CAIA: A BENCHMARK FOR TOOL-GROUNDED LANGUAGE AGENTS IN REAL-LIFE ADVERSARIAL CRYPTOCURRENCY TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

An agent answering a cryptocurrency question must find the relevant historical records and compute the requested quantity. Counting transactions, for example, does not answer a question about distinct wallets. We introduce CAIA, 178 expert-curated tasks with reference answers and checked tool sequences for obtaining them. We compare 17 models through a shared 23-tool interface, allowing up to five tool-use rounds within each run. Tools raise GPT-5's single-attempt success from 28.1% to 70.2%. Repeating the task in five independent runs produces a correct GPT-5 answer on 77.0% of tasks, but majority voting selects a correct answer on only 67.4%. This exposes an answer-selection bottleneck: correct candidates are available but lost during aggregation. Across 23,463 calls, an individual attempt's call count also has near-zero association with correctness. The interaction records connect these outcomes to source choice, historical scope, and calculation. We release the questions, expert reference routes, and records to support evidence-based analytical agents.

1 INTRODUCTION

Tool-using language agents can answer analytical questions by retrieving records and calculating from them. Each step must preserve what the user asked. For example, counting distinct wallet addresses over a quarter requires deduplicating the participating addresses; counting transactions gives a different quantity. Even a credible summary may cover the wrong interval. The task runs from understanding the question to finding sources, selecting records, and performing the calculation.

Cryptocurrency analysis brings these demands together in an inspectable but contested information environment. Public ledgers record transactions, while documentation, market data, and social discussion differ in authority and temporal scope. Market manipulation and transaction-ordering incentives further complicate source interpretation (Xu & Livshits, 2019; Ardia & Bluteau, 2023; Daian et al., 2019). Financial question answering studies numerical reasoning over supplied evidence (Chen et al., 2021b; Zhu et al., 2021; Chen et al., 2022). An analytical agent must additionally find that evidence and establish that it concerns the requested entity and historical state.

CAIA turns these requirements into 178 tasks drawn from community requests and refined by domain experts. Each has a reference answer and a validated toolchain: a sequence of calls that obtains the information needed to answer it. Block identifiers or timestamps specify the historical state where relevant. Agents can use other routes; we score their final answers and retain their interaction records, or trajectories, for diagnosis. Six categories cover on-chain analysis, project discovery, tokenomics, cross-domain questions, trend analysis, and general knowledge, from direct retrieval to multi-source synthesis.

We compare 17 models answering directly or using a shared toolkit in a standardized ReAct-style loop (Yao et al., 2023). The toolkit provides 23 interfaces. Within a run, the agent has up to five tool-selection iterations, each allowing complementary calls. Across runs, we sample five independent attempts at the task. The first budget controls evidence gathering; the second tests whether repeated attempts produce correct answers and whether voting selects them.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

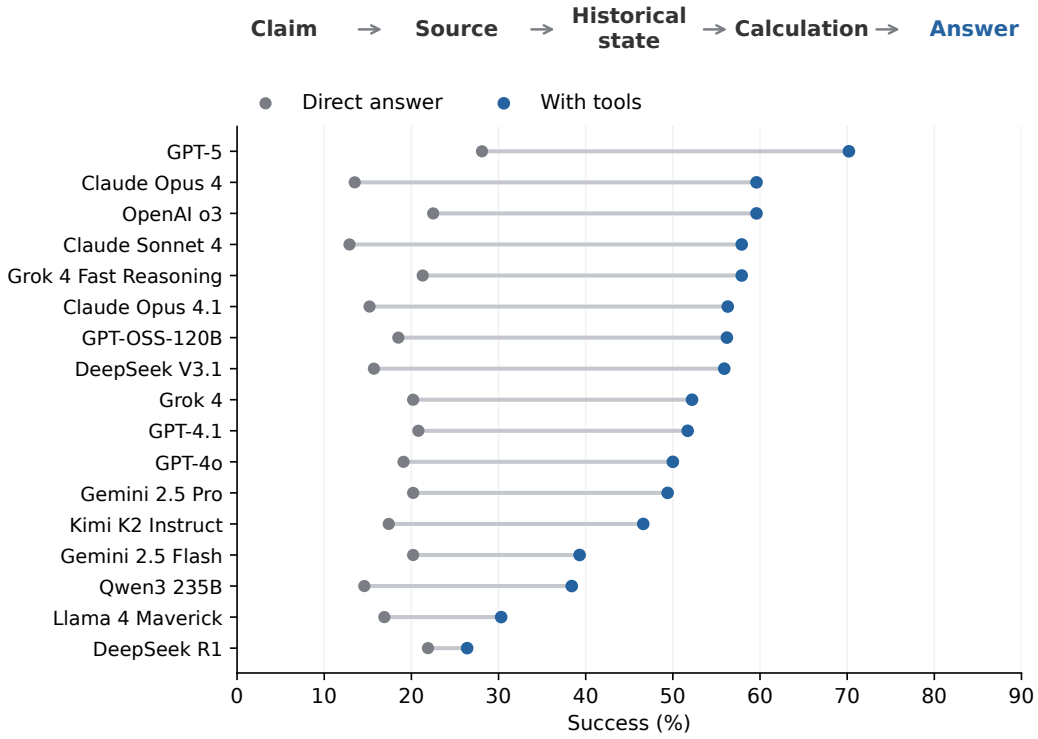


Figure 1: **Evidence construction connects tool access to an answer.** Top: the reference route identifies the source, historical state, and calculation needed for the claim. Bottom: single-attempt answer success for all 17 models; each line connects direct answering to the shared tool-enabled loop. Trajectories diagnose how agents seek evidence; final-answer correctness is scored.

The experiments expose an answer-selection bottleneck. Tools raise GPT-5's single-attempt success from 28.1% to 70.2%, yet roughly three in ten answers remain incorrect. Repeated attempts supply additional correct candidates, but majority voting leaves some unselected. More calls within an attempt offer no simple indicator of correctness either: the task-level association is near zero, even though more active models tend to score higher. The central challenge is to turn retrieved evidence and candidate answers into a correct final response.

CAIA contributes a time-grounded task bank, a common tool-use protocol, and an evaluation that connects answer correctness to interaction and attempt budgets.

2 Benchmark Design

2.1 From a Request to a Checkable Answer

A CAIA task specifies a question, its relevant entities and temporal scope, a reference answer, and an associated route through the available tools. Some questions concern a single transaction or protocol property. Others require aggregating observations over a time window or combining documentation with numerical evidence. In each case, correctness is tied to the requested claim rather than to the fluency of the explanation.

Consider the request to count unique wallets participating in Bored Ape Yacht Club ERC-721 transfers during calendar Q4 2024. Its reference route identifies the collection, retrieves transfer events within the interval, and counts distinct participating wallets, giving 1,966. Counting transfer events instead produces a different quantity even if every retrieved record is relevant. A secondary sum-

Table 1: **What a task asks an agent to establish.** Examples paraphrase task-bank questions. The evidence column identifies the information needed to answer each question, illustrating why a tool's relevance depends on the claim.

Question	Evidence needed
What is the timestamp of Ethereum block 19,560,000?	The specified chain's block record, with an explicit timestamp convention.
What was the USD swap volume of the Ethereum Uniswap V3 ETH/USDC 0.05% pool on January 2, 2025 (UTC)?	The correct pool and fee tier, observations within the UTC interval, and a consistent USD-volume measure.
How many MYSHELL tokens are scheduled to unlock from June 30 through September 30, 2025?	The official vesting schedule, the requested interval, and aggregation of the applicable releases.

Table 2: **The 178-task bank.** Categories describe the analytical question rather than a required tool. Cross-domain tasks combine several kinds of evidence.

Category	Tasks	Analytical focus
On-chain analysis	77	Transaction patterns, flows, and execution records
Project discovery	49	Protocol properties and documentation-based investigation
Tokenomics	23	Incentives, supply, and value-accrual mechanisms
Cross-domain	14	Synthesis across several analytical categories
Trend analysis	8	Time-dependent patterns and aggregation
General knowledge	7	Foundational concepts and reference facts

mary can fail differently: it may concern the right collection but cover the wrong quarter. This task requires the question's unit and interval to survive every step of the calculation.

