STAR-BENCH: EVALUATING ANTI-MONEY LAUNDERING AGENTS FOR REGULATORY REPORTING WORKFLOWS

Anonymous authors
Paper under double-blind review

ABSTRACT

Anti-money laundering (AML) investigations are regulated tool-using workflows that end in a suspicious transaction report (STR), and they are harder than general tool use in four ways: tools reading the same fields serve different regulatory roles, reporting is an obligation, evidence must carry across turns, and models are chosen by proxies whose bearing on this work is unknown. Existing function-calling and financial benchmarks do not score whether an agent carries out a mandated reporting step. We introduce `STAR-Bench`, a compliance-oriented benchmark derived from an AML agent platform and expert-verified scenarios: 23 AML tools and 1,308 evaluation cases spanning single-turn tool use and multi-turn STR workflows, where every tool executes against the database. Experiments on 28 open-weight configurations show that single-turn tool accuracy does not resolve workflow readiness: configurations within seven points on single-turn tool hit differ by thirty-two points in STR workflow completion, and the best completer ranks 17th of 28 on single-turn hit. Workflow failures concentrate at the STR-writing turn and persist under an oracle that injects executed tool outputs, while end-to-end execution costs parameter grounding rather than tool selection. Reporting tools are about as hard to trigger as analysis tools on average but fail differently: three quarters of STR-field validation failures are calls never made. General function-calling rank, finance specialization and schema language give no reliable ordering of these results, and reasoning mode lowers them on average under our output budget. We release the `STAR-Bench` dataset, executable evaluation environment, evaluation code[1], and AML agent platform.[2]

1 INTRODUCTION

Money laundering is estimated to account for 2–5% of global GDP, and the amounts available for laundering were put at US$1.6 trillion for 2009 (United Nations Office on Drugs and Crime, 2011). To combat it, financial institutions operate anti-money laundering (AML) programs that monitor suspicious transactions, investigate alerts, and decide whether cases must be reported to regulators. This process remains labor-intensive: of roughly 16 million alerts US institutions reviewed in 2017, 4% became a report, and 4% of those reports drew law-enforcement contact (Bank Policy Institute, 2020). The burden grows as open finance, instant payments, and agent-mediated payment systems accelerate the volume and complexity of financial activity (Juneja et al., 2024; PwC, 2023; Aldasoro & Desai, 2025; Chua et al., 2026).

AML investigations are regulated tool-using workflows. An analyst queries transaction records, inspects account behavior, detects suspicious patterns, analyzes money flows, checks reporting obligations, and decides whether a case must be reported to the financial intelligence unit (FIU) as a suspicious transaction report (STR) or, for large cash activity, a cash transaction report (CTR). Progress in function calling (Patil et al., 2024) and regulatory interest in AI for AML operations (Financial Action Task Force, 2021) make an agentic system for parts of this workflow plausible.

[1] https://anonymous.4open.science/r/STAR-Bench-157A

[2] https://anonymous.4open.science/r/STAR-Bench-Web-B163

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.

Four properties of the workflow make it harder than general tool use. First, tool roles overlap (C1): a transaction query and a CTR-candidate detector read the same account fields, and only the second involves a reporting decision, so confusing them mishandles a reporting obligation. Second, reporting is an obligation (C2): FATF Recommendation 20 requires an institution that suspects criminal proceeds to report them to the FIU, whatever the amount (Financial Action Task Force, 2012), and the benchmark scores the requested reporting step as a tool call, so skipping it counts as a missed action rather than being judged as prose. Third, evidence has to carry across turns (C3): an STR is assembled from the outputs of earlier tools (Naik et al., 2025), and a call with the right name still fails when its arguments come from a result the agent lost. Fourth, models are chosen by proxy (C4): practitioners select models by function-calling leaderboards, finance or language specialization, schema language and reasoning mode, none of them validated on this workflow.

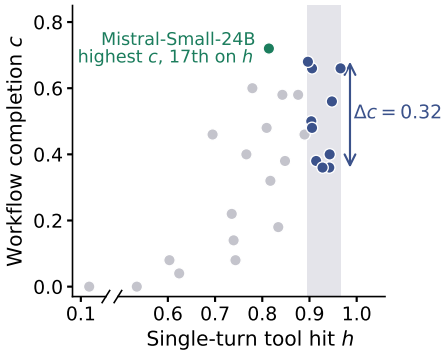


Figure 1: Single-turn tool hit h against STR workflow completion c, one point per serving configuration. The configurations within seven points of one another on h (blue, shaded band) still differ in completion by Δc=0.32; Mistral-Small-24B (green) completes the most workflows.

Existing benchmarks do not evaluate this combination. General function-calling benchmarks such as BFCL, T-Eval, and ToolSandbox (Patil et al., 2025; Chen et al., 2024; Lu et al., 2025) cover C1 in general domains and parts of C3, and financial benchmarks test knowledge, reasoning and document retrieval (Islam et al., 2023; Xie et al., 2024). None of them scores a mandated reporting step (C2).

We introduce `STAR-Bench` (**S**uspicious **T**ransaction and **A**ctivity **R**eporting **Bench**mark), a compliance-oriented benchmark for evaluating AML agent readiness, derived from an AML agent platform and grounded in de-identified synthetic interbank funds transfer data supplied by the national institution that clears such transfers between banks: 23 executable tools and 1,308 expert-authored, cross-reviewed cases over single-turn tool use and 50 multi-turn STR workflows, in Korean and English under a controlled query $\times$ tool-schema ablation. Similar and decoy tools test C1 (Section 4.2); a Regulatory Reporting subdomain with a failure taxonomy and an STR checker tests C2 (Section 4.3); multi-turn scenarios under an executed oracle and under end-to-end execution test C3 (Section 4.4); and paired comparisons across the proxies test C4 (Section 5).

Across 28 open-weight configurations, single-turn tool hit does not follow model scale, and the hardest tools are those whose inputs other tools also accept (C1). Reporting tools are about as hard to trigger as analysis tools but fail in their own ways: STR-field validation fails by going uncalled, CTR detection by giving way to a general query, and FIU reference lookup by receiving the wrong keyword after the right tool is chosen (C2). Single-turn accuracy tracks workflow completion only loosely: as Figure 1 shows, configurations within seven points of one another on single-turn tool hit complete a third to two thirds of STR workflows, failures concentrate at the STR-writing turn, and end-to-end execution costs argument accuracy more than tool choice (C3). No proxy reliably orders configurations on this workflow, and reasoning mode lowers scores on average (C4).

Our contributions are as follows:

- `STAR-Bench`, a compliance-oriented benchmark of expert-authored cases in which every tool executes against the transaction database, so single-turn and end-to-end runs answer the model's own calls with real results.
- An executed oracle: the oracle setting injects the outputs the tools produce rather than written-out ones, and it is separated from end-to-end execution, so a failure is attributed to the workflow stage rather than to accumulated error.
- We release the benchmark, the executable tool layer, the run records and the analysis, so every number recomputes without serving a model again.

Table 1: Comparison with prior function-calling benchmarks. **Expert**: cases authored and cross-reviewed by domain experts. **Bilingual**: ✓ for a controlled query × tool-definition language ablation, △ for multilingual queries without one. n/r: not reported. **Mandated report**: scores whether a mandated reporting step is taken and grades the report against earlier tool outputs.

Benchmark	Cases	Tools	Domain	Expert	Multi-turn	Bilingual	Mandated report
T-Eval (2024) (Chen et al., 2024)	23,305	15	General	✗	✗	✗	✗
ToolSandbox (2024) (Lu et al., 2025)	1,032	34	General	✗	✓	✗	✗
BFCL (2025) (Patil et al., 2025)	5,551	n/r	General	✗	✓	△	✗
ComplexFunc. (2025) (Zhong et al., 2025)	1,000	43	Travel	✗	✗	✗	✗
NESTful (2025) (Basu et al., 2025)	1,861	900+	Math	✗	✗	✗	✗
OrcheSTR. (2026) (Ahn et al., 2026)	706	∼100	General	✗	✓	✓	✗
STAR-Bench (2026)	**1,308**	**23**	**AML**	✓	✓	✓	✓

2 Related Work

2.1 Anti-Money Laundering as a Regulated Workflow

The FATF Recommendations set the baseline for anti-money laundering (AML): customer due diligence, know your customer, suspicious transaction reports (STRs), and cash transaction reports above a value threshold (Financial Action Task Force, 2012). AML work is therefore more than suspicious-transaction detection: once activity is flagged, an analyst decides whether it triggers a reporting obligation. Recommendation 20 attaches that obligation to suspicion and, under its Interpretive Note, to every suspicious transaction whatever the amount, which makes AML a measurable setting. In other regulated domains an agent is judged on what it writes; here it must also take a specified action. These workflows are hard to evaluate because a failure appears far from where it occurs: the tools of an investigation, from transaction lookup through money-flow analysis to STR-field validation and drafting, share inputs while serving different roles, so an agent that substitutes one for another returns useful-looking information and still fails the reporting step, a failure visible only at the workflow level.

2.2 Financial and AML Evaluation

Financial LLM benchmarks span knowledge, reasoning and safety (Islam et al., 2023; Xie et al., 2024) and agentic retrieval over financial documents (Choi et al., 2025), while AML research addresses suspicious-transaction detection, graph-based monitoring and explainability (Han et al., 2020; Kute et al., 2021). Neither line tests whether an agent can select a reporting tool, hold investigation state across turns, or write an STR grounded in what earlier tools returned. Agentic compliance systems such as CoInvestigator (Naik et al., 2025) show that such support is possible without providing a benchmark that measures it.

