Evaluating Deep Research Agents on Expert Consulting Work: A Benchmark with Verifiers, Rubrics, and Cognitive Traps

Anonymous ACL submission

Abstract

Frontier deep research agents (DRAs) are being deployed in enterprise workflows faster than they are being evaluated. Existing benchmarks measure factual recall, single-hop QA, or generic agentic skill, and miss the multi-document, decision-grade deliverables DRAs are asked to produce. We introduce a benchmark of 70 SME-authored management consulting prompts, each embedding cognitive traps that penalize surface-pattern reasoning. Three frontier agents, namely Claude Opus 4.6, OpenAI o3-deep-research and Gemini 3.1 Pro deep-research, are scored on two complementary layers: deterministic binary verifiers (mean 14.9 per task) and a five-criterion 0–3 SME rubric (Data Integrity, Analytical Rigor, Relevance & Focus, Execution Precision, Format & Deliverability), combined into a Verifier-Rubric Score (VRS, 0–100).

Acceptance under a joint threshold (rubric mean ≥ 2.5 and verifier pass rate $\geq 80\%$) is uniformly low: Claude 15.7%, o3 12.9%, Gemini 12.9%. Pairwise differences are statistically indistinguishable. On the continuous VRS, Claude leads (61.4 [CI: 55.2, 67.5]), followed by Gemini (52.6) and o3 (38.5); the Claude–o3 gap (Δ=22.9, p<0.001) survives Bonferroni correction. No agent averages above the rubric's "adequate" threshold of 2.0; no agent's mean verifier pass rate reaches the 80% acceptance floor. Each agent fails distinctively: o3 leads on data fabrication and file-access failures; Claude propagates cascading computation errors; Gemini oscillates between the highest perfect-verifier rate and the most catastrophic collapses. The benchmark, evaluation code, and full prompt corpus are publicly released.

1 Introduction

The companies that sell deep research agents have moved faster than the people evaluating them. DRAs are already being wired into enterprise pipelines where the answers feed multi-million-dollar decisions. Most of the benchmarks used to vet these systems were not built for that kind of use. The dominant ones measure factual recall (MMLU (Hendrycks et al., 2021a)), single-hop question answering (TriviaQA (Joshi et al., 2017)), web navigation (WebArena (Zhou et al., 2024)), or generic agentic skill (GAIA (Mialon et al., 2024), AgentBench (Liu et al., 2024)). The recent wave of professional-domain benchmarks like FinanceBench (Islam et al., 2023) for finance, LegalBench (Guha et al., 2024) for law, MedQA (Jin et al., 2021) for medicine, is a step in the right direction. However, these still frame evaluation as question-answering rather than the production of decision-grade structured deliverables. The methodological literature on agentic evaluation (Xi et al., 2023; Gu et al., 2024; Liang et al., 2023; Srivastava et al., 2023) has flagged exactly this gap.

The cost of the gap is concrete. One task in our corpus involving an inbound-freight cost program for a packaging firm would commit the company to roughly €4.5 billion of capital expenditure on a defective basis, if a single Year-5 segment revenue figure is miscalculated. Modern deep research agents have measurable rates of exactly this kind of miscalculation, and well-documented tendencies to confabulate when the source material is silent (Ji et al., 2023; Huang et al., 2023; Kadavath et al., 2022).

Our benchmark comprises 70 tasks tested over 3 agents: Claude Opus 4.6 with web search, OpenAI o3-deep-research, and Google Gemini 3.1 Pro deep-research. It is built around three design choices that distinguish it from prior research-agent benchmarks:

- The scoring is two-layered: every response is checked by a suite of binary task-specific verifiers and then independently scored by a subject-matter expert on a five-criterion 0-3

rubric (Data Integrity, Analytical Rigor, Relevance & Focus, Execution Precision, Format & Deliverability), with the two layers combined into a Verifier-Rubric Score (VRS) on 0-100. The dual layer exposes agent-distinct failures that single-metric benchmarks systematically miss (Section 4.5).

- The prompt corpus is organized not by topic but by cognitive capability: five classes targeting Constrained Research Prompts (CRP), Relevance Compression Prompts (RCP), Structural Compliance Prompts (SCP), Latent Decomposition Prompts (LDP), and Failure-Sensitive Prompts (FSP), each isolating a kind of reasoning we wanted to test independently (Sections 4.2–A.1).

- Many of the prompts deliberately embed cognitive traps in the form of realistic human-style errors in the source documents (inconsistent units, footnote-body contradictions, non-standard date formats) and deterministic precision failures (averaging where weighting is required, defaulting to plausible numbers in the absence of an explicit value). These keep the benchmark hard for shallow heuristics; a benchmark with only clean inputs would systematically understate the difficulty of professional research work.

The framework of our benchmark captures distinct failure modes besides just providing a comparison of success rates. At n=70, Claude leads on continuous VRS (61.4), reasoning mean (1.89), and fewest criterion-zeros (35), yet ties the other agents on the binary ACCEPT threshold at ∼13–16%. o3 has the highest auto-reject rate (41.4%) driven by 22 Data Integrity zeros and 62 total criterion-zeros. Gemini accumulates the most criterion-zeros (74) but wins the most per-prompt VRS argmax matches (26.3 of 70) and leads on two prompt types. The continuous and binary views diverge: continuous VRS distinguishes agents, but the stringent ACCEPT threshold ($\bar{r} \geq 2.5$ and $V \geq 80\%$) compresses them into statistical indistinguishability. These per-agent signatures, the prompt-class-conditional effect sizes, and the criterion correlation structure together support this benchmark as a discriminating evaluation framework for frontier deep research agents.

Section 2 situates the benchmark against related work. Section 3 describes the prompt taxonomy, cognitive-trap design, dual-layer scoring framework, and multi-agent infrastructure. Section 4 reports headline results, per-type performance, failure signatures, and cross-benchmark validation. Section 5 discusses caveats; Section 6 concludes. Extended diagnostics, worked examples, and the full QC protocol are in the appendices.

Reproducibility: All evaluation code can be found here.

Code: https://anonymous.4open.science/status/dra-response-gen-4288.

Dataset (70-prompt corpus and grading workbook) will be released publicly on paper publication.

2 Related Work

Research agent and domain-specific benchmarks. General-purpose agentic benchmarks (GAIA (Mialon et al., 2024), BrowseComp (OpenAI, 2025), WebArena (Zhou et al., 2024), AgentBench (Liu et al., 2024), ToolBench (Qin et al., 2024), ToolEval (Patil et al., 2023)) target tool use, browsing, and multi-step reasoning but not domain-expert research. Deep-research benchmarks (DeepResearch Bench (Du et al., 2025), ResearcherBench (Xu et al., 2025), SciAgent (Ma et al., 2024), SciBench (Wang et al., 2024b)) target scientific QA but do not enforce corpus discipline or test business deliverable production. Domain-specific benchmarks (Liang et al., 2023; Srivastava et al., 2023) establish the methodology of expert-graded evaluation in finance (FinanceBench (Islam et al., 2023), FinQA (Chen et al., 2021b), TAT-QA (Zhu et al., 2021)), economically valuable knowledge work (APEX-v1 (Vidgen et al., 2025), APEX-Agents (Vidgen et al., 2026), ProfBench (Wang et al., 2025), GDPval (Patwardhan et al., 2025)), medicine (MedQA (Jin et al., 2021), PubMedQA (Jin et al., 2019)), and law (LegalBench (Guha et al., 2024), CaseHOLD (Zheng et al., 2021)). Of these, the closest in methodological setup are APEX-v1 (LLM-judged expert binary rubric on knowledge-work deliverables) and APEX-Agents (LLM-judged binary criteria over environment-state snapshots in multi-application simulations). ResearchRubrics (Sharma et al., 2025) introduces an expert-rubric framework similar in spirit. None of these benchmarks combines deterministic SME-graded verifiers with a separate multi-criterion ordinal rubric on the same response,

nor embeds cognitive traps as a design choice; Table 1 summarizes the key methodological distinctions.