Other tasks need only a precise retrieval. The timestamp of Ethereum block 19,560,000 is fixed by that block's record. Resolving the chain and block, then rendering the timestamp in the requested convention, suffices. The qualitative examples include failures that substitute a search snippet or an estimate for this direct lookup. Together, these cases show why a longer tool trajectory is not necessarily a better one.

Cross-domain questions require compatibility between individually plausible observations. To calculate Base's share of layer-2 total value locked, an agent needs a numerator and denominator with the same date and metric definition. Retrieving both values from credible sources is insufficient if one covers a different period or set of networks. Likewise, a token-unlock total requires the releases that fall inside the requested interval, not a headline supply figure. These examples extend the wallet-counting requirement to multi-source arithmetic: the calculation is only meaningful after its inputs refer to the same analytical question (Appendix G).

2.2 Curation and Reference Evidence

The initial pool contains more than 10,000 queries from over 3,000 contributors, including protocol developers, researchers, and market participants. Automated filtering removes off-topic, ambiguous, trivial, and unanswerable requests, then ranks the surviving candidates for expert review. The expert stage involves 92 specialists assigned 50 reviews each. Candidate questions receive at least four reviews; the ranking uses the mean after removing the highest and lowest scores.

The 200 highest-ranked questions are deduplicated to 186 candidates. Format standardization makes entity references and temporal constraints explicit. Reference-toolchain validation then checks that the intended answer can be obtained through the benchmark's information sources, yielding the final 178 tasks (Figure 2).

Reference routes connect a curated question to the records and calculation supporting its answer. Retained trajectories show where an agent's retrieval, interpretation, or computation diverges from that evidence.

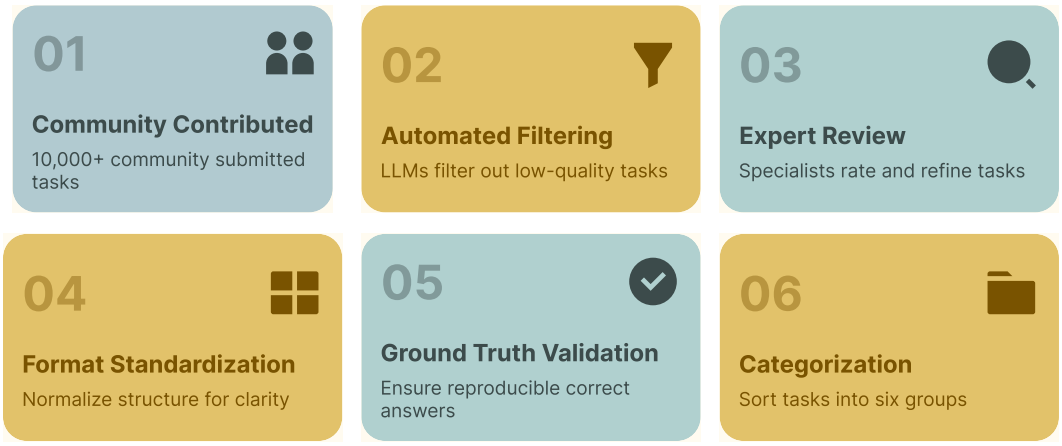


Figure 2: **Curation connects community questions to reference evidence.** Read left to right across each row. Automated filtering and expert review select analytical requests; standardization fixes their scope; reference validation checks the answer route; and categorization describes the resulting task bank.

2.3 Temporal and Adversarial Context

Historical anchors give time-sensitive questions a stable meaning. A specified block height or timestamp identifies the state being queried, even as search rankings, documentation, and service interfaces evolve. Recording the reference answer, temporal anchor, and returned tool outputs connects an evaluation score to the evidence available during that run. This separates answering a historical question from reporting whichever value a live source currently displays.

The environment is *adversarial* in the practical sense that available sources can serve competing incentives. Promotional claims, conflicting descriptions, and stale summaries coexist with ledger records and official documentation. The agent must decide what each source can establish about the requested fact. CAIA evaluates the resulting answer against expert references, while tool traces make source selection and historical alignment available for diagnosis.

Source authority is specific to the question. A social post can establish that a claim was made, but it cannot by itself establish a historical account balance. A project's documentation can identify a deployment, yet a similarly named deployment on another chain does not answer the same address query. Apparent contradictions can also disappear when dates or metric definitions are aligned. The benchmark's temporal and entity constraints make these choices inspectable: an evaluator can compare the requested object with the arguments sent to a tool and the scope of the returned record, rather than judging credibility from the source name alone.

3 Evaluation Protocol

Models and conditions. The model panel in Table 4 spans proprietary and open-weight systems. Every model answers the same task bank in direct-answer and tool-enabled conditions. Direct answering probes the model's available knowledge and reasoning without retrieval. Tool-enabled answering adds web search, URL retrieval, blockchain analytics, market and protocol data, social search, and computation through a common 23-tool interface.

A bounded agent loop. At each of at most five iterations, the model receives the question, tool descriptions, and accumulated tool results. It chooses further calls or answer synthesis. Multiple tools can be called in one iteration when the model requests complementary evidence. The iteration budget resets for each independently sampled run. The loop ends on synthesis or budget exhaustion, after which the model produces a structured answer. The same control structure, tool descriptions,

Table 3: **The toolkit exposes complementary evidence interfaces.** Representative names follow the shared tool catalogue. Correct routing depends on the requested entity, quantity, and historical state. The complete catalogue contains 23 tools.

Interface	Representative tools	Evidence role
Discovery	Web Search; URL Fetching	Locate a source and inspect the underlying document.
Ledger records	Block Query; Transaction Query; Data Query	Retrieve the specified block, transaction, or aggregate.
Historical state	Time Conversion; Balance Query; Supply Query	Resolve a temporal reference and the relevant state.
Protocol metrics	Protocol Metrics; Revenue Analysis; Network Activity	Establish an application- or network-level measure.
Market and supply	Price Feed; Release Schedule	Recover a price series or the applicable token releases.
Social evidence	Twitter Search; Social Stream	Inspect discussion and claims about the requested subject.
Computation	Code Execution	Aggregate retrieved values and check arithmetic.

output schema, and scoring criteria are used across the model panel. This fixes the scaffold while allowing the models to choose different actions within it.

Answer evaluation. The LLM evaluator checks the requested facts or calculation against expert-verified references, accepting semantically equivalent answers in both tool conditions. On 500 question–model pairs sampled uniformly without replacement, a ten-member domain-expert team's binary correctness judgments matched the evaluator in all 500 cases (Appendix B). Selected traces provide qualitative diagnoses of source selection, temporal alignment, and computation; overall model success rates use the full task bank.

Answer scores and tool records serve different parts of the analysis. The score records whether the final response answers the question. The record shows the sources consulted, query arguments, returned evidence, and calculation leading to that response. Reading them together distinguishes a retrieval error from an incorrect aggregation of relevant records. It also reveals successful shortcuts, such as resolving a historical timestamp with a direct block lookup.

Table 3 lists the available source types and computational interfaces; the agent chooses which to query and how to constrain the request.

Attempt budgets. Each task has five independently sampled runs per model and condition. Pass@1 measures single-attempt success. Pass@5 measures whether at least one of the five answers is correct; we call this *candidate coverage*. Majority-vote accuracy measures whether the answer selected from that set is correct. All use the same task references. Pass@5 tests whether sampling supplies a correct answer; voting tests whether aggregation recovers it.

Costs and trajectories. We log token usage and convert it to query cost at evaluation-time prices. The cost column reports these model-token charges. Tool counts provide a second resource measure: distinct interfaces describe the breadth of source use, while total calls include repeated access to the same interface. The action history connects both measures to the sequence of queries, retrieved evidence, and final synthesis within the five-iteration budget.

4 Results

4.1 Access to Tools Produces Large, Uneven Gains

GPT-5 leads single-attempt tool-enabled success at 70.2%, followed by Claude Opus 4 and OpenAI o3 at 59.6% (Table 4). Figure 1 places each result beside the same model's direct-answer score, showing substantial but uneven gains from the shared toolkit.