2.3 Function Calling and Agent Benchmarks

General tool-use benchmarks ask whether an LLM can select APIs and construct valid calls: API selection (Patil et al., 2024), structured call matching (Patil et al., 2025), and coverage and task decomposition (Li et al., 2023; Qin et al., 2024; Wu et al., 2024; Chen et al., 2024); others add nested or constrained calls (Basu et al., 2025; Zhong et al., 2025), stateful conversation, user interaction and multi-domain orchestration (Lu et al., 2025; Ahn et al., 2026; Yao et al., 2024; Alkhouli et al., 2025), instruction following and robustness under perturbation (Qi et al., 2025; Rabinovich & Tavor, 2025), and general agentic capability (Liu et al., 2024; Mialon et al., 2024; Ma et al., 2024). What none of them scores is a reporting step kept apart from the analysis that feeds it. The nearest is τ-bench (Yao et al., 2024), which tests rule-following under the operator's own policy and grades the database state a conversation ends in; here the reporting call is scored separately from analysis calls that read the same fields, and the report it produces is graded against the tool outputs before it. Table 1 sets out the differences. STAR-Bench asks whether an agent can move from investigation to STR generation without losing the evidence, the tool role or the workflow state it needs.

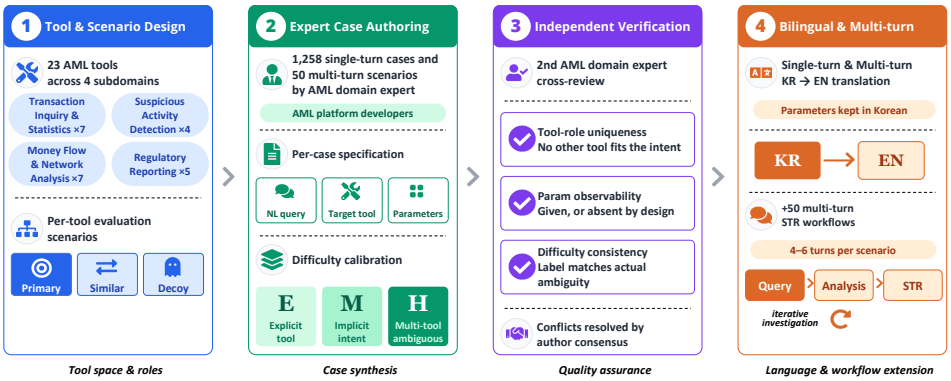


Figure 2: Expert-driven construction pipeline of STAR-Bench: tool and scenario design, expert case authoring, independent verification, and bilingual and multi-turn extension.

3 Benchmark Design

3.1 Tool Suite

STAR-Bench contains 23 tools grouped into 4 AML subdomains: transaction inquiry and statistics, suspicious-activity detection, money-flow and network analysis, and regulatory reporting. They are derived from the core modules of an AML agent platform designed with AML practitioners and correspond to the stages of an AML investigation. Each benchmark tool executes the platform's own code, and both are released. Tools within a subdomain share inputs and partially overlap in function, which makes semantic disambiguation central to the benchmark. `generate_str`, which composes an STR from prior investigation results, is evaluated only in the multi-turn STR workflow; Table 5 in Appendix A summarizes the full suite.

3.2 Task Definition

In the single-turn setting, each example pairs a natural-language query q with a shared tool set $\mathcal{T} = \{t_1, \ldots, t_n\}$. The model produces tool calls $\hat{F} = [\hat{f}_1, \ldots, \hat{f}_k]$, each $\hat{f}_i = (\hat{t}_i, \hat{p}_i)$ a selected tool $\hat{t}_i \in \mathcal{T}$ and a parameter dictionary $\hat{p}_i$, and $\hat{F}$ is compared against the gold sequence F^*. Examples are single-tool, multi-tool, missing-parameter, or abstention cases: multi-tool examples require several tools in a specified order within one query, while missing-parameter cases test whether the model can tell when a call lacks information it requires. We analyze failures along five aspects: wrong tool selection, missing gold tools, over-calling, parameter violations, and incorrect call order.

The multi-turn setting extends this to STR workflows. Each scenario runs four to six turns that simulate an iterative AML investigation. The main evaluation is an executed oracle: after each turn the next receives the gold call's output as the tool layer produced it, rather than a written-out stand-in, so turn-level tool selection is measured without accumulated execution errors. Section 4.4 reports an end-to-end analysis over the same scenarios, where each model's own tool outputs are carried into subsequent turns, for the 25 configurations whose context window admits its longer prompts.

3.3 Construction Pipeline

STAR-Bench is constructed through a four-stage expert-driven pipeline, shown in Figure 2. First, AML platform developers define each tool's operational role, required arguments, semantically similar alternatives, and decoy tools. Second, AML domain experts author natural-language queries, expected tool calls, parameter constraints, and difficulty labels under that specification. Third, a second AML expert reviews each case for defects against three criteria: tool-role uniqueness, parameter observability and difficulty consistency, defined in Appendix D. A case that fails one is revised until author and verifier agree. Finally, Korean queries and tool definitions are translated into English for bilingual deployment analysis, while account identifiers, transaction attributes, and regulatory values are kept fixed across languages.

3.4 BENCHMARK COMPOSITION

`STAR-Bench` contains 1,308 evaluation cases spanning single-turn tool use and multi-turn STR workflows. Both parts use the de-identified synthetic interbank funds transfer environment (2021 Q3–2024 Q4). The transaction environment is highly imbalanced, with suspicious transactions appearing at roughly a 1:325 ratio. It spans six suspicious-activity types: abrupt transaction-pattern changes, new-counterparty transfers, structuring, concurrent multi-transaction requests, same-day withdrawal after large deposits, and large late-night transfers. Single-turn cases are organized along three axes: difficulty, call type, and category, where difficulty reflects both semantic ambiguity among similar tools and information completeness in the query.

Multi-turn STR scenarios are divided into base, missing-parameter, and long-context categories. Base scenarios provide sufficient information for sequential tool use; missing-parameter scenarios require clarification before tool execution; and long-context scenarios require the model to reuse prior tool results in later turns. The gold STR summaries follow a seven-part narrative template derived from the Korean STR form; Appendix E gives the sections and what anonymization removes from them. The obligations these scenarios test are set by FATF (Financial Action Task Force, 2012): examining unusual transaction patterns for suspicion (Interpretive Note to Recommendation 10), reporting what that examination finds (Recommendation 20), and keeping the analysis as evidence (Recommendation 11). Every jurisdiction supplies its own form schema, reference codes and threshold; Appendix G maps the two levels. Tables 6 and 7 in Appendix B show a representative single-turn case and a multi-turn STR scenario.

3.5 EVALUATION METRICS

Single-turn cases are scored on five metrics: h, tool hit, whether every gold tool is among the calls; r, the fraction of gold tools called; p, the fraction of calls that name a gold tool; a, parameter accuracy over the gold's key-value constraints; and o, which is 1 when the tools the gold orders first appear in that order and 0 otherwise. p, a and o are undefined rather than perfect where they do not apply, so a mean over one of them is taken over the cases that have it and carries that count. h is the primary single-turn metric because it measures whether the required AML tool is triggered at all; the rest diagnose parameter grounding and multi-tool sequencing. For workflows we report $\bar{h}$ and $\bar{a}$, tool hit and parameter accuracy averaged over a configuration's turns, and c, the fraction of scenarios in which every turn achieves $h = 1$. c counts tool selection only: argument accuracy is $\bar{a}$ and the written report is scored separately, so a completed workflow can still end in a weak report. c is the workflow-level metric, and Section 4.4 shows that h does not substitute for it. Across both settings, parser failures are scored zero, and an abstention on an irrelevant or underspecified query counts as a successful refusal.

For STR generation, we score the scenarios whose ground truth includes a `generate_str` turn. We first report the production rate, the fraction of such scenarios in which the model invokes `generate_str`. Produced drafts are then evaluated with a deterministic evidence checker over three dimensions: required-field completeness, coverage of the expected STR fields and sections; evidence grounding, the share of numeric factual slots (identifiers, amounts, percentages, probabilities) that appear in the tool outputs of preceding turns; and regulatory terminology use. A scenario without an STR scores zero on all three, and a report that states no figure scores zero on grounding. The overall score is their mean.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

We evaluate 24 open-weight models as 28 thinking and non-thinking configurations, 18 served locally with vLLM v0.20.1 (Kwon et al., 2023) and 10 through a commercial inference gateway serving the same open-weight models, each from its own registry entry, so a table row names one serving configuration and not a model family. The cohort is open-weight because the jurisdiction this benchmark models keeps systems that process transaction data separated from external networks, and its 2026 relaxations cover office software rather than generative AI, so an AML agent runs on-premises and a model reachable only through an external API needs a case-by-case exemption to run at all (Appendix G). All are evaluated through their native function-calling interface with a fixed system

Table 2: Overall per-model performance on STAR-Bench for all models (Korean prompts). (T) and (NT) denote thinking mode on and off. Per column, the best value is in **bold** and the second is underlined.