LLM-as-Judge and code generation. The LLM-as-Judge paradigm (Zheng et al., 2023; Li et al., 2023b) has known calibration failures including length bias, sycophancy, and self-preference (Wang et al., 2024a; Gu et al., 2024; Liu et al., 2023b; Dubois et al., 2024; Panickssery et al., 2024; Koo et al., 2023); we therefore anchor on SME annotation and use binary verifiers as in HELM-style hybrid evaluation (Liang et al., 2023). Code-generation benchmarks (HumanEval (Chen et al., 2021a), MBPP (Austin et al., 2021), SWE-bench (Jimenez et al., 2024; OpenAI, 2024), CodeXGLUE (Lu et al., 2021), API-Bank (Li et al., 2023a)) and tool-use leaderboards (Patil et al., 2023) evaluate code or API selection in isolation, not the end-to-end research-to-deliverable pipeline. The library-specific failures we observe (Section 4.5) connect to training-data coverage gaps for niche packages (Zhang et al., 2023a; Liu et al., 2023a).

Hallucination, calibration, and statistics. Hallucination in long-form generation is well-studied (Ji et al., 2023; Huang et al., 2023; Zhang et al., 2023b; Li et al., 2024; Manakul et al., 2023), mostly in QA settings; faithfulness in summarization (Maynez et al., 2020; Kryściński et al., 2020) is methodologically related. Our two-layer instrument allows fabrication and structural completion failures to be observed in the same response, extending this literature into the structured-deliverable setting where polished formatting can mask fabricated content. Our statistical apparatus (paired McNemar tests on agent comparisons (McNemar, 1947; Dietterich, 1998) and multiple-testing correction (Holm, 1979; Demšar, 2006; Bouthillier et al., 2021)) follows established benchmark-comparison methodology.

3 Benchmark Design

3.1 Task Design and Taxonomy

The benchmark's tasks are structured around five **Prompt Types** that capture distinct deep-research capabilities, each designed to test a specific failure mode that surface-level reasoning would not catch. The prompt types description is presented in Table 2.

Concrete worked out examples for each prompt class, including the input materials, expected output structure, and the embedded cognitive traps, are provided in Appendix L.

3.2 Task Corpus and Cognitive Traps

Our evaluation comprises 70 unique SME-authored prompts in the Management Consulting (MC) domain, distributed as: RCP 18, SCP 18, CRP 13, LDP 13, FSP 8.

Each prompt is accompanied by 2–4 proprietary input files in mixed formats (CSV, XLSX, PDF, DOCX; 2KB–430KB). A subset requires the agent to produce structured output files via code generation. Prompts carry a corpus-discipline annotation (closed, hybrid, or open); enforcement is intrinsically limited because the agents are partial black boxes, so corpus adherence is measured rather than guaranteed.

Cognitive traps. A distinguishing feature of our prompt design is the deliberate embedding of *cognitive traps*. These serve two purposes: (i) *Human-error mimicry*: Input documents contain realistic mistakes (misnamed product line, inconsistent units between tables, footnote contradicting body, non-standard date formats) that penalise agents that copy surface text without reconciling against context. (ii) *Deterministic precision traps*: At least one numerical step is constructed so a shallow agent (taking the first plausible number, averaging without weighting, applying a default assumption) yields a confidently wrong answer, while the correct path requires reading a footnote or applying an explicit qualifier. A benchmark with only clean inputs understates professional research difficulty, where small source-material ambiguities are the norm.

All prompt packages, including the embedded cognitive traps, verifier specifications, and authorized-source designations, were independently vetted before annotation by a Principal Expert with 15+ years of management consulting sector experience to ensure realism and difficulty calibration. Worked examples illustrating each of the five prompt classes are provided in Appendix L.

The 70-prompt corpus, including verifier specifications, authorized-source lists, and embedded cognitive-trap definitions will be released publicly on paper publication.

3.3 Evaluation Framework

Rubric development. The rubric reported here is the result of a two-phase development process. In the first phase, the authors drafted a hierarchical in-

Table 1: Methodological comparison with closest prior benchmarks. ✓ = uses feature; ✓[a,b,c] = partial use (see footnotes).

Benchmark	Primary grading instrument	Det. verifiers	Multi-criterion rubric	SME-graded	Cognitive traps
GDPval	Human pairwise comparison vs. expert deliverable	—	✓[a]	✓	—
APEX-v1	LLM-judge over expert binary rubric	—	✓	—	—
APEX-Agents	LLM-judge over env. snapshot, binary criteria	✓[b]	✓	—	—
ProfBench	LLM-judge over expert rubric, bias-mitigated	—	✓	—	—
ResearchRubrics	LLM-judge over expert rubric	—	✓	—	—
DeepResearch Bench	RACE (4-axis LLM-judge) + FACT (citation accuracy)	✓[c]	✓	—	—
DRBench	LLM-judge across 4 axes (GPT-4o backbone)	—	✓	—	—
Ours	Deterministic verifiers + SME 0–3 ordinal rubric	✓	✓	✓	✓

[a] GDPval task writers created detailed scoring rubrics that guide pairwise comparison; the rubric supplements rather than replaces the primary blinded pairwise instrument.
[b] APEX-Agents uses environment-state snapshots (deterministic) alongside LLM-judged binary criteria for grading.
[c] DeepResearch Bench's FACT framework checks citation factuality against retrieved sources deterministically; the per-criterion content judgment uses an LLM judge.

strument comprising seven universal meta-criteria together with prompt-type-specific additional criteria, augmented with a set of binary task-specific verifiers. In the second phase, the draft instrument was reviewed by a Principal Domain Expert with 15+ years of experience in the management consulting sector, with two explicit goals: (i) ensuring the meta-criteria were mutually exclusive and collectively exhaustive (MECE) over the dimensions of management-consulting deliverable quality, and (ii) reducing SME cognitive load during annotation by avoiding redundant or overlapping judgments. The expert review collapsed the two-level structure into the final flat five-criterion rubric (DI, AR, RF, EP, FD; Table 3) and reorganized prompt-type-specific signals into the binary verifier layer rather than the rubric layer. The same expert vetted the prompt corpus, cognitive-trap embeddings, and verifier specifications before annotation began, as noted in Section 3.

Our benchmark uses a dual-layer scoring scheme: task-specific binary verifiers and a five-criterion SME rubric. Binary verifiers provide automatable, objective pass/fail gates that prevent high-quality-looking but factually incorrect responses from receiving inflated scores; the verifier suite for each task is included in the public codebase released alongside this paper (Appendix O). Beyond verifiers, each response is scored on five ordinal criteria from 0 (absent/seriously flawed) to 3 (excellent). The criteria, what each measures, and what a 0 and 3 on each criterion indicates are summarized in Table 3. The full SME annotation protocol and the detailed 0–3 ordinal scoring rubric (i.e., how the intermediate scores 1 and 2 are awarded for each criterion) are provided in Appendix M. Each (prompt $\times$ agent) cell was graded by one SME from a recruited pool of former MBB and Big Four consultants.