Table 4: **Answer correctness and attempt budget.** Pass@1 measures single-attempt success; Pass@5 measures the presence of a correct answer within five runs. Cost is the mean model-token charge per tool-enabled query at evaluation-time prices. Models are ordered by tool-enabled Pass@1 and identified by their evaluation labels.

Model	Direct P@1	Tools P@1	Tools P@5	Cost ($)
GPT-5	28.1	70.2	77.0	0.1154
Claude Opus 4	13.5	59.6	71.9	1.1139
OpenAI o3	22.5	59.6	73.6	0.1047
Claude Sonnet 4	12.9	57.9	66.9	0.2291
Grok 4 Fast Reasoning	21.3	57.9	71.9	0.0098
Claude Opus 4.1	15.2	56.3	69.0	0.9357
GPT-OSS-120B	18.5	56.2	72.5	0.0066
DeepSeek V3.1	15.7	55.9	71.2	0.0216
Grok 4	20.2	52.2	66.9	0.1980
GPT-4.1	20.8	51.7	60.7	0.0913
GPT-4o	19.1	50.0	55.6	0.0909
Gemini 2.5 Pro	20.2	49.4	61.2	0.0407
Kimi K2 Instruct	17.4	46.6	64.6	0.0273
Gemini 2.5 Flash	20.2	39.3	62.4	0.0070
Qwen3 235B	14.6	38.4	61.0	0.0062
Llama 4 Maverick	16.9	30.3	64.6	0.0031
DeepSeek R1	21.9	26.4	54.5	0.0121

A separate human task-solving pilot provides context on a stratified 10% subset of the bank (Appendix B). Sixteen participants from university blockchain clubs and early-stage blockchain companies answered with tool access, achieving a reported mean of 80% after the highest- and lowest-scoring participants were removed. The model table uses the full 178-task bank, so the pilot supplies a subset-based human reference. It complements the expert toolchains: the pilot records participants' success on sampled requests, while the validated routes document how the intended answers can be obtained throughout the task bank.

Roughly three in ten answers remain incorrect for the strongest tool-enabled model. Can more activity close this gap? We examine additional independent attempts first, then the number and type of calls within an attempt.

4.2 Repeated Attempts Find Answers That Voting Misses

GPT-5 improves from 70.2% single-attempt success to 77.0% success within five attempts. DeepSeek R1 improves from 26.4% to 54.5%, and Llama 4 Maverick from 30.3% to 64.6%. Repeated sampling more than doubles the chance of including a correct answer for the latter two models.

Those correct candidates are not always selected. With the same five-run budget, GPT-5 reaches 77.0% candidate coverage but only 67.4% majority-vote accuracy, below its 70.2% single-attempt score (Figure 4). Agreement therefore fails to exploit the available correct answers. This motivates checking candidates against their supporting records rather than using frequency alone.

Consider a hypothetical five-answer set: three attempts count transfers and two correctly count wallets. It succeeds under Pass@5, but voting can select the wrong quantity. All five attempts could use records from the right collection and interval; the error lies in what they count. Selecting the correct answer requires checking the requested unit against each calculation, even when the wrong interpretation is more frequent.

4.3 Call Count Has Little Association with Individual Success

Web and social search together account for 55.5% of calls, compared with 35.6% for specialized blockchain tools (Table 5). Search dominates the interaction budget, whether it leads to an authoritative document or another summary that leaves the requested fact unresolved.

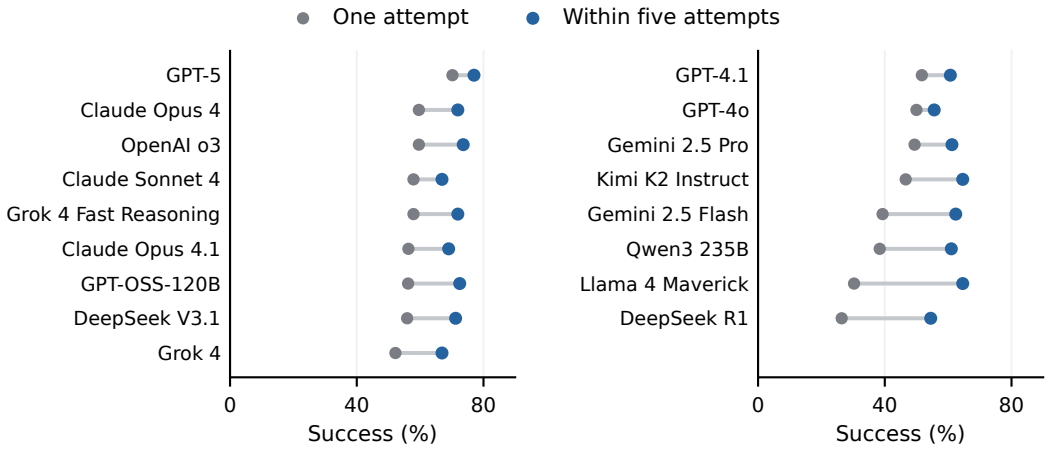


Figure 3: **Additional attempts expand the set of solvable questions.** Pass@1 and Pass@5 in the tool-enabled condition. The gap measures how often repeated sampling supplies a correct answer beyond single-attempt success. Figure 4 evaluates the complementary step: selecting an answer through majority vote.

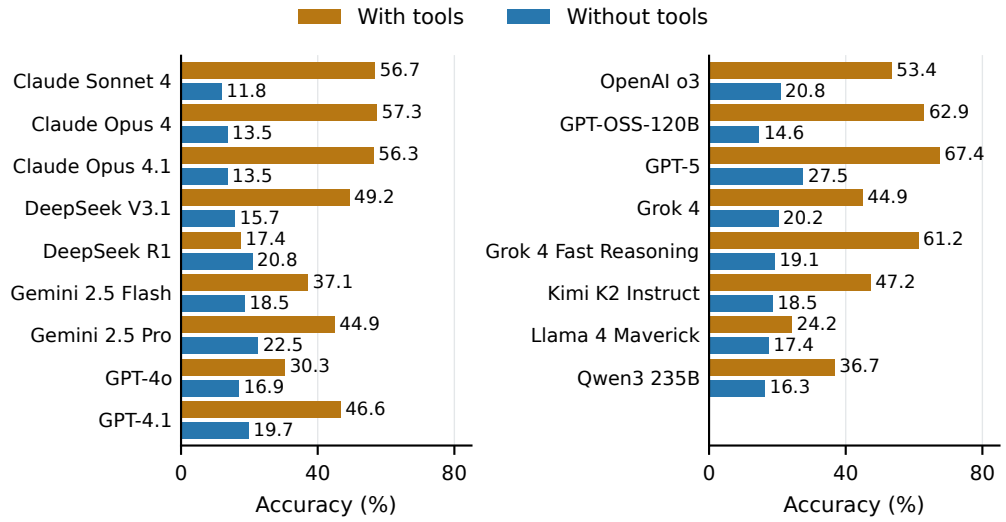


Figure 4: **Answer aggregation gives a third view of reliability.** Five-run majority-vote accuracy scores the aggregated answer, whereas Pass@5 scores whether the candidate set contains a correct answer. The bars therefore use a different aggregation rule from Figure 1 and Table 4.

The model-level Pearson correlation between call volume and accuracy is 0.82; the item-level correlation is approximately 0.03. More active models tend to score higher, yet the length of an individual trajectory has little linear association with its correctness.

The two tasks in Section 2 illustrate why useful work per call varies. A single block lookup can settle a timestamp; a wallet count needs retrieval and aggregation under the right interval. Inspecting the queries reveals which of these steps the agent completes.

The recorded cases locate errors at different steps of the workflow. In the timestamp case, the agent estimates from an average block interval instead of retrieving the specified block record. In the

Table 5: **Tool use across the evaluated trajectories.** Shares are computed from 23,463 invocations. A category's frequency describes source use, not the correctness of each call.