Model	Single-turn					Multi-turn		
	h	r	p	a	o	$\bar{h}$	$\bar{a}$	c
Korean-Specialized								
A.X-4.0 (72B)	0.905	0.915	0.931	0.857	0.943	0.909	0.799	0.660
Kanana-2-Think	0.897	0.916	0.897	0.870	1.000	**0.922**	0.783	0.680
EXAONE-4.0-32B	0.890	0.897	0.877	0.875	0.977	0.849	0.740	0.460
Kanana-2-Instruct	0.809	0.824	0.738	0.770	0.912	0.840	0.702	0.480
EXAONE-4.0-1.2B	0.743	0.786	0.737	0.702	0.971	0.689	0.459	0.080
A.X-4.0-Light (7B)	0.534	0.489	0.881	0.461	**1.000**	0.352	0.253	0.000
Finance-Specialized								
Qwen-Open-Finance-R-8B	0.843	0.846	0.904	0.819	1.000	0.881	0.762	0.580
Llama-Open-Finance-8B	0.735	0.747	0.789	0.653	1.000	0.703	0.588	0.220
General-Purpose								
Gemma-4-31B	**0.966**	0.974	0.929	0.947	1.000	0.904	**0.899**	0.660
Qwen3.5-27B (NT)	0.948	0.967	0.892	0.950	1.000	0.890	0.860	0.560
Qwen3.6-27B	0.943	**0.980**	0.881	**0.960**	1.000	0.840	0.847	0.400
Qwen3.5-27B (T)	0.942	0.974	0.900	0.950	1.000	0.822	0.834	0.360
Qwen3.6-35B-A3B	0.928	0.960	0.860	0.942	1.000	0.813	0.808	0.360
Qwen3.5-4B (T)	0.914	0.954	0.854	0.907	1.000	0.826	0.801	0.380
Qwen3.5-4B (NT)	0.905	0.921	0.851	0.871	1.000	0.849	0.784	0.480
gpt-oss-120B (NT)	0.904	0.927	0.903	0.901	1.000	0.872	0.858	0.500
gpt-oss-20B (NT)	0.876	0.906	0.876	0.854	0.949	0.890	0.772	0.580
gpt-oss-120B (T)	0.848	0.865	0.826	0.858	1.000	0.813	0.749	0.380
Ministral-3-3B	0.834	0.862	0.831	0.765	0.968	0.685	0.592	0.180
gpt-oss-20B (T)	0.817	0.827	0.878	0.811	0.967	0.740	0.701	0.320
Mistral-Small-24B	0.814	0.801	0.882	0.791	1.000	0.922	0.811	**0.720**
Llama-3.3-70B	0.779	0.865	0.788	0.825	1.000	0.900	0.692	0.600
Gemma-4-E4B	0.766	0.746	0.914	0.733	1.000	0.840	0.753	0.400
xLAM-2-3B	0.739	0.757	0.723	0.651	0.962	0.571	0.457	0.140
xLAM-2-70B	0.695	0.664	**0.936**	0.650	1.000	0.849	0.626	0.460
Hermes-3-8B	0.624	0.612	0.830	0.551	1.000	0.507	0.412	0.040
Llama-3.2-3B	0.603	0.704	0.508	0.580	0.909	0.594	0.440	0.080
Phi-4-mini	0.120	0.059	0.673	0.057	–	0.361	0.323	0.000

prompt, the full tool schema, temperature 0, and at most five tool-calling rounds per single-turn case and one per workflow turn. Case-level bootstrap resampling with 10,000 iterations shows stable rankings (Kendall τ=0.962, 95% CI $[0.935, 0.985]$). Single-turn and end-to-end runs execute the model's own calls, and the oracle setting injects the executed outputs of the gold calls instead; the Reproducibility Statement gives the serving conditions in full and the record each run writes. Table 2 reports every configuration on both settings.

4.2 AML Tool Operation

AML tool operation is not determined by model size alone. Single-turn tool hit spans 0.120 to 0.966 across the cohort, and larger models do not always rank higher: Mistral-Small-24B and Ministral-3-3B both place above Llama-3.3-70B.

Difficulty varies by tool. Per-tool hit ranges from 0.651 for general statistics to 0.908 for fraud prediction, and the hardest tools are the ones whose inputs several other tools also accept. Wrong-tool errors follow two patterns. First, models often confuse tools with similar inputs but different analytic roles, such as account-profile and receiving-account-profile tools. Second, when specialized AML tools are not recognized, models frequently fall back to broad inquiry tools such as statistics or transaction queries. Smaller models also show more schema-level failures, including missing required parameters and invalid arguments.

4.3 How Reporting Fails

Reporting tools are not uniformly harder than analysis tools, but they fail differently. Figure 7 in Appendix H compares tool hit between the Regulatory Reporting subdomain and the remaining AML tools. Regulatory Reporting comprises four tools: STR-field validation, CTR-candidate detection, FIU reference-type lookup, and AML glossary lookup. Averaged over the cohort the two groups sit within a point and a half of one another, and 13 of 28 configurations are worse on reporting. The distinctive property is not the rate of failure but its shape. Across analysis tools, 44% of failures are wrong-tool substitutions and 23% are no-calls. STR-field validation inverts this: 75% of its failures are no-calls, the model answering without ever invoking the reporting tool, and AML glossary lookup behaves the same way at 66%. CTR-candidate detection fails a third way, with 40% wrong-tool fallback to a general transaction query in place of the regulatory decision.

Regulatory terminology fails at the parameter rather than at the tool. FIU reference-type lookup is selected correctly in 87.6% of cases and grounded correctly in 53.0%, a 34.6-point gap and the widest of any tool in the suite. Models reach the right tool and then fail to convert Korean regulatory terminology into the search keyword the tool expects. These three patterns imply different remedies. A no-call means the model leaves the reporting step it was asked for undone; a wrong-tool fallback means it does not separate reporting from analysis; a keyword error means it is not grounded in local regulatory vocabulary. A single aggregate hit rate hides all three, which is why `STAR-Bench` reports them separately.

Reporting weaknesses persist after the tool is called. Table 3 scores the reports themselves with the checker of Section 3.5, which checks their form, terminology and the provenance of their figures. It reports six configurations spanning the best report score, the best workflow completion, the highest single-turn tool hit, and both finance-specialized models. Many models rarely reach the report-writing step at all, re-issuing analysis tools instead of producing the report, so their STR scores stay near zero. Among models that do reach the stage, stronger models fill required STR fields and cite figures from prior tool outputs more reliably, but regulatory terminology use remains weak across the board, even for finance-specialized models. Completion and the checker score answer different questions: Mistral-Small-24B completes the most workflows and Kanana-2-Instruct writes the highest-scoring reports, so completion says whether an agent carries each step through and the checker what the report it writes contains. The aggregate hides one failure: Llama-3.3-70B produces an STR in all 45 scenarios and scores 0.075 on field coverage, because it writes in English a report whose fields and terms are defined in Korean. Appendix F reports how closely two practitioners reading the same reports agree with this scoring.

Table 3: STR checker scores over the 45 scenarios whose gold ends in report writing. Rate is the production rate; Field, Ground. and Term are required-field completeness, evidence grounding and terminology use, and Overall is their mean. Each is penalized so that a scenario without a report scores zero.

Model	Rate	Field	Ground.	Term	Overall
Kanana-2-Instruct	0.933	0.625	0.714	0.285	0.541
Qwen-Open-Finance-R-8B	0.911	0.636	0.823	0.137	0.532
Mistral-Small-24B	0.867	0.584	0.775	0.156	0.505
Gemma-4-31B	0.756	0.550	0.544	0.278	0.457
Qwen3.5-27B (NT)	0.667	0.498	0.524	0.181	0.401
Llama-Open-Finance-8B	0.644	0.328	0.459	0.096	0.294

4.4 Workflow Readiness Beyond Single-Turn Calls

The central result of this paper is that single-turn tool accuracy does not resolve workflow completion. Single-turn tool hit does order the cohort loosely (ρ=0.55, p=0.002), because a model that cannot call a tool at all also cannot finish a workflow. It does not, however, separate the models a deployment would choose between. The ten configurations within seven points of one another on single-turn tool hit, from 0.897 to 0.966, complete anywhere from 36% to 68% of STR scenarios, a spread of 32 points, and the configuration with the highest completion rate, Mistral-Small-24B at 0.72, ranks 17th of 28 on single-turn tool hit at 0.814. The spread is larger than scenario sampling accounts for: Gemma-4-31B completes 30 points more of the workflows than Qwen3.5-27B (T), with a 95% paired bootstrap interval of $[16, 44]$ points over the 50 scenarios and McNemar p=0.0007, and Mistral-Small-24B leads that same configuration by 36 points, $[20, 52]$, p=0.0001. Within the workflow, turn-level hit and scenario completion agree almost exactly (ρ=0.99); the disagreement lies between single-turn and workflow evaluation. A benchmark that stops at one turn therefore cannot rank agents for this deployment.

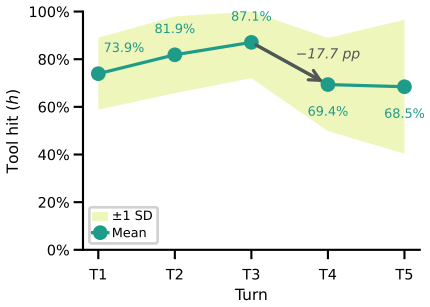


Figure 3: Tool hit h by turn: mean with a ± 1 SD band across all configurations. T4 is where 28 of the 50 scenarios ask for the STR; the arrow marks the 17.7-point fall from T3.