The five rubric scores and the binary verifier pass rate are aggregated into a single Verifier-Rubric Score (VRS) on a 0–100 scale. Let $r_i \in \{0, 1, 2, 3\}$ denote the SME score for criterion i, let $\bar{r} = \frac{1}{5}\sum_{i=1}^{5} r_i$ be the reasoning average, and let $V \in [0, 100]$ denote the verifier pass rate. VRS has two variants:

$$\mathrm{VRS}_0 = 0.5 \cdot V + 0.5 \cdot \frac{\bar{r}}{3} \cdot 100 \quad \text{(relaxed)} \quad (1)$$

$$\mathrm{VRS} = \mathrm{VRS}_0 \cdot \mathbf{1}[\min_i r_i > 0] \quad \text{(strict)} \quad (2)$$

The strict variant zeros out the score whenever any criterion is zero. VRS is a descriptive aggregate; ACCEPT is defined directly on the underlying components rather than on a VRS threshold:

$$\mathrm{ACCEPT}(r, V) \;\Leftrightarrow\; \min_i r_i > 0 \wedge \bar{r} \geq 2.5 \wedge V \geq 80\%. \quad (3)$$

The choice of equal weights $(0.5/0.5)$ in Equation 1 is a defensible default rather than the only possible choice; the verifier layer is in fact the second-strongest predictor of binary ACCEPT, not the first, motivating a sensitivity analysis. We show in Section I.1 that ACCEPT is invariant to VRS reweighting by construction (the rule is defined on

Table 2: The five-class capability-targeted prompt taxonomy. Each class targets a distinct cognitive capability rather than a topical domain.

Prompt class	Capability tested	Description
CRP — Constrained Research Prompt	Source discipline	Information retrieval from a narrow, explicit source set. Agent must restrict itself to authorized sources when tempted by easier external alternatives.
RCP — Relevance Compression Prompt	Signal extraction	Large noisy corpora where 60–70% of the material is irrelevant. Agent must act as an intelligent filter and locate buried qualifiers, footnotes, or dispersed references.
SCP — Structural Compliance Prompt	Algorithmic obedience	Multi-layer non-trivial output structure (nested sections, fixed-column tables, JSON schemas, cross-references). Agent must hold and faithfully execute a structural specification across a long deliverable.
LDP — Latent Decomposition Prompt	Problem decomposition	Tasks requiring inference of unlisted variables (intermediate calculations, latent factors, model components not directly given). Agent must identify what needs to be computed before the analysis can proceed.
FSP — Failure-Sensitive Prompt	Precision and exactness	Tasks where small factual or logical errors invalidate the entire output (a single mis-pulled value cascades to a wrong recommendation). This tests tolerance for the kinds of small mistakes that downstream consumers cannot recover from.

the raw components $\bar{r}$ and V, not on the VRS aggregate), and that mean VRS per agent moves by less than one point under four alternative weightings, with the agent ordering o3 $>$ Gemini $>$ Claude preserved throughout. The headline conclusions reported in this paper are therefore robust to the weighting choice.

Each (prompt $\times$ agent) cell then undergoes an independent quality-control (QC) pass by a second SME from a non-overlapping pool. QC is a verification rather than a re-annotation; the QC reviewer may Confirm, Edit (with a documented one-line reason), or Reject/Return the row, with priority re-derivation on final-answer, trap, citation-dependent, and output-file verifiers, and full citation validation for source existence, claim support, and corpus-discipline compliance. The full QC protocol is in Appendix N.

3.4 Multi-Agent Evaluation Infrastructure

Our evaluation infrastructure dispatches all agents on identical task packages simultaneously through agent-specific adapters (Anthropic Messages API for Claude, OpenAI Responses API with Containers for o3, Google Interactions API for Gemini), with file-format normalization and merge-on-write result storage. All three agents themselves write the Python code that produces output files; only the execution environment differs, so file-generation failures are attributable to model code quality, not infrastructure asymmetries. The architectural comparison (Table **??**) and execution-environment details are in Appendix **??**.

4 Empirical Results

We grade all 70 prompts in our benchmark on three frontier deep research agents (210 responses in total). Verifier-Rubric Scores follow Equations 1–2; the strict variant is the default for headline reporting unless stated otherwise.

4.1 Main Results

Table 4 reports aggregate metrics across all 70 graded prompts and three agents. The most decision-relevant view is the ACCEPT rate: all three agents cluster near 13–16%, with overlapping CIs. The strict-VRS ordering is Claude (61.4) $>$ Gemini (52.6) $>$ o3 (38.5), with reasoning-mean values spanning 1.55 (o3) to 1.89 (Claude) out of 3.0. The inversion relative to ACCEPT is not a contradiction: VRS gives partial credit to all-criteria-non-zero responses regardless of whether they reach production quality, while ACCEPT is a binary threshold gate. o3's profile is dominated by catastrophic failures: 41.4% auto-reject rate (vs. 10.0% for Claude), driven by 62 criterion-zeros. Gemini accumulates even more zeros (74) but distributes them across fewer prompts.

Three patterns are worth flagging in the main results. The auto-reject rate separates agents sharply. o3 (41.4%) and Gemini (30.0%) sit well above Claude (10.0%), with the largest gap for any metric in the table. o3 accumulates 62 criterion-zeros; Gemini accumulates 74 (more than the other two combined), even though its mean reasoning of 1.68 sits close to Claude's 1.89. Claude's zero distri-

Table 3: The five SME-graded ordinal criteria (0–3 scale). Score-0 and Score-3 anchors are shown to clarify the dimension being scored; the full ordinal rubric (including the 1 and 2 anchors per criterion) is in Appendix M.

Code	Criterion	What is scored	Score 0 indicates	Score 3 indicates
DI	Data Integrity	Whether facts, numbers, citations, and references are accurate	Fabricated or seriously mis-stated data; the response asserts something verifiably false or invented	Every quantitative and source claim verifiable; no fabrication
AR	Analytical Rigor	Whether the reasoning chain is sound, sufficiently deep, and free of logical gaps	Reasoning absent, circular, or seriously flawed	Steps shown explicitly, individually correct, sufficient depth for the question
RF	Relevance & Focus	Whether the response addresses the asked question without scope drift or filler	Largely answered a different question or padded with off-topic material	Wholly on-task; every section advances the asked deliverable
EP	Execution Precision	Whether requested operations like calculations, transformations, filtering, structural construction etc are performed correctly	Right operation attempted, executed wrong; numerical or structural mistake in the output	All requested operations performed accurately, including edge cases and traps
FD	Format & Deliverability	Whether the output is a usable MC deliverable: layout, completeness, readability, professional tone	Unusable artifact: truncated, malformed, missing sections	Drop-in-ready deliverable in the requested format; all sections present, polished tone

Table 4: Main performance metrics across our evaluation ($n = 70$ graded prompts $\times$ 3 agents = 210 attempts; VRS uses the strict variant of Equation 2).