Tool category	Calls	Share (%)
Web search	11,626	49.6
Specialized blockchain tools	8,351	35.6
URL retrieval	1,743	7.4
Social search	1,388	5.9
Code execution	355	1.5

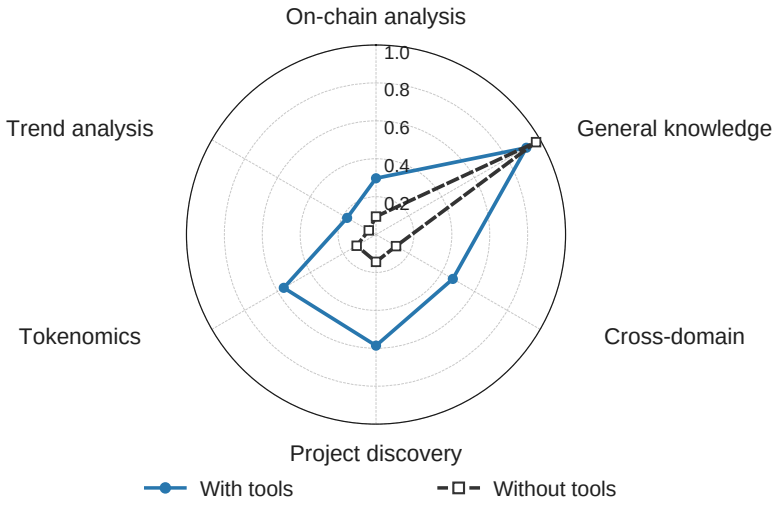


Figure 5: **Tool benefit spans different analytical categories.** Majority-vote category scores with tools (solid, filled circles) and without tools (dashed, open squares). The profiles describe different evidence requirements; Table 4 gives the overall answer measures.

wallet case, relevant transfer events still need to be converted into a set of distinct participating addresses. The first correction is a lookup; the second is an aggregation. Both are visible in the tool arguments and calculations, but total call count records neither. Trajectory inspection checks whether the agent retrieved the right record and computed the requested quantity, identifying the step that needs correction.

4.4 Efficiency Depends on the Required Reliability

Reliability also depends on the question type. Figure 5 reports majority-vote accuracy by question type. Applications can use these profiles to identify relevant categories, then inspect their reference routes and answer errors. The smaller trend and general-knowledge categories are focused diagnostics; on-chain analysis supplies the largest share of tasks.

Query costs differ substantially at similar accuracy levels. GPT-OSS-120B averages $0.0066 per tool-enabled query with 56.2% single-attempt success; Claude Opus 4 averages $1.1139 with 59.6%. GPT-5 reaches 70.2% at $0.1154. All models share the five-iteration cap, but their token usage and evaluation-time token prices differ: a larger charge does not necessarily buy more useful retrieval. Figure 1 shows what tool access adds within each model; Table 4 shows the cost of its tool-enabled answers.

A first-response service prioritizes Pass@1; a workflow that checks several candidates also needs coverage and selection accuracy. The token-charge column reports model-side costs. Deployment must additionally account for tool latency, access fees, and verification.

5 DISCUSSION

Selection should use support, not only agreement. A candidate selector could check whether cited records concern the right object and interval, and whether the calculation produces the claimed value. This connects source attribution (Gao et al., 2023; Asai et al., 2023) with answer verification (Lightman et al., 2024). CAIA's references and validated routes support testing whether such checks select correct answers more reliably than agreement alone.

A stable target in a changing information environment. The historical target remains fixed even as retrieval services change. Repeating a live evaluation tests both the agent and current source access; replaying retained tool responses holds that information fixed. These serve different purposes. The reference answer, temporal anchor, and recorded responses allow an evaluator to distinguish changes in reasoning from changes in what the tools returned.

A new selector can first be tested on the retained candidate sets, holding available answers and evidence fixed. Changes in final accuracy then measure how well the selector uses that information. A second test can let the agent request additional records before choosing, measuring the value and cost of active verification. This separates better use of existing evidence from the benefit of acquiring more. In both tests, the selector would receive the question, candidate answers, and retained evidence; the expert reference would remain reserved for scoring its selected answer.

6 RELATED WORK

Tool-grounded agents. ReAct, Toolformer, ToolLLM, and Gorilla connect language reasoning to external interfaces (Yao et al., 2023; Schick et al., 2023; Qin et al., 2024; Patil et al., 2023). WebArena, Mind2Web, GAIA, and τ-bench evaluate interaction with websites, tools, and users (Zhou et al., 2024; Deng et al., 2023; Mialon et al., 2024; Yao et al., 2024). ToolQA, LiveMCP-101, and VitaBench stress external information and multi-tool orchestration (Zhuang et al., 2023; Yin et al., 2025; He et al., 2025). CAIA follows the analytical evidence chain from source selection to a computed answer.

Financial analysis. FinQA and TAT-QA combine financial language and numerical reasoning (Chen et al., 2021b; Zhu et al., 2021); FinanceBench and Finance Agent Benchmark evaluate evidence-backed financial research (Islam et al., 2023; Bigeard et al., 2025). CAIA joins ledger queries, documentation, and historical records, making entity resolution, temporal alignment, and aggregation part of the task.

Evidence, time, and answer selection. Retrieval-augmented generation, FEVER, and HotpotQA link answers to evidence (Lewis et al., 2020; Thorne et al., 2018; Yang et al., 2018); TimeQA and RealTime QA make temporal scope explicit (Chen et al., 2021a; Kasai et al., 2022). Self-consistency selects answers by agreement (Wang et al., 2023b), while verification and attributed generation evaluate their support (Lightman et al., 2024; Gao et al., 2023). CAIA's candidate-coverage and majority-vote results motivate testing these selectors. Expert checks complement model-based judging (Zheng et al., 2023); correctness, cost, and trajectories follow HELM's multi-metric perspective (Liang et al., 2023). Appendix J develops these connections.

7 CONCLUSION

CAIA evaluates the path from an analytical request through source selection, historical alignment, and calculation to a final answer. Tool access substantially improves accuracy; repeated attempts also expose a selection bottleneck when majority voting misses correct candidates. The task bank, expert reference routes, and interaction records support evaluating both parts of this workflow: constructing the evidence needed to answer a question and selecting the answer that the evidence supports.

AI USAGE STATEMENT

Language models supported automated task filtering and response evaluation, as described in the methods, and they served as the evaluation objectives for the benchmark. **With human in the loop for supervision and quality control**, coding agents assisted with writing, better result presentation, and experiment execution. **All research ideas, structure of the manuscript, and result analysis strategies originated with the human authors**, who have reviewed and revised the codebase and the manuscript, and human authors take full responsibility for all the scientific claims and artifacts.

REPRODUCIBILITY STATEMENT

Benchmark code and data are available at `https://anonymous.4open.science/r/CAIA-ICLR2027-CCD4`. The repository includes all 178 questions with expert reference answers and category labels, a public two-tool demo agent with web search and webpage retrieval, and configurable answer-evaluation code. Its README documents direct-answer and web-tool runs, credentials, and scoring; dataset provenance records the source revision and checksum. Sections 2 and 3 describe task construction and the evaluated 23-tool interface, with the toolkit detailed in Appendix H. The manuscript source archive includes the reported numerical summaries and scripts that reproduce the figures and tables from those summaries.

REFERENCES

David Ardia and Keven Bluteau. The Role of Twitter in Cryptocurrency Pump-and-Dumps. *arXiv preprint arXiv:2306.02148*, 2023. URL https://arxiv.org/abs/2306.02148.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. *arXiv preprint arXiv:2310.11511*, 2023. URL https://arxiv.org/abs/2310.11511.

Antoine Bigeard, Langston Nashold, Rayan Krishnan, and Shirley Wu. Finance Agent Benchmark: Benchmarking LLMs on Real-world Financial Research Tasks. *arXiv preprint arXiv:2508.00828*, 2025. URL https://arxiv.org/abs/2508.00828.

Wenhu Chen, Xinyi Wang, and William Yang Wang. A Dataset for Answering Time-Sensitive Questions. In *NeurIPS Datasets and Benchmarks*, 2021a. URL https://arxiv.org/abs/2108.06314.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, et al. FinQA: A Dataset of Numerical Reasoning over Financial Data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2021b. URL https://arxiv.org/abs/2109.00122.

Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. ConvFinQA: Exploring the Chain of Numerical Reasoning in Conversational Finance Question Answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2022. URL https://arxiv.org/abs/2210.03849.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, et al. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Lin William Cong, Xi Li, Ke Tang, and Yang Yang. Crypto Wash Trading. *Management Science*, 69 (11):6427–6454, 2023. doi: 10.1287/mnsc.2021.02709. URL https://arxiv.org/abs/2108.10984.

Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, et al. Flash Boys 2.0: Frontrunning, Transaction Reordering, and Consensus Instability in Decentralized Exchanges. *arXiv preprint arXiv:1904.05234*, 2019. URL https://arxiv.org/abs/1904.05234.

Edoardo Debenedetti, Jie Zhang, Mislav Balunović, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. AgentDojo: A Dynamic Environment to Evaluate Prompt Injection Attacks and Defenses for LLM Agents. *arXiv preprint arXiv:2406.13352*, 2024. URL https://arxiv.org/abs/2406.13352.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, et al. Mind2Web: Towards a Generalist Agent for the Web. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2306.06070.

Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. Ragas: Automated Evaluation of Retrieval Augmented Generation. *arXiv preprint arXiv:2309.15217*, 2023. URL https://arxiv.org/abs/2309.15217.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling Large Language Models to Generate Text with Citations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://arxiv.org/abs/2305.14627.

Zhicheng Guo, Sijie Cheng, Hao Wang, Shihao Liang, Yujia Qin, Peng Li, et al. StableToolBench: Towards Stable Large-Scale Benchmarking on Tool Learning of Large Language Models. In *Findings of the Association for Computational Linguistics: ACL*, 2024. URL https://arxiv.org/abs/2403.07714.

Wei He, Yueqing Sun, Hongyan Hao, Xueyuan Hao, Zhikang Xia, Qi Gu, et al. VitaBench: Benchmarking LLM Agents with Versatile Interactive Tasks in Real-world Applications. *arXiv preprint arXiv:2509.26490*, 2025. URL https://arxiv.org/abs/2509.26490.

Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. FinanceBench: A New Benchmark for Financial Question Answering. *arXiv preprint arXiv:2311.11944*, 2023. URL https://arxiv.org/abs/2311.11944.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *arXiv preprint arXiv:1705.03551*, 2017. URL https://arxiv.org/abs/1705.03551.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, et al. RealTime QA: What's the Answer Right Now? *arXiv preprint arXiv:2207.13332*, 2022. URL https://arxiv.org/abs/2207.13332.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, et al. Natural Questions: A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, 2020. URL https://arxiv.org/abs/2005.11401.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, et al. Holistic Evaluation of Language Models. *Transactions on Machine Learning Research*, 2023. URL https://arxiv.org/abs/2211.09110.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, et al. Let's Verify Step by Step. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2305.20050.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, et al. AgentBench: Evaluating LLMs as Agents. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2308.03688.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: a benchmark for General AI Assistants. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2311.12983.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, et al. FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://arxiv.org/abs/2305.14251.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large Language Model Connected with Massive APIs. *arXiv preprint arXiv:2305.15334*, 2023. URL https://arxiv.org/abs/2305.15334.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, et al. KILT: a Benchmark for Knowledge Intensive Language Tasks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, 2021. URL https://arxiv.org/abs/2009.02252.

Kaihua Qin, Liyi Zhou, Benjamin Livshits, and Arthur Gervais. Attacking the DeFi Ecosystem with Flash Loans for Fun and Profit. *arXiv preprint arXiv:2003.03810*, 2020. URL https://arxiv.org/abs/2003.03810.

Kaihua Qin, Liyi Zhou, and Arthur Gervais. Quantifying Blockchain Extractable Value: How dark is the forest? *arXiv preprint arXiv:2101.05511*, 2021. URL https://arxiv.org/abs/2101.05511.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, et al. ToolLLM: Facilitating Large Language Models to Master 16000+ Real-world APIs. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2307.16789.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2016. URL https://arxiv.org/abs/1606.05250.

Yangjun Ruan, Honghua Dong, Andrew Wang, Silviu Pitis, Yongchao Zhou, Jimmy Ba, et al. Identifying the Risks of LM Agents with an LM-Emulated Sandbox. *arXiv preprint arXiv:2309.15817*, 2023. URL https://arxiv.org/abs/2309.15817.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, et al. Toolformer: Language Models Can Teach Themselves to Use Tools. *arXiv preprint arXiv:2302.04761*, 2023. URL https://arxiv.org/abs/2302.04761.

Yifan Song, Weimin Xiong, Dawei Zhu, Wenhao Wu, Han Qian, Mingbo Song, et al. RestGPT: Connecting Large Language Models with Real-World RESTful APIs. *arXiv preprint arXiv:2306.06624*, 2023. URL https://arxiv.org/abs/2306.06624.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for Fact Extraction and VERification. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, 2018. URL https://arxiv.org/abs/1803.05355.

Christof Ferreira Torres, Mathis Steichen, and Radu State. The Art of The Scam: Demystifying Honeypots in Ethereum Smart Contracts. *arXiv preprint arXiv:1902.06976*, 2019. URL https://arxiv.org/abs/1902.06976.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop Questions via Single-hop Question Composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL https://arxiv.org/abs/2108.00573.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, et al. Large Language Models are not Fair Evaluators. *arXiv preprint arXiv:2305.17926*, 2023a. URL https://arxiv.org/abs/2305.17926.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, et al. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *International Conference on Learning Representations*, 2023b. URL https://arxiv.org/abs/2203.11171.

Sam M. Werner, Daniel Perez, Lewis Gudgeon, Ariah Klages-Mundt, Dominik Harz, and William J. Knottenbelt. SoK: Decentralized Finance (DeFi). *arXiv preprint arXiv:2101.08778*, 2021. URL https://arxiv.org/abs/2101.08778.

Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, et al. FinBen: A Holistic Financial Benchmark for Large Language Models. *arXiv preprint arXiv:2402.12659*, 2024. URL https://arxiv.org/abs/2402.12659.

Jiahua Xu and Benjamin Livshits. The Anatomy of a Cryptocurrency Pump-and-Dump Scheme. In *USENIX Security Symposium*, 2019. URL https://arxiv.org/abs/1811.10109.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, et al. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2018. URL https://arxiv.org/abs/1809.09600.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. WebShop: Towards Scalable Real-World Web Interaction with Grounded Language Agents. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2207.01206.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, et al. ReAct: Synergizing Reasoning and Acting in Language Models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.03629.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains. *arXiv preprint arXiv:2406.12045*, 2024. URL https://arxiv.org/abs/2406.12045.

Ming Yin, Dinghan Shen, Silei Xu, Sixun Dong, Mian Zhang, Yebowen Hu, et al. LiveMCP-101: Stress Testing and Diagnosing MCP-enabled Agents on Challenging Queries. *arXiv preprint arXiv:2508.15760*, 2025. URL https://arxiv.org/abs/2508.15760.

Qiusi Zhan, Zhixiang Liang, Zifan Ying, and Daniel Kang. InjecAgent: Benchmarking Indirect Prompt Injections in Tool-Integrated Large Language Model Agents. In *Findings of the Association for Computational Linguistics: ACL*, 2024. URL https://arxiv.org/abs/2403.02691.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, et al. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2306.05685.

Liyi Zhou, Kaihua Qin, Antoine Cully, Benjamin Livshits, and Arthur Gervais. On the Just-In-Time Discovery of Profit-Generating Transactions in DeFi Protocols. *arXiv preprint arXiv:2103.02228*, 2021. URL https://arxiv.org/abs/2103.02228.

Liyi Zhou, Xihan Xiong, Jens Ernstberger, Stefanos Chaliasos, Zhipeng Wang, Ye Wang, et al. SoK: Decentralized Finance (DeFi) Attacks. *arXiv preprint arXiv:2208.13035*, 2022. URL https://arxiv.org/abs/2208.13035.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, et al. WebArena: A Realistic Web Environment for Building Autonomous Agents. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2307.13854.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, et al. TAT-QA: A Question Answering Benchmark on a Hybrid of Tabular and Textual Content in Finance. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*, 2021. URL https://arxiv.org/abs/2105.07624.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. ToolQA: A Dataset for LLM Question Answering with External Tools. *arXiv preprint arXiv:2306.13304*, 2023. URL https://arxiv.org/abs/2306.13304.