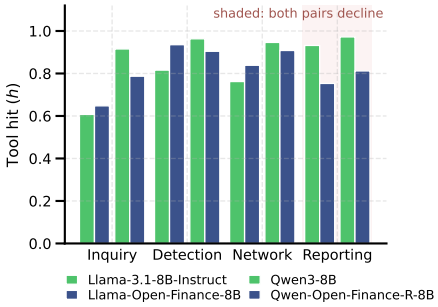


Figure 4: Each finance-specialized model against the model it was tuned from, by AML subdomain. The shaded subdomain, Reporting, is the only one where both pairs decline.

The workflow breaks at the reporting stage. Figure 3 reports mean tool hit by turn. Under the oracle setting, which injects the ground-truth call and its executed output after every turn, hit climbs through the investigation turns, from 0.739 at the first to 0.871 at the third, then falls to 0.694 at the fourth. The fourth turn is where 28 of the 50 scenarios ask for the report to be written and 22 do not. At that same turn, STR turns score 0.564 against 0.859, a gap of 29.5 points (95% scenario-bootstrap interval $[20.8, 38.2]$) that holds within each scenario type and after regression adjustment for prompt length (Appendix H). The drop survives an oracle that erases every earlier mistake and holds at a fixed turn position, so it is tied to the report-writing turn rather than to accumulated error.

End-to-end execution costs parameter grounding rather than tool selection. Table 11 in Appendix I compares the oracle setting against end-to-end execution, where the agent consumes its own tool outputs. Over the 25 configurations run in both settings, mean turn-level hit falls by 4.2 points and completion by 2.2, while parameter accuracy falls by 8.9. The errors that remain under execution are argument-level: the tool layer returned an error for 2.3% of executed calls, and every attributed one was caused by the model rather than by the platform or the data.

5 Deployment Analysis

5.1 General Function-Calling Scores Are Not Sufficient

Table 10 in Appendix H ranks the overlapping models under BFCL and under `STAR-Bench`. Rank under BFCL carries over only loosely. EXAONE-4.0-32B is ninth under BFCL and fourth here, and xLAM-2-70B is second under BFCL and seventh here. Over the ten models whose BFCL runs elicited tool calls at all the rank correlation is moderate and not separable from zero at that sample size (Spearman ρ=0.47, p=0.17), so BFCL neither predicts this benchmark nor is shown to be independent of it. It does not give an ordering a deployment could act on.

5.2 Specialization Helps Unevenly Across AML Subdomains

Figure 4 compares mean tool hit across AML subdomains for the two finance-specialized models, their own base models, and the Korean-specialized and small general models at comparable scale. The two bases are run on the same benchmark and scored by the same code, outside the 28-configuration cohort, because a base model is the reference a fine-tune is measured against. Specialization shifts capability between subdomains without a consistent overall gain, and the two pairs disagree about where. Llama-Open-Finance-8B gains 3.4 points overall on Llama-3.1-8B-Instruct, and gains on three of the four subdomains, while Qwen-Open-Finance-R-8B loses 8.3 overall and loses on all four. Regulatory Reporting is the only subdomain where both decline, and by the largest margins either pair shows: 17.9 points for the Llama pair and 16.0 for the Qwen pair.

Korean specialization is also insufficient by itself: small Korean-specialized models remain weak in Regulatory Reporting while larger ones perform strongly overall, so a specialization label cannot replace a subdomain-level measurement.

5.3 Language Alignment and Reasoning Are Unreliable Remedies

Schema language matters less than the model's own alignment. Table 4 separates the query language from the tool-definition language in a controlled 2×2 over the configurations run on all four cells. The average effects are small. Holding the tool definitions in Korean, English queries move tool hit by at most a point in five of the seven configurations. Translating the tool definitions into English does not cost tool hit on English queries either; five of seven hold or improve, Llama-3.3-70B by 8.6 points. The table shows that the exceptions are specific to particular models and do not favor one language. A.X-4.0 is the configuration that loses most on English queries, by 4.5 points with Korean tool definitions, and Llama-3.3-70B loses 3.9 on that same axis. Llama-3.3-70B recovers once the schema follows the query, and is best of all when both are English. For deployment this argues against a single rule about schema language and for measuring the pairing on the model that will actually be served.

Table 4: Tool hit h by query language $\times$ tool definition language (KR/EN) for the seven configurations run on all four cells, sorted by KR-KR within each group.

Model	KR-KR	EN-KR	KR-EN	EN-EN
Korean-Specialized				
A.X-4.0 (72B)	0.905	0.860	0.893	0.867
EXAONE-4.0-32B	0.890	0.884	0.858	0.872
General-Purpose				
Gemma-4-31B	0.966	0.965	0.963	0.964
Qwen3.5-27B (NT)	0.948	0.940	0.960	0.948
Qwen3.6-27B	0.943	0.934	0.946	0.948
Qwen3.5-27B (T)	0.942	0.949	0.959	0.951
Llama-3.3-70B	0.779	0.740	0.785	0.826

Within our output budget, reasoning mode does not help and often hurts. Four configurations pair a thinking and a non-thinking arm of the same model under an otherwise identical serving entry. Thinking lowers single-turn tool hit in three of the four, by 5.9 points for gpt-oss-20B and 5.6 for gpt-oss-120B, both significant under McNemar's test ($p<10^{-5}$), and by 0.6 for Qwen3.5-27B, which is not; Qwen3.5-4B gains 0.9, also not significant. Averaged over the four the effect is -2.8 points single-turn and -7.5 on multi-turn workflows. The mechanism is visible in the records: reasoning configurations exhaust their output budget far more often, and a turn that runs out of budget produces no call at all. Explicit reasoning does not repair bilingual tool selection or regulatory terminology mapping on this workload.

6 Conclusion and Future Work

We introduced `STAR-Bench`, a compliance-oriented benchmark for AML agent readiness, built from an AML agent platform and expert-verified scenarios, in which every tool executes against the transaction database. Single-turn tool accuracy does not resolve workflow readiness (C3): ten configurations within seven points of one another on single-turn tool hit complete anywhere from a third to two thirds of STR workflows, and the model that completes the most sits in the lower half on single-turn tool hit. Reporting fails in three distinct ways (C2): models never call the validation tool, fall back to a general query in place of CTR-candidate detection, or reach the right reference-lookup tool and then fail to convert Korean regulatory terminology into its argument; a single aggregate hit rate hides all three. General function-calling scores, model size, specialization labels and language alignment give no reliable ordering on either, and within our output budget reasoning mode lowers single-turn and workflow scores on average (C4). `STAR-Bench` diagnoses these weaknesses before deployment rather than certifying compliance; Appendix J states what it measures and what it does not. Future work will extend the benchmark to related regulatory workflows such as KYC, sanctions screening and market surveillance.

AI Use Statement

We used a large language model to aid and polish the writing of this paper and to draft sections of it. Every passage it drafted or edited was reviewed and revised by the authors, who checked each figure quoted in the text against the released artefacts that produce it. The authors take responsibility for the final text and for every claim it makes.

ETHICS STATEMENT

STAR-Bench is an evaluation benchmark, not a compliance-certification or STR-submission system, and nothing in it should be used to decide a real reporting obligation. Its transaction environment is a synthetic table derived from interbank transfer data under the operator's de-identification procedure: every individual and institutional identifier is replaced by a serial number and the data is joined with no external source, so no additional re-identification risk arises; the released table records dates to the day and times to three-hour slots. What we release is that synthetic environment, the tool schemas and their executable code, the queries, gold calls and oracle outputs, the STR scripts, the evaluation code and the run records. The source transaction records stay private, and STR filings are confidential by law: FATF Recommendation 21 requires that institutions be prohibited by law from disclosing that a report is being filed (Financial Action Task Force, 2012), so production case data cannot be released for benchmarking and STAR-Bench is built on expert-authored scenarios instead. Deploying an agent evaluated here would require a separate governance process following the principles for evaluating AI in high-stakes domains (Arnaout et al., 2026). The expert evaluation of generated reports reported in Appendix F was carried out by authors of this paper; no external participants were recruited and no personal data was collected from them.

REPRODUCIBILITY STATEMENT

Every number in this paper is recomputed from released artefacts by one command. The benchmark data, the executable tool layer, the serving registry that pins each configuration's model revision, chat template, tool-call parser, reasoning mode, context window and output budget, the runner, the scorer, the analysis and the two expert raters' scores behind Appendix F are in the released repository, and the figures stated in prose are written to a file by the same command as the tables. Each configuration is served alone from its registry entry: 18 of the 28 on local NVIDIA L40S 48 GB and H100 80 GB hosts, and 10 (gpt-oss-20B and gpt-oss-120B in both reasoning modes, Llama-3.3-70B, Mistral-Small-24B, Ministral-3-3B, Gemma-4-31B, Qwen3.6-27B and Qwen3.6-35B-A3B) through a commercial inference gateway, in every setting. The gateway's provider serves its own copy of the weights, so for those ten the registry fixes the request but cannot pin the model revision. Each run writes a record per case or turn holding the raw response of every round, the parsed calls with their arguments and whether the server or the text fallback produced them, the tool output, and the stop reason; those records are released too, so the scoring can be repeated without serving a model again. Scoring is separate from running and reads only the records and the gold, and each record names the sha256 of the benchmark directory it was run against, so a mismatch is refused rather than scored. Section 4.1 states the decoding and tool-calling settings, Appendix D the case review criteria, and the Ethics Statement what is withheld.

REFERENCES

Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, et al. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-LoRAs. *arXiv preprint arXiv:2503.01743*, 2025.