Metric	Claude Opus 4.6	o3-deep-research	Gemini 3.1 Pro
Mean reasoning $\bar{r}$ (0–3)	1.89	1.55	1.68
Mean verifier pass rate V (%)	60.2	48.6	58.2
Mean VRS, strict (0–100)	**61.4**	**38.5**	**52.6**
ACCEPT rate (binary)	15.7%	12.9%	12.9%
Auto-reject rate ($\exists\, i : r_i = 0$)	10.0%	41.4%	30.0%
Total criterion-zeros (out of $5 \times 70 = 350$)	35	62	74
Mean verifiers per task		14.9	

bution is strikingly even (exactly 7 on every criterion), consistent with a "conservative" failure mode: when Claude fails, it fails proportionally rather than catastrophically on one dimension. This high-variance pattern is examined in Section C.1.

4.2 Per-Prompt-Type Performance

Per-prompt-type strict VRS (Table 5, Appendix A) reveals agent specialization: Claude wins CRP (55.5), FSP (77.1), and RCP (71.6); o3 wins LDP (65.0); Gemini wins SCP (68.0). o3's extended reasoning excels at latent decomposition but collapses on structural compliance (SCP 12.8, lowest cell). Claude leads most strongly on FSP, where precision under cascading error is the test. Effect sizes confirm: large $|d| > 0.8$ on FSP, LDP, and SCP; small on CRP (Appendix A).

4.3 Discriminating Power

Under the binary ACCEPT criterion, 45 of 70 prompts (64.3%) are universally rejected; no prompt is universally accepted. On VRS argmax, Gemini wins 26.3 prompts (37.6%), Claude 24.8 (35.5%), o3 18.8 (26.9%). Full distributions are in Appendix B.

4.4 Robustness: Criterion Zero-Counts

Gemini accumulates 74 criterion-zeros (more than Claude's 35 and o3's 62 combined), concentrated on DI, EP, and FD (16 each). o3 has 22 DI-zeros. Claude's distribution is strikingly even (exactly 7 per criterion). Full counts are in Appendix C.

4.5 Agent-Distinct Failure-Mode Signatures

We complement the quantitative diagnostics with an LLM-based failure-tag analysis on SME free-text justifications (1,050 cells). An LLM classi-

fier (Claude Opus 4.5) assigned zero or more tags from an eight-tag taxonomy per cell; a deterministic regex classifier cross-checks (mean Jaccard 0.339). Full tag counts are in Appendix D (Table 9). Three agent-distinct signatures emerge.

o3 — Failure by breadth. o3 leads on six of eight tags. The 48 `fabricated_data` tags (2.8× Claude), 27 `no_files_read` (9× Claude), and 24 `trap_not_caught` (2.4× Claude) characterize an agent that aggressively generates content when source access fails. On CRP closed-corpus tasks, o3 produced citations from real but topically unrelated URLs, undetectable without domain verification.

Claude — Failure by imprecision. Claude's profile is dominated by `cascading_math_errors` (37, co-highest with o3) and `citation_hallucination` (11, highest). Claude's `no_files_read` count (3) is the lowest, indicating reliable file access; the count dropped from 20 (v2) to 3, confirming run-to-run variance.

Gemini — Failure by system collapse. 11 `tech_failure` tags (highest; 2.8× o3), distributed across timeouts, context-window saturation, and python-docx crashes. Combined with 74 criterion-zeros alongside the highest perfect-verifier rate (14.3%), this characterizes an agent with no graceful-degradation regime.

4.6 Pairwise Significance

Wilcoxon signed-rank tests (paired by prompt) with Bonferroni correction: Claude–o3 on VRS ($\Delta = +22.9$, $p_{\text{Bonf}}=0.001$) and on $\bar{r}$ ($\Delta = +0.33$, $p_{\text{Bonf}}=0.039$) both survive correction. All other VRS pairs and all ACCEPT-rate comparisons (McNemar, all $p > 0.79$) are not significant.

Concrete code-level and citation-level failure examples are in Appendix E.1.

4.7 Cross-Benchmark Comparison

DRA's binary verifier layer is structurally comparable to both APEX benchmarks (APEX-v1 (Vidgen et al., 2025), APEX-Agents (Vidgen et al., 2026)): all three measure the fraction of task-specific binary checks passed, enabling direct cross-benchmark comparison on that axis. DRA's ordinal rubric layer ($\bar{r}$) and dual-gate ACCEPT have no APEX counterpart. This second evaluation dimension is the methodological contribution that catches "mechanically correct but analytically shallow" responses invisible to binary-only instruments. DRA currently evaluates management consulting tasks only. APEX-v1 covers MC, IB, Law, and Medicine; APEX-Agents covers MC, IB, and Law. Cross-benchmark comparisons are therefore most directly valid for the MC subset, though APEX does not report MC-only scores separately. The full comparison is in Table 17 (Appendix J).

DRA's best-model mean verifier pass rate (60.2%, Claude) sits within the range spanned by the two APEX benchmarks (APEX-v1 64.2%, APEX-Agents 53.9%). The gap with APEX-v1 is consistent with APEX-v1's broader domain coverage and LLM-judged rather than SME-graded criteria; the proximity to APEX-Agents validates that DRA's verifier difficulty is not an outlier relative to published knowledge-work evaluation. DRA's V=100% rate (14.3%, Gemini) is mechanically harder than APEX-Agents' Pass@1 (24.0%) because DRA checks ∼15 verifiers per task vs. APEX-Agents' ∼4 criteria: the probability of passing all checks drops exponentially in the number of checks. APEX does not report paired significance tests, bootstrap CIs, or effect sizes; our statistical apparatus is a methodological contribution.

4.8 Additional Diagnostics

Several additional analyses support the robustness of the findings above and are reported in the appendices. The five rubric criteria covary at mean off-diagonal Spearman $\rho = 0.60$, with DI–EP the strongest pair ($\rho = 0.74$) and DI–FD the most independent ($\rho = 0.45$), consistent with 2–3 latent factors (Appendix G). The Spearman $\bar{r}$-vs-V correlation is 0.80 pooled, confirming the two scoring layers are correlated but not redundant. Rubric validation via Spearman-ACCEPT correlation shows EP as the strongest predictor ($\rho = +0.55$) followed by V ($+0.52$); sole-cause analysis identifies EP, DI, and FD as the three threshold-blocking criteria (Appendix H). The strict VRS auto-reject gate costs o3 the most ($+11.7$ points under relaxed scoring; Appendix F). The agent ordering Claude > Gemini > o3 on strict VRS is preserved under all five alternative weight schemes tested, with maximum shift < 1.5 points (Appendix I). Per-rubric pass rates, per-type ACCEPT rates, and per-class architectural observations are in Appendices H, A, and K respectively.

5 Limitations and Future Work

Sample size, inter-rater reliability, and significance (P0). The evaluation comprises 70 graded prompts $\times$ 3 agents (210 attempts), with per-class sample sizes of $n = 8$–18. At these sizes Cohen's $d > 0.8$ thresholds describe magnitude rather than inferential precision, and paired-comparison tests on per-agent ACCEPT outcomes do not reach $p < 0.05$ (Section H.1.4). Each (prompt $\times$ agent) cell was graded by one primary SME and independently reviewed by a second SME via the QC protocol of Appendix N. This provides an error-correction pass against rubric–evidence mismatches and surfaces fabricated citations, but does not yield a Cohen's κ inter-rater reliability statistic since QC is an asymmetric defensibility check rather than a parallel re-annotation.

Planned second release (v2). A formal IRR study with parallel double-grading on a held-out subset is in preparation, alongside an expanded prompt corpus. The v2 release will roughly double the prompt count, add Investment Banking (IB) tasks alongside Management Consulting (MC), and report bootstrap confidence intervals on every headline metric. We anticipate publishing the expanded results in a companion paper; the rubric, scoring formulae, and QC protocol used in v2 will be backward-compatible with the v1 instrument reported here so cross-release comparison remains valid.