A Agent Information Flow

The model receives the question, the shared tool descriptions, and the tool results gathered so far. It chooses whether to gather additional evidence or synthesize an answer. Each decision includes a structured action and reasoning field. Up to five decision iterations are permitted. Multiple tools may be selected in one iteration; the iteration count therefore differs from the total invocation count. Results are appended to the agent context before the next decision. The final answer is evaluated using the same task reference and criteria in both tool conditions.

The standardized loop gives every model the same decision structure. Its policy determines which interfaces to call, which observations to retain, and when to synthesize an answer. A complete evaluation record joins that policy's transcript to the task reference and tool outputs. Fixed-transcript replay preserves the information supplied during a run; live execution exercises the current retrieval interface.

B Human Evaluation

Evaluator audit. We sampled 500 question–model pairs uniformly at random without replacement. A team of ten domain experts assessed the sampled model answers against their corresponding gold answers, assigning a binary label of correct or incorrect. Comparing these human labels with the LLM evaluator's labels yielded agreement on all 500 responses (100%).

Human task-solving pilot. Sixteen participants from university blockchain clubs and early-stage blockchain companies completed a stratified 10% subset of CAIA with tool access. Responses were checked against expert reference answers, including approved equivalent formulations. The pilot mean was 80% after removing the highest- and lowest-scoring participants. The pilot uses this stratified subset; the model comparison uses the 178-task bank. These task sets and aggregation rules define the two evaluations.

C Interpreting Tool Counts

An active model and a long trajectory are not the same unit of analysis. Model-level averages compare policies across the task bank; item-level observations compare individual attempts, where difficulty and continued searching both affect call count. The positive model-level correlation and near-zero item-level correlation can therefore coexist. A model's overall activity profile is informative about that model, but does not make the length of any one trajectory a reliable correctness signal.

D Category and Cost Profiles

The category and cost figures use the reported majority-vote accuracy summaries. They complement the Pass@1 and Pass@5 comparisons by showing how the aggregated answer varies across analytical categories and historical query costs. Appendix F gives every model's category profile. The designed task bank has the category composition specified in Table 2.

E Research Use and Data Stewardship

The questions originate from analytical requests and are standardized for evaluation. Task records should retain the entity identifiers and historical scope required for verification while excluding unnecessary contributor information. An evaluation release should preserve the task version, reference answer, scoring criteria, tool descriptions, model configuration, and response traces together. This makes changes to tool availability or external data distinguishable from changes in agent behavior.

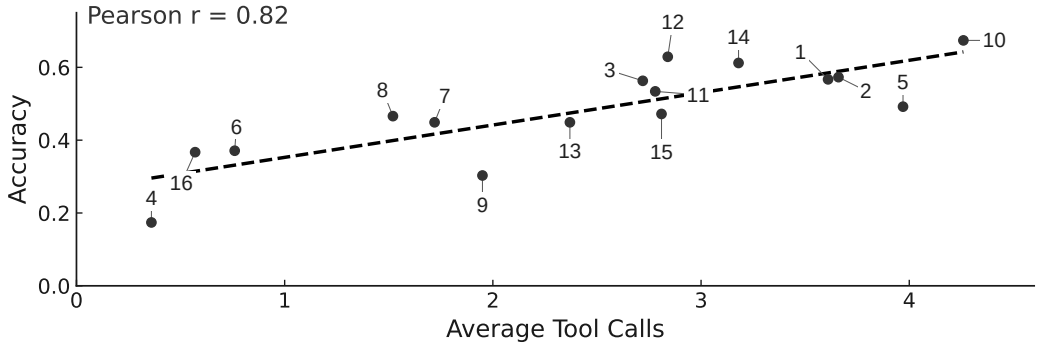


Figure 6: **Model-level call volume and accuracy.** Numbered points identify the 16 models in the key; the dashed line is the fitted between-model trend. Accuracy follows the majority-vote analysis. The positive association across models coexists with the near-zero association across individual attempts discussed in the main text.

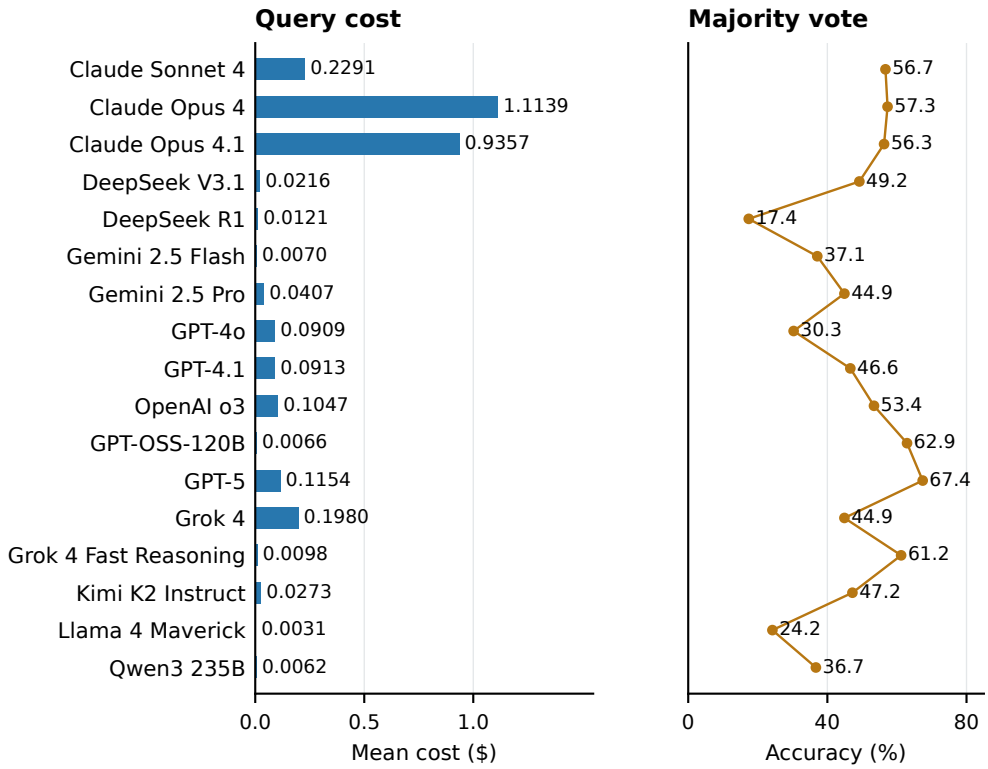


Figure 7: **Historical cost and majority-vote accuracy.** Left: evaluation-time query costs. Right: majority-vote accuracy in the same model order. Table 4 supplies the complementary single-attempt and success-within-five results. Token charges describe the evaluated configurations; tool latency and downstream verification add further operational costs.

F Complete Direct and Category Results

The direct-answer condition separates the value of repeated attempts from the value of external evidence. Table 6 gives Pass@1, Pass@5, and majority-vote accuracy, together with evaluation-time query costs. Its Pass@1 and Pass@5 columns measure how often additional direct-answer attempts produce a correct candidate. The tool-enabled comparison adds retrieval and execution through the shared action loop; majority voting measures how well consensus selects an answer from repeated attempts.

Table 6: Direct-answer results on the 178-task bank. Accuracy columns are percentages. Majority vote, Pass@1, and Pass@5 retain their distinct aggregation rules. Costs are evaluation-time mean token charges per query.

Model	Majority	Pass@1	Pass@5	Cost ($)
Claude Sonnet 4	11.8	12.9	18.0	0.0070
Claude Opus 4	13.5	13.5	16.9	0.0334
Claude Opus 4.1	13.5	15.2	17.4	0.0356
DeepSeek R1	20.8	21.9	35.4	0.0038
DeepSeek V3.1	15.7	15.7	29.2	0.0005
Gemini 2.5 Flash	18.5	20.2	21.9	0.0012
Gemini 2.5 Pro	22.5	20.2	29.8	0.0051
GPT-4.1	19.7	20.8	24.2	0.0025
GPT-4o	16.9	19.1	20.8	0.0016
GPT-5	27.5	28.1	42.7	0.0207
OpenAI o3	20.8	22.5	29.2	0.0085
GPT-OSS-120B	14.6	18.5	21.3	0.0003
Grok 4	20.2	20.2	24.2	0.0345
Grok 4 Fast Reasoning	19.1	21.3	23.6	0.0006
Kimi K2 Instruct	18.5	17.4	25.3	0.0006
Llama 4 Maverick	17.4	16.9	24.7	0.0003
Qwen3 235B	16.3	14.6	18.0	0.0010

F.1 Category-Level Answer Aggregation

The next two tables resolve the category profile into individual models. Entries are the original reported majority-vote proportions, multiplied by 100. They retain the reported category-level scoring coverage; their arithmetic mean is not the task-weighted overall score. The category counts in the main text describe the task bank.