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

Aelim Ahn, Sooyeon Lee, Hyosun Wang, Chiwan Park, Daeryong Kim, Jihyeon Roh, Kichang Yang, Wonjun Jang, Hwang Woosung, Min Kim, et al. OrchestrationBench: LLM-driven agentic planning and tool use in multi-domain scenarios. In *International Conference on Learning Representations*, volume 2026, pp. 115573–115599, 2026.

Iñaki Aldasoro and Ajit Desai. AI agents for cash management in payment systems. Technical report, Bank of Canada, 2025.

Tamer Alkhouli, Katerina Margatina, James Gung, Raphael Shu, Claudia Zaghi, Monica Sunkara, and Yi Zhang. CONFETTI: Conversational function-calling evaluation through turn-level in-

teractions. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7993–8006, 2025.

Hiba Arnaout, Anmol Goel, H Andrew Schwartz, Steffen T Eberhardt, Dana Atzil-Slonim, Gavin Doherty, Brian Schwartz, Wolfgang Lutz, Tim Althoff, Munmun De Choudhury, et al. Responsible evaluation of AI for mental health. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7625–7660, 2026.

Kyunghoon Bae, Eunbi Choi, Kibong Choi, Stanley Jungkyu Choi, Yemuk Choi, Kyubeen Han, Seokhee Hong, Junwon Hwang, Taewan Hwang, Joonwon Jang, et al. EXAONE 4.0: Unified large language models integrating non-reasoning and reasoning modes. *arXiv preprint arXiv:2507.11407*, 2025.

Yunju Bak, Hojin Lee, Minho Ryu, Jiyeon Ham, Seungjae Jung, Daniel Wontae Nam, Taegyeong Eo, Donghun Lee, Doohae Jung, Boseop Kim, et al. Kanana: Compute-efficient bilingual language models. *arXiv preprint arXiv:2502.18934*, 2025.

Bank Policy Institute. Getting to effectiveness: Report on U.S. financial institution resources devoted to BSA/AML & sanctions compliance. Technical report, BPI, 2020.

Kinjal Basu, Ibrahim Abdelaziz, Kiran Kate, Mayank Agarwal, Maxwell Crouse, Yara Rizk, Kelsey Bradford, Asim Munawar, Sadhana Kumaravel, Saurabh Goyal, et al. NESTful: A benchmark for evaluating LLMs on nested sequences of API calls. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 33526–33535, 2025.

Zehui Chen, Weihua Du, Wenwei Zhang, Kuikun Liu, Jiangning Liu, Miao Zheng, Jingming Zhuo, Songyang Zhang, Dahua Lin, Kai Chen, et al. T-Eval: Evaluating the tool utilization capability of large language models step by step. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9510–9529, 2024.

Chanyeol Choi, Jihoon Kwon, Alejandro Lopez-Lira, Chaewoon Kim, Minjae Kim, Juneha Hwang, Jaeseon Ha, Hojun Choi, Suyeol Yun, Yongjin Kim, et al. FinAgentBench: A benchmark dataset for agentic retrieval in financial question answering. In *Proceedings of the 6th ACM International Conference on AI in Finance*, pp. 632–637, 2025.

Joon Kiat Chua, Donghao Huang, and Zhaoxia Wang. A novel hierarchical multi-agent system for payments using LLMs. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 28–39. Springer, 2026.

Financial Action Task Force. International standards on combating money laundering and the financing of terrorism & proliferation: The FATF recommendations. https://www.fatf-gafi.org/content/dam/fatf-gafi/recommendations/FATF%20Recommendations%202012.pdf.coredownload.inline.pdf, 2012. Updated October 2025.

Financial Action Task Force. Opportunities and challenges of new technologies for AML/CFT. Technical report, FATF, 2021.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Jingguang Han, Yuyun Huang, Sha Liu, and Kieran Towey. Artificial intelligence for anti-money laundering: a review and extension. *Digital Finance*, 2(3):211–239, 2020.

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.

Sanya Juneja, Bill Roberts, Dana Salman, Pavle Avramovic, Alan Ainsworth, and Bryan Zhang. The global state of open banking and open finance report. Technical report, Cambridge Centre for Alternative Finance, 2024.

Dattatray Vishnu Kute, Biswajeet Pradhan, Nagesh Shukla, and Abdullah Alamri. Deep learning and explainable artificial intelligence techniques applied for detecting money laundering–a critical review. *IEEE Access*, 9:82300–82317, 2021.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pp. 611–626, 2023.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. API-Bank: A comprehensive benchmark for tool-augmented LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3102–3116, 2023.

Alexander H Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. AgentBench: Evaluating LLMs as agents. In *International Conference on Learning Representations*, volume 2024, pp. 52989–53046, 2024.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Haoping Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, et al. ToolSandbox: A stateful, conversational, interactive evaluation benchmark for LLM tool use capabilities. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1160–1183, 2025.

Chang Ma, Junlei Zhang, Zhihao Zhu, Cheng Yang, Yujiu Yang, Yaohui Jin, Zhenzhong Lan, Lingpeng Kong, and Junxian He. AgentBoard: An analytical evaluation board of multi-turn LLM agents. *Advances in Neural Information Processing Systems*, 37:74325–74362, 2024.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: a benchmark for general AI assistants. In *International Conference on Learning Representations*, volume 2024, pp. 9025–9049, 2024.

Mistral AI. Mistral Small 3.2. https://huggingface.co/mistralai/Mistral-Small-3.2-24B-Instruct-2506, 2025.

Prathamesh Vasudeo Naik, Naresh Kumar Dintakurthi, Zhanghao Hu, Yue Wang, and Robby Qiu. Co-Investigator AI: The rise of agentic AI for smarter, trustworthy AML compliance narratives. *arXiv preprint arXiv:2509.08380*, 2025.

Shishir G Patil, Tianjun Zhang, Xin Wang, and Joseph E Gonzalez. Gorilla: Large language model connected with massive APIs. *Advances in Neural Information Processing Systems*, 37:126544–126565, 2024.

Shishir G Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E Gonzalez. The Berkeley Function Calling Leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Forty-second International Conference on Machine Learning*, 2025.

Akshara Prabhakar, Zuxin Liu, Ming Zhu, Jianguo Zhang, Tulika Manoj Awalgaonkar, Shiyu Wang, Zhiwei Liu, Haolin Chen, Thai Hoang, Juan Carlos Niebles, et al. APIGen-MT: Agentic pipeline for multi-turn data generation via simulated agent-human interplay. *Advances in Neural Information Processing Systems*, 38, 2025.

PwC. Instant payments: a spotlight on the European Commission for regulation. Technical report, PwC, 2023. Accessed: 2026-04-26.

Yunjia Qi, Hao Peng, Xiaozhi Wang, Amy Xin, Youfeng Liu, Bin Xu, Lei Hou, and Juanzi Li. AgentIF: Benchmarking instruction following of large language models in agentic scenarios. *Advances in Neural Information Processing Systems*, 38, 2025.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. In *International Conference on Learning Representations*, volume 2024, pp. 9695–9717, 2024.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3.6-27B: Flagship-level coding in a 27B dense model, April 2026b. URL https://qwen.ai/blog?id=qwen3.6-27b.

Qwen Team. Qwen3.6-35B-A3B: Agentic coding power, now open to all, April 2026c. URL https://qwen.ai/blog?id=qwen3.6-35b-a3b.

Ella Rabinovich and Ateret Anaby Tavor. On the robustness of agentic function calling. In *Proceedings of the 5th Workshop on Trustworthy NLP (TrustNLP 2025)*, pp. 298–304, 2025.

Ryan Teknium, Jeffrey Quesnelle, and Chen Guang. Hermes 3 technical report. *arXiv preprint arXiv:2408.11857*, 2024.

United Nations Office on Drugs and Crime. Estimating illicit financial flows resulting from drug trafficking and other transnational organized crimes. Technical report, UNODC, 2011.

Mengsong Wu, Tong Zhu, Han Han, Chuanyuan Tan, Xiang Zhang, and Wenliang Chen. Seal-Tools: Self-instruct tool learning dataset for agent tuning and detailed benchmark. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pp. 372–384. Springer, 2024.

Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, et al. FinBen: A holistic financial benchmark for large language models. *Advances in Neural Information Processing Systems*, 37:95716–95743, 2024.

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.

Lucen Zhong, Zhengxiao Du, Xiaohan Zhang, Haiyi Hu, and Jie Tang. ComplexFuncBench: Exploring multi-step and constrained function calling under long-context scenario. *arXiv preprint arXiv:2501.10132*, 2025.

A TOOL SUITE

Table 5 lists the 23 tools of `STAR-Bench` grouped by AML subdomain (Section 3.1): transaction inquiry and statistics (7 tools), suspicious-activity detection (4), money-flow and network analysis (7), and regulatory reporting (5). `generate_str` is evaluated only in the multi-turn STR workflow, so 22 of the 23 carry single-turn cases.

