5191					0.0109 / 0.6223 / 0.1822					0.4359 / 0.7679 / 0.6815				
Overall Accuracy	0.3738					0.1631					0.5202				
Model	**Llama-3.1-8B**					**Mistral-7B**					**Qwen-7B**				
Evaluator	Claude	GPT	Llama	o3-mini	**Avg.**	Claude	GPT	Llama	o3-mini	**Avg.**	Claude	GPT	Llama	o3-mini	**Avg.**
Efficiency	0.5244	0.7423	0.5184	0.4633	**0.5621**	-	-	-	-	**-**	0.8893	0.9445	0.9118	0.8650	**0.9026**
Hallucination	0.6823	0.8355	0.9519	0.8069	**0.8192**	0.9959	1.0000	1.0000	1.0000	**0.9990**	0.8282	0.9435	0.9566	0.9193	**0.9119**
Adaptivity	0.5556	0.5556	0.5556	0.5556	**0.5556**	0.0000	0.0000	0.0000	0.0000	**0.0000**	0.8495	0.8656	0.8656	0.8979	**0.8696**
Inst. ↓	0.2397 / 0.0027					0.0480 / 0.061					0.1254 / 0.008				
Answer Accuracy	0.0321 / 0.2699 / 0.6193					0.0000 / 0.1890 / 0.0088					0.2628 / 0.5944 / 0.6823				
Overall Accuracy	0.1948					0.0154					0.3904				

Qwen-72B vs GPT-4.1

Comparable accuracy — but Qwen hallucinates more, GPT-4.1 adapts less.

o3-mini

Near-zero hallucination, yet struggles to recover after tool failures (low adaptivity).

Smaller models

More output tokens ↔ lower accuracy. Sometimes constraining 'thinking' helps.

Trajectories matter. TRACE makes them measurable — without ground truth, at any model scale.

Thank you!

[Full Paper] https://arxiv.org/abs/2510.02837v2

[Source Code] https://github.com/wonjoong-kim/TRACE

[Email] wjkim@kaist.ac.kr

[LinkedIn] https://linkedin.com/in/wonjoong-kim-7547aa1ba

[Lab Homepage] https://dsail.kaist.ac.kr

KAIST DSAIL Data Science & Artificial Intelligence 연세대학교 YONSEI UNIVERSITY