Rubric dimensionality and other limitations (P1–P2). The mean off-diagonal Spearman correlation of 0.60 across the five reasoning criteria (Appendix G) suggests 2–3 effective latent factors rather than five orthogonal traits, motivating a factor analysis. The evaluation is single-domain (MC); generalization to other professional domains requires parallel datasets. The Investment Banking (IB) study mentioned above and additional domains are part of the v2 roadmap.

6 Conclusion

What our benchmark actually measures is whether a frontier deep research agent can do the kind of structured, multi-document, decision-grade research a management consultant gets paid to do. Across the 70 graded prompts and 210 responses, the answer is: not yet, not reliably, and not in a way that any single performance metric captures.

The agents rank differently depending on which aggregation you look at. By strict VRS, Claude leads at 61.4. By criterion-zeros, Claude is cleanest (35 vs. o3's 62 and Gemini's 74). By perfect-verifier rate (V=100%), Gemini leads (14.3%). By VRS argmax, Gemini leads with 26.3 of 70 head-to-head wins. By binary ACCEPT rate, all three are statistically indistinguishable at 13–16%. The orderings are not in conflict: Claude builds its VRS lead on a lower catastrophic-failure rate rather than on higher peak quality, while Gemini's higher zero count is offset by its non-failures more often clearing the quality bar. Reporting any one of these views in isolation would mislead.

The per-type VRS decomposition reveals that agents specialize: o3 leads on LDP (65.0) where extended reasoning pays off, while struggling catastrophically on SCP (12.8) where structural obedience is required. Gemini leads on SCP (68.0), the inverse of o3's profile. Claude leads overall but does not dominate every type.

Across these views, the failure modes are agent-specific (Section 4.5), the prompt taxonomy probes capabilities the agents differ meaningfully on (Cohen's $d > 1.0$ on two of five classes), and the five rubric criteria correlate at mean Spearman 0.60, consistent with two to three latent factors rather than five orthogonal traits. Per-class architectural observations—CRP closed-corpus weakness, agent-specific file-generation failure profiles, and file-readability asymmetries—are recorded in Appendix K.

A Per-Type Performance and Effect Sizes

Table 5: Mean strict VRS by prompt type and agent. Bold marks the per-type winner.

Type	n	Claude	o3	Gemini
CRP	13	**55.5**	38.1	47.3
FSP	8	**77.1**	50.3	59.7
LDP	13	46.7	**65.0**	19.5
RCP	18	**71.6**	36.3	61.7
SCP	18	59.1	12.8	**68.0**

A.1 Effect Sizes for Per-Type Differences

To check whether the per-type VRS gaps in Table 5 reflect real signal or sampling noise from small per-arm samples ($n = 8$–18), we compute Cohen's $d = (\bar{r}_1 - \bar{r}_2)/s_{\text{pooled}}$ (Cohen, 1988) on the reasoning average $\bar{r}$ for each pair of agents within each prompt type (Table 6). Magnitudes follow the con-

ventional thresholds 0.2 (small), 0.5 (medium), 0.8 (large) (Sawilowsky, 2009).

The per-type winners on Table 5 carry $|d| > 1.0$ on FSP (Claude vs. o3 $d = +1.27$), where Claude has the strongest reasoning-mean profile, and on SCP where o3 is the loser ($d = +1.45$ Claude over o3; $d = -1.54$ o3 vs. Gemini). LDP shows a large o3 advantage over both ($d = -0.82$ Claude vs. o3; $d = +1.68$ o3 vs. Gemini). CRP shows small effect sizes across all pairs ($|d| \leq 0.45$), supporting the conclusion that no agent is meaningfully better on constrained-research tasks. We caution that at $n = 8$ per arm on FSP, Cohen's conventional thresholds describe magnitude but not inferential precision.

B Inter-Agent ACCEPT Distribution

B.1 Discriminating Power and Inter-Agent Agreement

We examine the benchmark's discriminating power from two complementary angles: a binary view (ACCEPT count per prompt) and a continuous view (VRS argmax per prompt).

Under the binary ACCEPT criterion of Equation 3, 45 of 70 prompts (64.3%) are universally rejected and no prompt produces a unanimous ACCEPT (Table 7). Under the continuous VRS view, however, we observe meaningful capability differentiation. Gemini wins 26.3 prompts (37.6%), Claude wins 24.8 prompts (35.5%), and o3 wins 18.8 prompts (26.9%) on the per-prompt VRS argmax (fractional tie-attribution; ties distributed equally between tied agents).

Reconciling the two views, the benchmark is binary-hard but continuously discriminating: the ACCEPT threshold ($\bar{r} \geq 2.5$ and $V \geq 80\%$) is set tight relative to current frontier-DRA capability, so most prompts receive zero ACCEPTs in absolute terms, yet the continuous VRS reveals large-effect-size differential capability among the same agents on the same prompts. For research-grade benchmark use, the continuous VRS view is more informative; for production-readiness assessment, the binary view is decision-relevant. The 90.5% VRS-SME concordance validates VRS as a faithful summary of human expert judgment.

C Per-Criterion Zero Counts

C.1 Robustness Diagnostic: Criterion Zero-Counts

Counting hard zeros per criterion is a robust statistic that does not depend on the rest of the score distribution and complements the means analyzed in Table 5. Table 8 reports zero counts for each (criterion, agent) cell, with the corresponding per-criterion means.

Gemini stands out: 74 zeros, more than the other two combined (Claude 35, o3 62). Gemini's zeros concentrate on DI, EP, and FD (16 each), the signature of an agent that either nails the deliverable or collapses on source fidelity, calculation, and format simultaneously. o3 stands out on DI with 22 zeros: in nearly a third of prompts the SME judged o3's response as containing fabricated or seriously misstated data. EP is equally concerning for o3 (16 zeros) and Gemini (16 zeros), indicating execution precision as a shared weak point. Claude's zero distribution is strikingly even (exactly 7 on every criterion), consistent with a "conservative" failure mode: when Claude fails, it fails proportionally rather than catastrophically on one dimension.

D Failure-Mode Tag Counts

E Concrete Failure Examples

E.1 Concrete Failure Examples

We provide code- and citation-level evidence for two of the failure signatures discussed above, to ground the agent-distinct narratives in concrete artefacts.

We include code-level and citation-level examples of two of the failure signatures discussed in Section 4.5, to support reproducibility and to ground the agent-distinct narratives in concrete artifacts.

Gemini — Wrong `python-docx` API

Listing 1: Gemini-generated code with wrong python-docx API call

```
# Extracted from Gemini response on a
    CRP manufacturing task
table1 = doc.add_table(rows=1, cols=5)
table1.style = 'Table␣Grid'
hdr_cells = table1.rows.cells  # ERROR:
    _Rows has no .cells attribute
hdr_cells[0].text = 'Line'
hdr_cells[1].text = 'Effective␣Hours'
```

Listing 2: Corrected python-docx API usage

Table 6: Cohen's d effect sizes for per-prompt-type pairwise gaps on the reasoning average $\bar{r}$. Sign convention: positive d favors the first-listed agent. Bold marks $|d| > 0.8$.