The single-turn cases are spread over those tools rather than concentrated in a few: each of the 22 single-tool categories carries between 25 and 61 cases, and the four regulatory-reporting tools that appear in single-turn form carry 131 between them. Of the 1,258 single-turn cases, counted from each case's own gold, 987 have one gold tool, 92 have two and 20 have three; the remaining 159 have none. They are of two kinds: 119 are out of scope, asking for something no gold call covers, and 40 omit a required argument, so any call made without it is unsupported; 39 of these come from the 44 missing-parameter cases, whose other five omit nothing the tool requires and expect the call. In both, the correct response is a written reply with no call. Fifteen cases also accept an alternative tool: 7 that have a gold tool, and 8 of the out-of-scope ones, where calling the alternative counts as correct alongside such a reply. The difficulty labels are 737 easy, 283 medium and 238 hard. Every case without a gold tool is labeled easy, which follows from the definition in Section 3.4: difficulty is graded on ambiguity among similar tools, and a case with no gold tool has no tool to be ambiguous about. Across the cohort, out-of-scope cases are answered correctly 84.5% of the time, in line with the 82.2% of easy cases that have a gold tool, while cases missing an argument are answered correctly 51.6% of the time, below the medium and hard ones at 78.2% and 77.8%.

Table 5: The tools of STAR-Bench grouped into AML subdomains derived from the reference AML agent platform.

Tool	Description
Transaction Inquiry & Statistics	
get_statistics	Retrieve summary statistics (total, fraud count/ratio, type dist.)
query_transactions	SQL-based raw transaction data query with filters
get_account_profile	Transaction statistics profile for a specific account
compare_periods	Compare fraud statistics between two time periods
get_fraud_type_summary	Transaction counts and ratios by fraud type (six types)
get_institution_report	Transaction volume and fraud statistics for a specific institution
get_receiving_account_profile	Transaction statistics profile for a receiving (deposit) account
Suspicious Activity Detection	
predict_fraud	Boosting-based fraud probability prediction for a transaction
rank_risky_transactions	Rank transactions by predicted fraud probability
score_account_risk	Composite account risk score (0–100) from five behavioral indicators
detect_monitoring_alerts	Rule-based transaction monitoring (6 predefined rules)
Money Flow & Network Analysis	
analyze_network	Transaction network around an account: centrality and key counterparties
detect_aml_patterns	Detect AML-specific patterns (ring, layering, funnel)
detect_smurfing_network	Identify smurfing (fund collection/dispersion) networks
detect_dormant_reactivation	Detect dormant accounts reactivated after long inactivity
analyze_cross_institution_flow	Analyze fund flows across financial institutions
get_trend_analysis	Monthly or quarterly trend of transaction volume and fraud rate
analyze_channel_risk	Fraud risk breakdown by transaction medium (channel)
Regulatory Reporting	
detect_ctr_candidates	Identify CTR-eligible high-value or structuring transactions
lookup_fiu_reference_types	Search the FIU suspicious-transaction reference type catalog
validate_STR_fields	Validate STR draft fields against the official field schema
get_aml_glossary	Look up AML terminology (CDD, STR, CTR, PEP, etc.)
generate_str[†]	Compose a suspicious transaction report draft from prior tool outputs

[†] generate_str is evaluated only in the multi-turn STR workflow (no single-turn cases).

Declining an out-of-scope request is as easy as its label says; holding back a call until a missing argument is supplied is the hardest case type in the single-turn set.

Tools within a subdomain are deliberately close, so the difficulty lies in choosing among them. get_account_profile and get_receiving_account_profile take the same arguments and differ only in which side of a transfer they profile. detect_monitoring_alerts applies the institution's six predefined rules, while detect_aml_patterns searches for typologies such as rings, layering and funnels. predict_fraud scores one transaction, while rank_risky_transactions orders many by that same score. detect_ctr_candidates reads the same transaction fields as query_transactions but returns a reporting decision rather than records. A model that matches on the tool name without reading the description reaches for the wrong member of each pair, and Section 4.2 reports how often that happens.

B Benchmark Examples

Table 6 shows a single-turn Hard case, and Table 7 a multi-turn STR scenario (Section 3.4).

The single-turn case is Hard for three reasons at once, and each is scored by a different metric. Its three tools are all reachable from the same institution, so a model that treats get_institution_report and query_transactions as interchangeable calls one and skips the other, which r records. The third call's arguments are given in the query as ordinary words and have to be mapped onto the schema's coded fields, so "salary" becomes fund_type 1,

Table 6: Example of a single-turn Hard case: a natural-language query with its ground-truth tool call(s).

Category multi-tool **Difficulty** Hard

USER QUERY

"Check the report for financial institution 158, query that institution's suspicious transactions with SQL, and predict the fraud risk score of a KRW 20M salary transaction (fund type 1) with time slot 0, sending bank 158, receiving bank 134, over PC banking (medium type 1)."

EXPECTED TOOL CALLS (Ordered)

❶ `get_institution_report`(`bank_id`: 158)

❷ `query_transactions`(SQL on `sender_bank` = 158, `is_fraud` = 1)

❸ `predict_fraud`(`time_slot`: 0, `sender_bank`: 158, `receiver_bank`: 134, `fund_type`: 1, `media_type`: 1, `amount`: 20,000,000)

METRICS tool selection (h, r, p), parameter acc. (a), call order (o)

Table 7: Example of a multi-turn STR scenario: per-turn user messages with the ground-truth tool call at each turn.

Category missing-parameter **Fraud Type** concurrent multi-transaction requests

Turn 1 *"Can you help me draft an STR?"*
→ Clarification (no tool call); key info missing (account, period, suspicion type)

Turn 2 *"The sending account is 9000000004236284 and I suspect concurrent multi-transaction requests. Query its 2024 suspicious transactions, top 20 by amount."*
→ `query_transactions`(`sender_acc`=9000000004236284, `is_fraud`=1, `date` in 2024)

Turn 3 *"Predict the risk score of the first transaction."*
→ `predict_fraud`(`time_slot`: 9, `sender_bank`: 147, `receiver_bank`: 157, `fund_type`: 0, `media_type`: 2, `amount`: 100,000,000)
Parameters from a representative row of Turn 2 results.

Turn 4 *"Also check whether 2024 transactions over KRW 10M are CTR-eligible."*
→ `detect_ctr_candidates`(`mode`: high_value, `date_from`: 20240101, `date_to`: 20241231)

Turn 5 *"Draft an STR for concurrent multi-transaction requests."*
→ `generate_str`(`summary`: "Account 9000000004236284 …", `fraud_type`: concurrent multi-transaction requests, `tools_used`: [`query_transactions`, `predict_fraud`, `detect_ctr_candidates`])
Synthesized from Turn 2–4 cumulative context into the STR narrative section structure.

"PC banking" becomes `media_type` 1 and "KRW 20M" becomes 20,000,000, which a records. And the report has to precede the query it scopes, so this is one of the cases where o is defined at all.

The multi-turn scenario shows what a workflow adds to that. Turn 1 withholds the account, the period and the suspicion type, so the correct action is to ask rather than to call anything, and any call scores zero. Turn 2 supplies all three and the gold call is a query carrying the account, the fraud flag and the year it asks about. Turn 3 asks for a fraud probability whose six arguments appear nowhere in the user's message: they come from a row of what Turn 2 returned, so the turn is unanswerable without carrying that result forward, which is the property the long-context category generalises and the oracle setting supplies on the model's behalf. Turn 4 keeps the period but drops the account and moves from analysis to reporting, and answering it with another transaction query is the substitution Section 4.3 measures. Turn 5 asks for the report, and its gold call names the three tools whose outputs the narrative has to rest on, so a draft that reaches this turn can still fail on grounding rather than on tool selection.

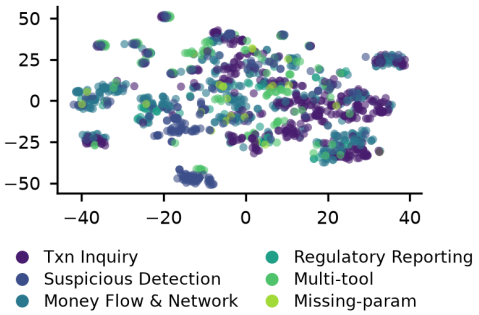


Figure 5: t-SNE projection of question embeddings for the single-turn queries.

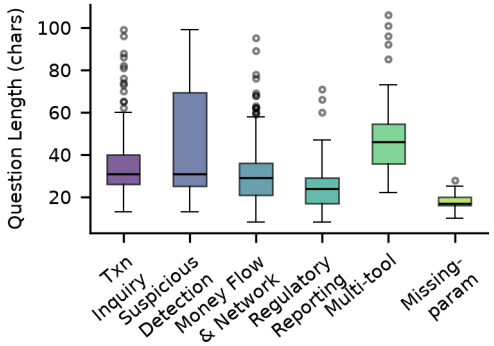


Figure 6: Question-length distribution by subdomain.

C Benchmark Question Diversity

Figure 5 projects the single-turn questions into two dimensions with t-SNE over character n-gram TF-IDF features. The questions spread broadly through the semantic space, and subdomains form partial clusters that overlap substantially, which qualitatively supports how hard it is to discriminate between functionally similar tools (RQ1). Figure 6 shows the question length distribution per subdomain.

Length separates the case types more sharply than it separates the subdomains. The median query runs 35 characters in transaction inquiry, 37 in suspicious-activity detection, 32 in money-flow and network analysis and 41 in regulatory reporting, so no subdomain is systematically wordier than another. The two ends of the plot are case types rather than subdomains: multi-tool queries have a median of 54 characters because they name two operations in one sentence, and missing-parameter queries a median of 26.5 because they leave out details a fuller query would carry. A model therefore cannot use length as a proxy for which tool a query wants, or for whether it should call one at all.

The multi-turn half is composed along a different axis. Its 50 scenarios divide into 14 base, 14 missing-parameter and 22 long-context cases; they run four turns (32 scenarios), five turns (17) or six turns (1), for 219 turns in total, and they cover the six suspicious-activity types listed in Section 3.4. Long-context is the largest group because reusing an earlier turn's result is a property the oracle setting cannot supply on the model's behalf: the result is injected, but deciding to carry it forward is still the model's to get right.