Table 7: Direct-answer majority-vote accuracy (%) by category. GK: general knowledge; Chain: on-chain analysis; Cross: cross-domain; Project: project discovery; Token: tokenomics.

Model	GK	Chain	Cross	Project	Token	Trend
Claude Sonnet 4	100.0	10.4	7.1	10.2	0.0	0.0
Claude Opus 4	85.7	10.4	7.1	14.3	4.3	12.5
Claude Opus 4.1	100.0	10.4	7.1	12.2	8.7	0.0
DeepSeek R1	100.0	16.9	14.3	18.4	21.7	12.5
DeepSeek V3.1	85.7	14.3	21.4	16.3	0.0	0.0
Gemini 2.5 Flash	100.0	11.7	14.3	18.4	26.1	0.0
Gemini 2.5 Pro	100.0	15.6	14.3	24.5	26.1	12.5
GPT-4.1	100.0	15.6	21.4	18.4	13.0	12.5
GPT-4o	100.0	13.0	14.3	16.3	8.7	12.5
GPT-5	100.0	22.1	28.6	28.6	26.1	12.5
OpenAI o3	100.0	14.3	14.3	22.4	21.7	12.5
GPT-OSS-120B	85.7	11.7	7.1	12.2	13.0	12.5
Grok 4	100.0	14.3	7.1	24.5	17.4	12.5
Grok 4 Fast Reasoning	100.0	14.3	14.3	20.4	17.4	0.0
Kimi K2 Instruct	100.0	14.3	14.3	18.4	13.0	12.5
Llama 4 Maverick	100.0	16.9	7.1	18.4	4.3	0.0
Qwen3 235B	100.0	13.0	14.3	14.3	13.0	0.0

Table 8: Tool-enabled majority-vote accuracy (%) by category. GK: general knowledge; Chain: on-chain analysis; Cross: cross-domain; Project: project discovery; Token: tokenomics.

Model	GK	Chain	Cross	Project	Token	Trend
Claude Sonnet 4	100.0	40.3	50.0	73.5	73.9	37.5
Claude Opus 4	100.0	40.3	57.1	75.5	73.9	25.0
Claude Opus 4.1	100.0	44.0	57.1	66.7	72.7	25.0
DeepSeek R1	57.1	20.8	14.3	6.1	26.1	0.0
DeepSeek V3.1	85.7	33.8	42.9	70.8	56.5	25.0
Gemini 2.5 Flash	85.7	29.9	35.7	44.9	43.5	0.0
Gemini 2.5 Pro	85.7	31.2	50.0	57.1	60.9	12.5
GPT-4.1	100.0	32.5	50.0	59.2	56.5	25.0
GPT-4o	85.7	29.9	42.9	26.5	26.1	0.0
GPT-5	100.0	57.1	78.6	73.5	78.3	50.0
OpenAI o3	100.0	44.2	71.4	57.1	65.2	12.5
GPT-OSS-120B	100.0	54.5	57.1	71.4	78.3	25.0
Grok 4	100.0	36.4	42.9	55.1	43.5	25.0
Grok 4 Fast Reasoning	100.0	46.8	57.1	73.5	82.6	37.5
Kimi K2 Instruct	85.7	28.6	64.3	67.3	52.2	25.0
Llama 4 Maverick	71.4	23.4	28.6	18.4	26.1	12.5
Qwen3 235B	85.7	29.9	30.8	49.0	34.8	0.0

G Task Categories and Evidence Requirements

The six categories distinguish the object of an analytical request. A category describes what must be established, while a reference route describes how the available interfaces can establish it. The examples below preserve the source task descriptions and illustrate the relevant scope constraints.

On-chain analysis. These tasks concern ledger records or historical state. Examples include the timestamp of Ethereum block 19,560,000 and USDC supply at a specified block. Correctness depends on resolving the chain, object identifier, and historical reference before interpreting the returned value. An aggregate over events additionally requires the appropriate filters and unit of counting.

Project discovery. These tasks establish protocol properties and canonical references, such as the Uniswap V3 router address on Ethereum mainnet or a specified project's router contract. A project name alone may refer to several deployments or versions. Documentation, chain, and version must therefore agree with the requested object. General search is useful for discovery; the final answer requires evidence about the particular deployment.

Tokenomics. These questions concern supply, unlock schedules, vesting, and allocation rules. Counting MYSHELL tokens scheduled for release during a specified interval requires combining the applicable releases rather than quoting a single headline supply figure. A circulating-supply question also depends on the definition of circulation and the requested date. The arithmetic becomes meaningful only after those quantities are aligned.

Cross-domain questions. The source labels this category "Overlap." It combines information across assets, protocols, or analytical categories. For example, calculating Base's share of layer-2 total value locked requires a numerator and denominator with compatible dates and metric definitions. Comparing two lending protocols similarly requires consistent coverage. A plausible pair of independently retrieved values can still answer different questions.

Trend analysis. These tasks compare periods or interpret time-dependent activity, such as token launch counts on Pump.fun in two months. They require consistent event definitions, interval boundaries, and aggregation across periods. Explanatory questions about a market move require an additional distinction between an observed temporal association and a supported explanation of that move.

General knowledge. These tasks ask for explanations of mechanisms, such as the relationship between Optimism and the Superchain or the incentive structure of Bitcoin mining after block subsidies end. A successful answer connects the relevant concepts and conditions. Retrieval can supply documentation, while conceptual consistency remains central to the final explanation.

H THE FULL EVIDENCE TOOLKIT

The catalogue below records all 23 shared interfaces. The names describe the information exposed by each interface. Multiple interfaces can contribute to one question, and repeated calls to one interface can refine a query. The five-iteration limit applies to selection rounds, allowing more than five invocations when a round requests complementary observations.

Table 9: The 23-tool catalogue used in the evaluation. Interface roles are stated in terms of the evidence they return.

Tool	Evidence role
Price Feed	Price observations over a requested period.
Ranking Query	Ranked assets under the service's selected measure.
Positioning Analysis	Signals about trader positioning.
Volatility Alerts	Recent changes in market volatility.
Indicator Analysis	Derived momentum and trend indicators.
Web Search	Candidate public sources related to a query.
URL Fetching	The underlying content of a selected page.
Trend Detection	Topics or assets showing increased attention.
Twitter Search	Public posts matching a subject or query.
Social Stream	Aggregated discussion from tracked commentators.
Protocol Metrics	Activity and usage measures for a protocol.
Revenue Analysis	Measures of fees or revenue under the service's definitions.
Network Activity	Network-level activity measures.
Exchange Summary	Concentration of trading or usage across platforms.
Exchange Overview	A combined view of exchange-level signals.
Supply Query	Supply information for the specified asset.
Data Query	Queries over public ledger records.
Time Conversion	Mapping between calendar time and ledger references.
Balance Query	Holdings of a specified account.
Transaction Query	Details of a specified transaction.
Block Query	Metadata for a specified block.
Release Schedule	Scheduled asset releases and unlocks.
Code Execution	Arithmetic and programmatic checks on obtained values.

I DECISION PROTOCOL AND ERROR DIAGNOSIS

I.1 INFORMATION AVAILABLE AT EACH STAGE

Tool selection receives the question, the common tool descriptions, and the accumulated results. The prompt asks the agent to identify the information still needed, select relevant tools, and review previous observations before requesting more. The structured decision has two principal fields: a decision to call tools or proceed to synthesis, and the reasoning for that choice. Actual calls carry the chosen interface and its arguments.

Table 10: The shared information flow. All models use the same stage structure; the policy determines the actions within it.