Type	n/arm	Claude vs o3	Claude vs Gemini	o3 vs Gemini	Reading
CRP	13	$+0.45$	$+0.22$	-0.29	all small; no winner
FSP	8	$\mathbf{+1.27}$	$\mathbf{+0.94}$	$+0.19$	large edge for Claude over o3
LDP	13	$\mathbf{-0.82}$	$+0.54$	$\mathbf{+1.68}$	large edge for o3 over Claude; very large o3 over Gemini
RCP	18	$\mathbf{+0.91}$	$+0.34$	-0.43	large edge for Claude over o3
SCP	18	$\mathbf{+1.45}$	-0.36	$\mathbf{-1.54}$	very large edge for Claude over o3; very large Gemini over o3

Table 7: Inter-agent ACCEPT-count distribution under the binary decision rule of Equation 3, across the 70 graded prompts.

# agents accepting (out of 3)	# prompts	%
0 (universally rejected)	45	64.3
1 (one agent accepted)	21	30.0
2 (two agents accepted)	4	5.7
3 (universally accepted)	0	0.0

```
# Correct API:
table1 = doc.add_table(rows=1, cols=5)
table1.style = 'Table␣Grid'
hdr_cells = table1.rows[0].cells  # rows
    [0] returns _Row, which has .cells
hdr_cells[0].text = 'Line'
hdr_cells[1].text = 'Effective␣Hours'
```

The `table.rows.cells` error appears identically across multiple independent Gemini responses, supporting the interpretation that the failure is a systematic gap in Gemini's `python-docx` pre-training coverage rather than a one-off slip.

o3 — Hallucinated Citations on a Closed-Corpus Task

The following citations appeared in o3-deep-research's response to a closed-corpus CRP task on last-mile logistics rider reallocation in India:

- https://datasheet.eeworld.com.cn/view/85833854.html — Chinese electronics datasheet (ZVY11K1210401 component)
- https://stackoverflow.com/questions/13385860 — Stack Overflow: "How can I remove extra whitespace from strings when parsing a CSV file in Pandas?"
- https://stackoverflow.com/questions/42039629 — Stack Overflow: "pandas: Read multiline CSV like input with different separators"

None of these sources have any relationship to the logistics domain or the rider reallocation analysis. All URLs are real pages that exist on the internet, which makes them difficult to identify as hallucinated without domain verification. This underscores our recommendation that closed-corpus benchmarks include automated citation domain classification as a binary verifier.

F Strict vs. Relaxed VRS

F.1 Strict vs. Relaxed VRS

Table 10 compares the two VRS variants of Equations 1–2.

The agent ranking o3 > Gemini > Claude holds under both variants. The Claude-o3 gap shrinks from 20.74 points (strict) to 11.89 points (relaxed), so the ordering is not driven by the auto-reject penalty alone. There is genuine reasoning-quality separation underneath. Claude gains the most when the strict penalty is removed ($+8.85$), reflecting that 14 of its 42 attempts had at least one criterion zero, typically DI (because the response carried fabricated content), zeroing out an otherwise-respectable composite. The relaxed VRS therefore overstates Claude's production-readiness while the strict VRS understates its reasoning capability; both views together give the full picture. We recommend that for external reporting one use strict VRS (decision-faithful with respect to ACCEPT); and for internal architecture analysis, report both variants alongside the auto-reject rate.

Table 8: Hard-zero counts per criterion (out of 70 attempts per agent), with per-criterion means in parentheses.

Criterion	Claude	o3	Gemini
DI — Data Integrity	7 (1.67)	**22** (1.23)	16 (1.63)
AR — Analytical Rigor	7 (1.73)	8 (1.57)	12 (1.77)
RF — Relevance & Focus	7 (2.37)	5 (2.04)	14 (1.89)
EP — Execution Precision	7 (1.53)	**16** (1.17)	16 (1.46)
FD — Format & Deliverability	7 (2.13)	11 (1.74)	16 (1.67)
Total zeros	**35**	**62**	**74**

Table 9: Per-prompt failure-mode tag counts from LLM classification of SME free-text justifications. Each cell is the number of unique (agent, prompt) pairs out of 70 where any of the five criterion-justifications carried that tag; bold marks the highest agent for each tag.

Failure tag	Claude	o3	Gemini	Total
`fabricated_data`	17	**48**	16	81
`cascading_math_errors`	37	**38**	21	96
`noise_inclusion`	10	**28**	21	59
`missing_section`	16	**25**	16	57
`no_files_read`	3	**27**	4	34
`tech_failure`	7	4	**11**	22
`trap_not_caught`	10	**24**	13	47
`citation_halluc.`	**11**	7	5	23

G Internal Structure: Criterion Correlations

G.1 Internal Structure: Criterion Correlations

The five rubric criteria are designed to capture distinct dimensions, but in practice they covary substantially. Table 11 reports the Pearson correlation matrix across the 210 pooled attempts.

The off-diagonal correlations range from 0.45 (DI $\times$ FD) to 0.74 (DI $\times$ EP), with mean $\rho \approx 0.60$. This indicates the rubric is most likely measuring 2-3 underlying latent factors rather than five orthogonal traits. DI and EP at $\rho = 0.74$ are highly overlapping implying that data integrity and execution correctness are not cleanly separable in practice. The lowest correlation, DI $\times$ FD at 0.45, suggests Data Integrity (whether facts are correct) and Format & Deliverability (whether the output is presented usably) are the most independent pair since fabrication and presentation can vary independently. Methodological implication of this is that the rubric criteria are statistically correlated, so treating them as orthogonal dimensions in an aggregate score overstates the rubric's effective *statistical* dimensionality. This statistical-correlation finding does not, however, imply that any criterion is dispensable for the ACCEPT decision: the rubric-validation analysis in Section H.1 (specifically the sole-cause attribution diagnostic) finds that each criterion contributes distinct information at some point along the decision surface, with two criteria contributing primarily through their distributional correlation and three through threshold-localized blocking. A formal factor analysis to determine the true latent structure is listed as a P1 next step (Section 5).

We also report the Spearman correlation between the verifier pass rate V and the reasoning average $\bar{r}$ within each agent: 0.75 (Claude), 0.76 (o3), 0.77 (Gemini), pooled 0.80. The two scoring layers are strongly correlated overall but not redundant.

H Rubric Validation

H.1 Rubric Validation: Composite Calibration and Sole-Cause Analysis

A composite scoring rubric must satisfy two desiderata to be defensible: (i) no single criterion should dominate the composite outcome, otherwise the remaining criteria are decorative; (ii) each criterion should provide non-redundant signal, other-

Table 10: Strict vs. relaxed VRS comparison and counts of attempts with at least one criterion zeroed.

Agent	Strict VRS	Relaxed VRS	Δ	# attempts with $\min_i r_i = 0$
Claude Opus 4.6	61.44	61.51	+0.06	7 / 70
o3-deep-research	38.53	50.18	+11.65	29 / 70
Gemini 3.1 Pro	52.56	57.15	+4.59	21 / 70

Table 11: Pearson correlation matrix among the five reasoning criteria, pooled across all 210 attempts. Bold marks the two extremes.

	DI	AR	RF	EP	FD
DI	1.00	0.62	0.54	**0.74**	**0.45**
AR	0.62	1.00	0.58	0.70	0.59
RF	0.54	0.58	1.00	0.56	0.66
EP	**0.74**	0.70	0.56	1.00	0.54
FD	**0.45**	0.59	0.66	0.54	1.00

wise the rubric is over-specified. We assess both with three diagnostics, applied in the spirit of Krippendorff (Krippendorff, 2011) on rubric calibration and following the analytical structure used by Liang et al. (Liang et al., 2023) for the HELM benchmark. Throughout this subsection we use Spearman rank correlation in preference to Pearson, since the rubric scores are ordinal and not interval-spaced (Spearman, 1904).