D Case Review Criteria

The second expert's defect-finding pass checks three criteria, and a case that fails one is revised until author and verifier agree. Tool-role uniqueness: the gold tool is the only tool that satisfies the user intent under the provided schema. Parameter observability: every required argument is explicitly recoverable from the query or from prior workflow context, or is intentionally absent in a clarification or abstention case. Difficulty consistency: the assigned difficulty matches the case's actual tool ambiguity, multi-tool structure and information completeness. Cases with several plausible tools, unsupported parameters, underspecified gold calls or inconsistent labels are the ones this pass removes.

E STR Narrative Coverage

Table 8 details the STR narrative coverage summarized in Section 3.4. For each of the 45 scenarios whose gold ends in a `generate_str` call, we measure by regular expression whether the ground-truth `summary` satisfies the keywords of seven sections of the Korean STR form; the

Table 8: Narrative section coverage of the ground-truth STR draft summaries (n=45 scenarios whose gold ends in an STR). Coverage is the share of summaries that satisfy the section's keywords; the overall-assessment row scores the six 5W1H items and so takes fractional values.

Section	Coverage (%)	Mean score
Transacting parties	97.8	0.978
Date and time	88.9	0.889
Institution and branch	86.7	0.867
Instrument and channel	100.0	1.000
Transaction method	100.0	1.000
Grounds for suspicion	100.0	1.000
Overall assessment (6W)	100.0	0.804
Average		**0.934**

overall-assessment row scores the six 5W1H items separately and averages them, so it alone takes fractional values at full coverage.

Every section is satisfied in at least 86.7% of the drafts and three in all of them, so the gold summaries carry the mandatory STR narrative fields consistently. The two lowest, date and time at 88.9% and institution and branch at 86.7%, are also the two the released environment records most coarsely: transaction times fall in three-hour slots, and financial institutions appear only as serial numbers.

A section counts as satisfied when the draft matches any pattern in its family, and the families are released with the scorer rather than described only here. Transacting parties looks for an account serial number or for the words that name a party as a business, a corporation or an individual. Date and time looks for a year, an eight-digit date, a quarter, a month or a half-year, and for the time of day, which the environment keeps to three-hour slots, including late-night, dawn, morning, afternoon, same-day and next-day. Institution and branch looks for a withdrawing or depositing institution, an institution serial number, or a branch or office. Instrument and channel looks for a medium, cash, cheque, securities, foreign exchange, transfer or electronic money, for the fund-type and medium-type code names, and for the access channel, including internet and PC banking, ATM, mobile and call centre. Transaction method looks for the behavior that makes the activity reportable: late-night, consecutive, repeated, multiple or simultaneous transfers, splitting, structuring or smurfing, a large or sudden change in pattern, and chained accounts. Grounds for suspicion looks for the evidence behind the judgement: a model or predicted probability, an anomaly or risk score, a network, linked accounts or a cluster, and a percentage. The overall assessment is the mean of six items, one per 5W1H question, which is why it scores 0.804 at full coverage while every other satisfied section scores 1 or close to it.

F Expert Agreement with the STR Checker

Section 4.3 scores a generated report by keyword and entity matching. Whether those numbers track what a compliance practitioner sees in the same report is a separate question, and this section reports what two of them found when they read the reports themselves.

The sample is 45 generated reports: three serving configurations spanning the checker's range at the draft level, its high, middle and low, each paired with the same 15 scenarios, drawn stratified by category from the 35 in which all three configurations wrote a report. Items carry codes rather than model names, their order differs between raters, and the raters worked independently of each other. Each rater scored, from 1 to 5, the checker's three dimensions, required fields, evidence match and regulatory register, and an overall judgement. Table 9 gives the rank agreement.

The checker's overall score tracks both raters closely, at ρ=0.865 and 0.827, and the raters agree with each other more closely still, at 0.913, which is the ordering a usable proxy should produce. The entity extractor finds no figure to trace in 13 of the 45 reports, and Section 3.5 scores those 0 rather than setting them aside, on the grounds that a report citing no figure grounds nothing. The comparison here follows that convention, and the raters support it: both scored all 13 at 1 overall. Dropping those reports instead leaves evidence-match agreement at +0.417 for rater A and

Table 9: Rank agreement (Spearman's ρ) between the automatic STR checker and two expert raters over the same 45 generated reports, and between the raters. Every coefficient is significant at $p < 0.0001$.

	Rater A	**Rater B**	**A vs B**
Required fields	+0.771	+0.750	+0.811
Evidence match	+0.728	+0.704	+0.865
Regulatory register	+0.623	+0.704	+0.793
Overall	+0.865	+0.827	+0.913

$+0.326$ for rater B over the 32 that remain, so much of that dimension's agreement comes from the convention.

One property of the exercise bounds what it establishes: regulatory register sits at the bottom of the human scale, scored 1 on 31 of 45 reports by one rater and 28 by the other, so its agreement rests on a narrow band. The table shows that the checker orders reports much as these readers do; it does not show that the checker could replace them.

G JURISDICTIONAL SCOPE

`STAR-Bench` scores three behaviors: separating analytical tools from reporting tools, carrying an investigation through to filing, and retaining evidence across turns. Each corresponds to a FATF standard, not to a national rule (Financial Action Task Force, 2012). Examining a situation to see whether it is suspicious is required of the institution by the Interpretive Note to Recommendation 10 (¶20), under which financial institutions examine the background and purpose of all complex, unusual large transactions and all unusual patterns of transactions that have no apparent economic or lawful purpose, and increase monitoring in order to determine whether those transactions appear unusual or suspicious. Reporting what the examination finds is Recommendation 20, which requires reporting once suspicion arises, whatever the amount (Interpretive Note to Recommendation 20), and the separation the benchmark scores between analytical and reporting tools is the separation those two provisions already draw. Grounding a report in what the tools returned corresponds to Recommendation 11, under which records must permit reconstruction of individual transactions so as to provide evidence, and must include the results of any analysis undertaken. FATF frames large-cash reporting the same way: the Interpretive Note to Recommendation 29 asks countries to consider a system under which institutions report all currency transactions above a fixed amount (§H, ¶14), and lists cash transaction reports among the information an FIU receives under national legislation (§B, ¶2). Every jurisdiction supplies its own parameters for these obligations: a form schema, a reference-code set, a reporting threshold, and a language. `STAR-Bench` instantiates one such set, and the released environment keeps part of it separable: the reporting threshold is an argument of the CTR tool carrying a default rather than a constant in its body, and the required-field schema and the reference catalog are named tables that one module holds and the tools read. Porting would still require code changes. The STR generator builds its output against the same field names, the tool descriptions restate them in prose, and the narrative checker of Appendix E matches Korean keywords, so a substituted schema has to be followed through all three. This section identifies which parameters the artifact separates; carrying out a port and testing whether the tasks transfer is left to future work.

The deployment constraint in Section 4.1 splits the same way. Recommendation 21 requires that institutions be prohibited by law from disclosing that an STR or related information is being filed with the FIU (Financial Action Task Force, 2012), so the material an AML agent handles is subject to a disclosure prohibition wherever the agent runs, and keeping the agent inside the institution is the conservative reading of that obligation in any jurisdiction. The national element is the form the constraint takes here: a network-separation rule that settles where the model may run instead of leaving it to each institution's own assessment.

The three reporting failures in Section 4.3 sit at different levels. Failure to call the STR validation tool is a missed reporting step in any jurisdiction. Mis-mapping an FIU reference type is specific to

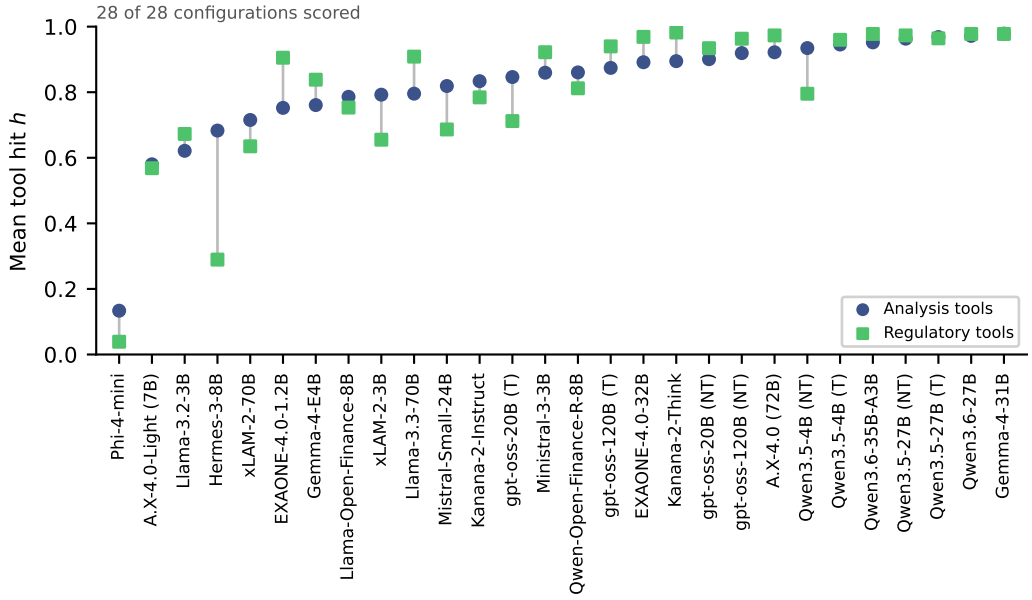


Figure 7: Per-model mean tool hit h for the Regulatory Reporting subdomain vs. all other tools. Each vertical segment is one model's gap.

the national code set. Calling a general query tool instead of the CTR detector is a behavioral failure everywhere, while the reporting regime it belongs to is national.