Stage	Input	Output
Interpret	Question and tool catalogue	Required entities, quantities, and temporal scope.
Select	Question and prior observations	Further calls, or a synthesis decision.
Observe	Tool identifiers and arguments	Returned records appended to the history.
Synthesize	Question and accumulated history	Final structured answer.
Evaluate	Answer and expert reference	Correctness under the common scoring criteria.

The selection stage repeats up to five times. A synthesis decision ends information gathering early; exhaustion of the iteration budget also triggers the final answer stage. This protocol exposes planning, retrieval, and synthesis as separate parts of a trace. It also distinguishes the number of decision rounds from the number of tools or calls, which can vary within a round.

I.2 Failure Types in the Qualitative Cases

The block-timestamp and unique-wallet cases identify several concrete failure types. They provide categories for reading a trajectory rather than frequency estimates for the whole benchmark.

Table 11: Trace-level error categories and the evidence needed to diagnose them.

Failure	Diagnostic observation
Wrong object	A query targets a different chain, deployment, pool, contract, or asset.
Wrong time	The response describes the current state or a different interval from the question.
Wrong source	A secondary snippet substitutes for the record needed to establish the claim.
Wrong quantity	The agent counts transfers instead of distinct participating wallets, or confuses supply definitions.
Aggregation error	Relevant records are retrieved but combined with an incorrect filter, boundary, or formula.
Unsupported synthesis	The final answer adds a value or conclusion not established by the retrieved evidence.

For the unique-wallet task, event retrieval and deduplication are separate operations. Identifying the right collection and quarter still leaves the question of which addresses count as participants. For the block-timestamp task, the relevant record directly supplies the answer; estimating the timestamp from an average block interval introduces avoidable error. These examples show why an agent's evidence route is informative even when its final answer is simply scored correct or incorrect.

I.3 Source Reliability and Historical State

The information environment presents three practical risks: stale or promoted search results, unofficial or spoofed project pages, and strategically amplified social claims. Their relevance depends on the requested fact. A social post can be evidence about discussion, whereas a canonical contract address calls for deployment evidence. An apparent disagreement between two sources can also arise from different dates or metric definitions.

Temporal anchors make the answer target explicit. Preserving the corresponding tool observations additionally fixes what the agent could see during that evaluation. This distinction matters for future comparisons: a new model queried against live services may encounter different source availability even when the intended historical answer is unchanged. Versioned tasks and archived observations allow these changes to be analyzed separately.

J Evidence Construction in Context

From a supplied document to an acquired evidence set. FinQA, TAT-QA, and ConvFinQA evaluate numerical reasoning over financial reports, tables, and conversational questions (Chen et al., 2021b; Zhu et al., 2021; Chen et al., 2022). Their inputs make the relevant document collection part of the problem. CAIA adds evidence acquisition: the agent decides which interface can establish the requested fact and supplies the entity, interval, and aggregation arguments. Its reference toolchain links the final answer to one validated acquisition route. This makes failures such as querying the wrong pool or counting transfers instead of wallets visible as errors in constructing the analytical quantity.

FinanceBench evaluates open-book financial question answering with reference evidence (Islam et al., 2023). FinBen broadens coverage across financial analysis, forecasting, and decision-making (Xie et al., 2024), while Finance Agent Benchmark equips agents to research expert-authored questions using financial filings and search (Bigeard et al., 2025). CAIA complements these settings with a cryptocurrency-specific evidence problem: combining historical ledger state, project documentation, and external records through heterogeneous tools. The domain matters because the entity named in a question can correspond to a token, contract, pool, or protocol, each supporting different valid measurements.

Evidence dependencies in question answering. SQuAD studies answers grounded in a supplied passage, and TriviaQA pairs questions with retrieved evidence documents (Rajpurkar et al., 2016; Joshi et al., 2017). Natural Questions uses naturally occurring search questions over Wikipedia (Kwiatkowski et al., 2019); KILT combines knowledge-intensive tasks with provenance evaluation (Petroni et al., 2021). HotpotQA and MuSiQue make dependencies among supporting facts central to multi-hop reasoning (Yang et al., 2018; Trivedi et al., 2022). In CAIA, an analogous dependency can span tool types: identify the correct contract, retrieve events within the requested interval, then calculate the quantity. The reference route records one way to establish those dependencies, while the correctness criterion admits alternative routes that establish the same answer.

Historical state is part of the answer. TimeQA associates questions with time-sensitive facts, and RealTime QA evaluates answers about newly developing events (Chen et al., 2021a; Kasai et al., 2022). CAIA anchors an analytical request to a particular historical state where relevant. The same collection can have different wallet counts across quarters; the same protocol can have different rules across versions. A source's authority and its temporal applicability therefore answer different questions. The agent must preserve both while combining records. Research on cryptocurrency pump-and-dump promotion and transaction ordering makes the surrounding source incentives concrete (Xu & Livshits, 2019; Ardia & Bluteau, 2023; Daian et al., 2019).

Domain semantics determine what a quantity means. DeFi research distinguishes protocol primitives and technical from economic security (Werner et al., 2021). Studies of flash loans and blockchain extractable value show why transaction ordering, atomicity, and intermediate state matter to financial interpretation (Qin et al., 2020; 2021). Analyses of DeFi attacks and profit-generating transaction sequences further expose dependencies across protocol actions (Zhou et al., 2022; 2021). CAIA's analytical questions operate within this domain: retrieving a number becomes useful when its contract, time interval, and aggregation semantics agree with the request. This motivates preserving the evidence chain alongside the final answer.

Access to a source and reliability of its contents. Crypto Wash Trading documents distortions in reported exchange activity, and honeypot research examines misleading smart-contract behavior (Cong et al., 2023; Torres et al., 2019). For tool-using agents, AgentDojo and InjecAgent study how external content can carry adversarial instructions (Debenedetti et al., 2024; Zhan et al., 2024); ToolEmu tests risky agent behavior in emulated tool environments (Ruan et al., 2023). These works identify distinct questions about the data source, interpretation, and action policy. CAIA measures analytical answer correctness and preserves trajectories that reveal source choice and use. The resulting traces connect a final claim to the observations and computations supporting it.

Candidate generation and evidence-based selection. Self-consistency aggregates sampled answers by agreement (Wang et al., 2023b); verifier-based reasoning evaluates candidate correctness

(Cobbe et al., 2021; Lightman et al., 2024). Retrieval-augmented and attributed generation connect outputs to external support (Lewis et al., 2020; Gao et al., 2023; Asai et al., 2023). CAIA exposes these as distinct requirements through Pass@1, Pass@5, and majority voting. A candidate can be frequent yet use the wrong interval; a rarer answer can be correct because it retrieves the requested historical record. A provenance-aware selection policy can inspect those distinctions using the candidates' actual tool outputs. The benchmark's final-answer criterion leaves that policy free to find any valid evidence route.

Separating retrieval quality, factual support, and correctness. FActScore decomposes generated text into claims supported by a knowledge source (Min et al., 2023); Ragas evaluates retrieval and generation along separate dimensions (Es et al., 2023). Model-judge studies examine agreement with people and sensitivity to presentation order (Zheng et al., 2023; Wang et al., 2023a). CAIA uses a reference-based final-answer judgment with the reported expert audit, while its trajectories expose how the answer was obtained. These two views serve different reading needs: the score determines whether the requested analysis is correct, and the trace explains which evidence or computation produced that outcome. An answer selector can exploit this structure by comparing the evidence underlying competing candidates.

Common interfaces and diagnostic trajectories. AgentBench and WebShop show how common environments support comparison across agent policies (Liu et al., 2024; Yao et al., 2022). ToolQA emphasizes questions requiring external tools, and RestGPT decomposes requests into API selection and execution (Zhuang et al., 2023; Song et al., 2023). StableToolBench addresses changing API behavior through caching, simulation, and stable evaluation (Guo et al., 2024); LiveMCP-101 and VitaBench stress multi-tool orchestration and interactive requests (Yin et al., 2025; He et al., 2025). CAIA fixes a shared interface and iteration budget while preserving each agent's chosen calls. A call count summarizes activity; a trace identifies the uncertainty that each call addresses. Comparing the requested claim, retrieved evidence, and final answer makes source routing, temporal alignment, and computation concrete diagnostic targets. These targets explain why the benchmark reports outcomes and action histories together.