H.1.1 Spearman Correlation of Each Rubric with the Composite ACCEPT Outcome

Table 12 reports the Spearman rank correlation between each rubric score and the binary ACCEPT outcome, both pooled and per-agent.

Every rubric is significantly correlated with ACCEPT, and the correlations span a moderate range (0.31 to 0.55) without a single dominant contributor. This is the empirical signature of a well-calibrated composite: if any one rubric had a near-perfect correlation with ACCEPT (e.g., $\rho > 0.85$), the others would be effectively redundant. The per-agent correlations are stable across all three architectures (Claude/o3/Gemini), indicating the rubric does not behave differently for any specific agent's response style.

H.1.2 Sole-Cause Failure Analysis

A sole-cause failure for rubric R is a response that would have ACCEPTed except that R alone scored below 2 (with $V \geq 80\%$ and all other rubrics ≥ 2). This diagnostic, used in similar form by Wang et al. (Wang et al., 2024c) for category-level error attribution, isolates the rubrics that genuinely act as gating constraints from those that fail in clusters with others.

Of the 59 responses that cleared the $V \geq 80\%$ verifier hurdle, 42 had all five rubrics ≥ 2 (29 of these ACCEPTed; the remaining 13 fell short of the $\bar{r} \geq 2.5$ threshold despite no rubric being below 2). The sole-cause cases were the 12 responses that had exactly one rubric below 2 with $V \geq 80\%$. Their distribution is shown in Table 13.

Spearman correlation and sole-cause attribution measure two complementary aspects of how each rubric contributes to ACCEPT. Correlation summarizes how a rubric tracks ACCEPT across the entire score distribution. Sole-cause attribution, on the other hand, measures *boundary-localized importance*, by which we mean a rubric's contribution restricted to cases where the decision is genuinely uncertain. This corresponds to responses sitting close to the $\bar{r} \geq 2.5$, $V \geq 0.80$, or $\min_i r_i > 0$ thresholds, where a single rubric's value can tip the decision either way. The two measures together paint a more complete picture than either alone, and the dataset surfaces three points worth recording.

First, three rubrics are independent threshold blockers. EP, DI, and FD can each single-handedly drop a response below ACCEPT, accounting for all 12 sole-cause rejections in the dataset, with EP doing so most often (5 of 12). EP has the highest sole-cause count and also the strongest Spearman correlation with ACCEPT ($\rho = 0.55$), confirming its role as the single most informative predictor of production readiness. The two measures disagree because they ask different questions. Correlation

Table 12: Spearman correlation between each rubric score and the binary composite ACCEPT outcome ($n = 210$ pooled; $n = 70$ per agent). All overall correlations are highly significant after Bonferroni correction for six tests.

Rubric	ρ **(overall)**	p	ρ **(Claude)**	ρ **(o3)**	ρ **(Gemini)**
V (verifier pass rate)	$+0.524$	3.35×10^{-16}	$+0.568$	$+0.534$	$+0.500$
AR – Analytical Rigor	$+0.470$	5.79×10^{-13}	$+0.536$	$+0.468$	$+0.426$
EP – Execution Precision	$+0.548$	7.68×10^{-18}	$+0.602$	$+0.546$	$+0.533$
DI – Data Integrity	$+0.491$	3.80×10^{-14}	$+0.475$	$+0.522$	$+0.490$
RF – Relevance & Focus	$+0.338$	5.18×10^{-7}	$+0.323$	$+0.375$	$+0.321$
FD – Format & Deliverability	$+0.354$	1.38×10^{-7}	$+0.379$	$+0.376$	$+0.312$

Table 13: Sole-cause failure attribution: of the 12 responses where exactly one rubric scored below 2 (with $V \geq 80\%$ and all other rubrics ≥ 2), the failing rubric is reported.

Rubric	**Sole-cause failures**	**% of sole-cause cases**
EP – Execution Precision	**5**	**41.7%**
DI – Data Integrity	4	33.3%
FD – Format & Deliverability	3	25.0%
AR – Analytical Rigor	0	0%
RF – Relevance & Focus	0	0%

is averaged across the whole distribution, so FD's signal is diluted by responses far from the decision boundary. Sole-cause attribution is restricted to the boundary, where FD's threshold-blocking role is dispositive. Removing EP, DI, and FD from the composite would have incorrectly accepted 12 currently rejected responses.

Second, two rubrics carry distributional information without driving rejections. RF and AR never single-handedly block ACCEPT, despite both carrying moderate-to-strong correlations with ACCEPT ($\rho \approx 0.39$ and 0.55 respectively). When RF or AR fails, the verifier pass rate or another rubric is failing alongside, because their failure modes are coupled with those of other criteria. This does not make them redundant. Their pairwise correlation with ACCEPT reflects substantial information about distinguishing higher-quality from lower-quality *accepted* responses, even though they do not drive new rejections at the boundary. The two measures are picking up different facets of the rubric: pairwise correlation captures graded distinctions across the score range, while sole-cause attribution captures decisiveness at the rejection threshold. Together, every rubric in the composite is doing distinct work by at least one of the two measures. Strictly, this does *not* establish that any rubric is causally independent of any other; multi-cause rejections (where two or more rubrics fail together) are excluded from the sole-cause count for all rubrics involved, and the dataset does not separate their individual contributions in those cases.

Third, the auto-reject rule is a dormant principled safeguard. The rule $\min_i r_i > 0$ caught zero unique cases. All 57 auto-rejected responses also failed either $\bar{r} \geq 2.5$ or $V \geq 0.80$. This makes the rule empirically inactive in this dataset, but it is retained as a defensive prior. A single criterion at zero is, by stipulation, a catastrophic failure that should override an otherwise-passing mean. The analogy to LLM calibration safeguards in Kadavath et al. (Kadavath et al., 2022) is direct. Both are policies that may never fire on the empirical evaluation set but exist to block failure modes the current dataset does not yet contain.

H.1.3 Per-Rubric Pass Rates and Agent-Specific Weak Points

Table 14 reports the share of responses scoring ≥ 2 on each rubric, by agent.

EP is the weakest rubric at 46.0% pooled pass rate, with Claude (31.0%) far below the others; Gemini achieves a majority pass at 59.5%. This is consistent with the broader observation that current deep research agents struggle with multi-step quantitative reasoning (Hendrycks et al., 2021b), manifesting as 50 `cascading_math_errors` tags across the three agents in Section 4.5. DI is Claude's specific weak point (42.9%), the

Table 14: Per-rubric pass rates (fraction of responses scoring ≥ 2), by agent.

Rubric	Claude	o3	Gemini	Pooled
EP – Execution Precision	47.1%	30.0%	51.4%	42.9%
DI – Data Integrity	58.6%	32.9%	62.9%	51.4%
AR – Analytical Rigor	67.1%	52.9%	70.0%	63.3%
RF – Relevance & Focus	84.3%	72.9%	70.0%	75.7%
FD – Format & Deliverability	77.1%	65.7%	62.9%	68.6%

lowest cell in the table other than EP, consistent with Claude's fabrication signature analyzed in Section 4.5: when input data is fabricated, downstream EP necessarily collapses too, which explains Claude's low EP despite a lower `cascading_math_errors` count than o3. EP is Gemini's weakest rubric (59.5%).