H Additional Results

This section collects results referenced from Sections 4 and 5: the per-configuration Regulatory Reporting gap (Figure 7), BFCL accuracy against `STAR-Bench` tool hit (Table 10), and what separates the STR turn from the other turns at the same position.

Phi-4-mini's BFCL score comes from prompt mode (Table 10), while here it is served through its native tool-calling interface. In 599 of its 1,258 single-turn cases (48%) it writes a complete call as plain JSON text, which that interface does not parse, so its rank shift reflects the output format as much as tool choice.

At the fourth turn (Section 4.4), over the 1,330 records that report token usage, the median prompt is 9,941 tokens for STR turns and 9,767 for the others. After adjusting by regression for configuration and prompt length, STR turns still score 28.9 points lower in hit, and each further thousand prompt tokens moves hit by 0.012. The STR turns differ in the step itself: the gold STR call takes three arguments where the other gold calls take a median of one, and a model answering it writes a median of 757 completion tokens against 65. These are what writing the report consists of, and the benchmark does not separate them from it.

The overall STR score is not sensitive to how its dimensions are combined. Under alternative definitions, namely counting grounding twice, field completeness with grounding only, and grounding alone, the ordering of the 25 configurations that write a report agrees with the reported one at Spearman ρ between 0.97 and 0.99, and the same three configurations lead under the reported definition and under the one that counts grounding twice. Which of them ranks first is not robust: every alternative puts Qwen-Open-Finance-R-8B ahead of Kanana-2-Instruct, so the claim rests on the ordering, not on the single top place.

Table 10: The same ten models ranked under BFCL and under `STAR-Bench`. The two scores measure different quantities, so only the ranks compare across the benchmarks. Rank Shift is the move from the BFCL rank to the `STAR-Bench` rank, green up for higher and red down for lower.

Model	Under BFCL		Under `STAR-Bench`		Rank Shift
	Acc (%)	Rank	h	Rank	
Qwen3.5-27B (NT)	48.12	1	0.948	1	0
xLAM-2-70B‡	48.06	2	0.695	7	↓ 5
xLAM-2-3B‡	40.88	3	0.739	6	↓ 3
Qwen3.5-4B (NT)	30.71	4	0.905	2	↑ 2
Llama-3.3-70B‡	28.80	5	0.779	5	0
A.X-4.0 (72B)	25.07	6	0.905	3	↑ 3
Phi-4-mini	22.43	7	0.120	10	↓ 3
Llama-3.2-3B‡	21.34	8	0.603	9	↓ 1
EXAONE-4.0-32B	21.29	9	0.890	4	↑ 5
Hermes-3-8B	21.08	10	0.624	8	↑ 2

‡ function-calling mode; all others prompt mode.

Table 11: Oracle versus end-to-end (E2E) evaluation over the 50 multi-turn STR scenarios, sorted by oracle $\bar{h}$. 3 configurations have no end-to-end column because their context window is under the setting's: Kanana-2-Think (Bak et al., 2025) (32768 tokens), Qwen-Open-Finance-R-8B (40960 tokens), Kanana-2-Instruct (Bak et al., 2025) (32768 tokens).

Configuration	Oracle $\bar{h}$	E2E $\bar{h}$	$\Delta\bar{h}$	Oracle c	E2E c
Kanana-2-Think (Bak et al., 2025)	0.922	–	–	0.680	–
Mistral-Small-24B (Mistral AI, 2025)	0.922	0.890	−0.032	0.720	0.640
A.X-4.0 (72B)	0.909	0.863	−0.046	0.660	0.480
Gemma-4-31B	0.904	0.895	−0.009	0.660	0.620
Llama-3.3-70B (Grattafiori et al., 2024)	0.900	0.890	−0.009	0.600	0.580
gpt-oss-20B (NT) (Agarwal et al., 2025)	0.890	0.890	0.000	0.580	0.600
Qwen3.5-27B (NT) (Qwen Team, 2026a)	0.890	0.868	−0.023	0.560	0.540
Qwen-Open-Finance-R-8B	0.881	–	–	0.580	–
gpt-oss-120B (NT) (Agarwal et al., 2025)	0.872	0.895	0.023	0.500	0.560
EXAONE-4.0-32B (Bae et al., 2025)	0.849	0.767	−0.082	0.460	0.320
Qwen3.5-4B (NT) (Qwen Team, 2026a)	0.849	0.772	−0.078	0.480	0.380
xLAM-2-70B (Prabhakar et al., 2025)	0.849	0.790	−0.059	0.460	0.480
Gemma-4-E4B	0.840	0.781	−0.059	0.400	0.420
Kanana-2-Instruct (Bak et al., 2025)	0.840	–	–	0.480	–
Qwen3.6-27B (Qwen Team, 2026b)	0.840	0.840	0.000	0.400	0.460
Qwen3.5-4B (T) (Qwen Team, 2026a)	0.826	0.744	−0.082	0.380	0.280
Qwen3.5-27B (T) (Qwen Team, 2026a)	0.822	0.808	−0.014	0.360	0.360
gpt-oss-120B (T) (Agarwal et al., 2025)	0.813	0.776	−0.037	0.380	0.300
Qwen3.6-35B-A3B (Qwen Team, 2026c)	0.813	0.817	0.005	0.360	0.460
gpt-oss-20B (T) (Agarwal et al., 2025)	0.740	0.735	−0.005	0.320	0.260
Llama-Open-Finance-8B	0.703	0.726	0.023	0.220	0.220
EXAONE-4.0-1.2B (Bae et al., 2025)	0.689	0.539	−0.151	0.080	0.040
Ministral-3-3B (Liu et al., 2026)	0.685	0.626	−0.059	0.180	0.140
Llama-3.2-3B (Grattafiori et al., 2024)	0.594	0.694	0.100	0.080	0.160
xLAM-2-3B (Prabhakar et al., 2025)	0.571	0.571	0.000	0.140	0.120
Hermes-3-8B (Teknium et al., 2024)	0.507	0.374	−0.132	0.040	0.060
Phi-4-mini (Abouelenin et al., 2025)	0.361	0.073	−0.288	0.000	0.000
A.X-4.0-Light (7B)	0.352	0.324	−0.027	0.000	0.000
Avg. (25 configs)	0.760	0.718	−0.042	0.361	0.339

I End-to-End Execution: All Configurations

Table 11 reports every configuration in the cohort under the oracle setting, and the 25 whose context window fits it under end-to-end execution, over the 50 multi-turn STR scenarios. $\Delta\bar{h}$ is the end-to-end minus oracle difference in mean turn-level hit.

Two observations do not survive aggregation. First, the cost of end-to-end execution is not uniform: $\Delta\bar{h}$ runs from -0.288 for Phi-4-mini to $+0.100$ for Llama-3.2-3B, with nine configurations losing more than five points, seven moving by less than one, and four improving. The largest gain belongs to Llama-3.2-3B, 24th of 28 under oracle, where a handful of recovered scenarios moves the mean by several points, so an improvement low in the ranking is not evidence that self-generated context helps. Second, $\bar{h}$ spans 0.352–0.922 under oracle and 0.073–0.895 under end-to-end, a wider range than the -0.042 cohort average suggests; the average is a statement about the cohort and not about any configuration in it.

J Evaluation Scope

`STAR-Bench` measures AML agent readiness: an agent's ability to select appropriate AML tools under overlapping regulatory roles, ground their parameters, retain investigation context across turns, and support regulation-related reporting workflows. Investigative work outside the tool surface stays outside this scope, including integration with heterogeneous back-office systems, manual review, retrieval of external documents, and analyst discretion over how far to pursue a lead.

The tool suite is derived from an AML agent platform designed with practitioners who carry out this work, and every case was authored and cross-reviewed by AML practitioners. STR filings and the information around them are confidential by law: FATF Recommendation 21 requires that institutions be prohibited from disclosing that a report is being filed (Financial Action Task Force, 2012). Production case data therefore cannot be released for benchmarking, and `STAR-Bench` is built instead on a de-identified synthetic transaction environment and expert-authored scenarios. The platform is a reference application for this work rather than a deployed production system.

Four properties of the current design bound what the benchmark observes. Every STR turn asks for the report, so the benchmark scores whether an agent carries out a reporting step it is given; deciding unprompted whether a case is reportable, with both missed and unneeded reports scored, is left to a later release. The tool surface is fixed at 23 tools, so behavior under an expanding toolset lies outside it. Oracle turns inject correct tool outputs, so robustness to erroneous or noisy tool results is not measured. Models are evaluated as served, and the benchmark includes no training pipeline.

Two further bounds concern what a score means rather than what it covers. The benchmark has no human baseline: every number here is a comparison among the configurations evaluated, and how an analyst working the same cases would score is unmeasured, so the scores have no human reference point. The STR narrative is also scored by keyword and entity matching rather than by a reader, which makes the score deterministic and reproducible at the cost of registering only what the patterns cover, and a matched figure is not checked against the account or transaction it is attached to; Appendix E lists them and Appendix F measures how far they agree with two readers.