H.1.4 Caveat: Pairwise Significance Testing on the Composite Outcome

A natural follow-on question is whether the per-agent ACCEPT rates (Claude 9.5%, o3 9.5%, Gemini 21.4%) reflect statistically detectable differences. We apply a paired binomial test on discordant prompts (the McNemar approach used by McNemar (McNemar, 1947) and adopted in benchmark-comparison contexts by Dietterich (Dietterich, 1998)) on the 70 prompts where all three agents were graded:

No pair shows a statistically detectable difference in ACCEPT rates at $\alpha = 0.05$. Discordant prompts are evenly split for every pair. The headline ordering is therefore a point estimate without paired-test significance support; the continuous VRS analysis (Sections 4.1–A.1) is on firmer statistical ground because the underlying gaps are larger relative to the sampling variability. This is consistent with the small-n caveat raised in Section 5 and reinforces the recommendation that bootstrap CIs and paired-comparison adjustments be added before any external publication (Efron, 1979; Holm, 1979).

I Weight Sensitivity

I.1 Sensitivity to VRS Weight Choice

The equal weighting of 0.5 on the verifier pass rate V and 0.5 on the reasoning-mean term $\bar{r}/3 \cdot 100$ in Equation 1 is a defensible default but not the only possible choice. The rubric validation analysis (Section H.1) shows that the verifier layer is the second-strongest predictor of binary ACCEPT, behind only EP: $\rho_{V,\text{ACCEPT}} = 0.524$, with EP at $\rho = 0.552$ and the remaining rubric criteria (AR, DI, RF, FD) all correlating with ACCEPT less strongly ($\rho = 0.466, 0.464, 0.375, 0.306$ respectively). The case for downweighting V inside the VRS aggregate is therefore weak, but for completeness we recomputed VRS under four alternative weightings:

- (A) $0.35V + 0.65\bar{r}_{\text{scaled}}$ — moderate rubric upweighting.
- (B) $0.40V + 0.60\bar{r}_{\text{scaled}}$ — mild rubric upweighting.
- (C) $0.25V + 0.75\bar{r}_{\text{scaled}}$ — aggressive rubric upweighting.
- (D) Per-criterion weights set proportional to each predictor's Spearman magnitude with ACCEPT (V 0.20, EP 0.21, AR 0.17, DI 0.17, RF 0.14, FD 0.11).

The analysis is shown in Table 16.

Three findings come out of this (Table 16). First, ACCEPT is invariant to VRS reweighting by construction: the rule (Equation 3) is defined on the raw components $\bar{r}$ and V, not on the VRS aggregate, so all four alternatives produce ACCEPT outcomes identical to the headline numbers (Claude 15.7%, o3 12.9%, Gemini 12.9%). Second, mean strict VRS per agent shifts modestly under rubric upweighting (maximum shift less than 1.5 points across all variants; Claude ranges from 60.79 to 62.15), and the agent ordering (Claude > Gemini > o3) is preserved under every variant. Third, the substantive conclusions reported elsewhere in this section are therefore robust to the weighting choice.

Table 15: Paired ACCEPT comparisons across agent pairs ($n = 70$ prompts where all three were graded). p-values from two-sided binomial test on discordant pairs.

Pair	Both ACC	A only	B only	Both REJ	p
Claude vs. o3	1	10	8	51	0.815
Claude vs. Gemini	3	8	6	53	0.791
o3 vs. Gemini	0	9	9	52	1.000

Table 16: VRS sensitivity to weight choice. Mean VRS per agent (**strict variant**, with the auto-reject gate enforced) under five weighting schemes. The two right-most columns report the result of redefining ACCEPT as "VRS $\geq T$" for a threshold T calibrated so the total ACCEPT count matches the baseline 17, and the Jaccard similarity of the resulting per-response ACCEPT set against the baseline rule $\bar{r} \geq 2.5 \wedge V \geq 0.80$. The agent ordering (o3 > Gemini > Claude) on mean VRS is preserved under every variant.

Variant	Weights	Claude	o3	Gemini	Cal. T	Split (C/o3/G)	Jaccard
Baseline (relaxed)	$0.50V + 0.50\,\bar{r}_{\text{sc}}$	61.51	50.18	57.15	—	11 / 9 / 9	—
(A) moderate rubric upweight	$0.35V + 0.65\,\bar{r}_{\text{sc}}$	61.87	38.86	52.14	—	—	—
(B) mild rubric upweight	$0.40V + 0.60\,\bar{r}_{\text{sc}}$	61.73	38.75	52.28	—	—	—
(C) aggressive rubric upweight	$0.25V + 0.75\,\bar{r}_{\text{sc}}$	62.15	39.07	51.85	—	—	—
(D) Spearman-proportional	V 0.19, EP 0.20, DI 0.18, AR 0.17, FD 0.13, RF 0.12	60.79	38.51	51.27	—	—	—

J Cross-Benchmark Comparison Table

K Architectural Observations

K.1 Architectural Observations on Per-Class Capability Differences

Two further architectural observations follow from the per-class analysis in Section A.1.

CRP is universally weak. Effect sizes on closed-corpus reasoning satisfy $|d| \leq 0.46$ on all CRP pairs. Current frontier agents do not reliably enforce closed-corpus instructions. In particular, o3-deep-research fabricates real-but-topically-unrelated citations on closed-corpus tasks. This is a failure invisible from response text alone, since the cited URLs resolve to legitimate published material. This observation prompted our subsequent URL-domain verifier, which checks whether cited domains point to a task-specified allowed-source list rather than merely checking citation syntactic well-formedness.

File-generation performance varies sharply across agents. Claude's high completion rate is accompanied by elevated fabrication-tag counts (Section 4.5); Gemini's research-quality content frequently succeeds while its code-generation step fails on a domain-specific `python-docx` API error (paragraph-style mishandling); o3 occasionally treats file-output instructions as optional and returns a structured response inline instead of as the requested artefact. Future benchmarks should test file-formatting code generation as a separate dimension rather than conflating it with research quality.

L Worked Examples per Prompt Class

To illustrate how the abstract prompt-class definitions translate into concrete tasks, we provide one worked example for each of the five prompt classes. Each example reproduces three artefacts from the underlying corpus: (i) the **Prompt** delivered to the agent (with company names anonymized per the data-release policy); (ii) the **Sanity Check** written by the SME during task authoring, which lists the failure mode a naive solver is expected to fall into (Lazy AI Test) versus the reasoning chain a domain-aware solver must execute (Expert Test); and (iii) the **Solution Logic**, which is the step-by-step deterministic derivation of the golden answer used during grading. A representative set of full prompt packages, including the underlying input files, is included in the public codebase (Appendix O).

L.1 CRP — Constrained Research Prompt (Market Strategy)

CRP tasks require the agent to conduct analysis under explicit operational constraints that limit the solution space. Constraints are typically methodological (use only provided files, plus a single specifically-authorized external source), procedural (a fixed entry method or a fixed accounting convention that the agent must respect), or scope-related (a single binary decision under defined thresholds).

Prompt (anonymized).

> *You are working as a management consultant and a client wants to enter a new geographical area*

Table 17: Cross-benchmark comparison of DRA with APEX-v1 and APEX-Agents. "Binary-check pass rate" rows are structurally comparable across benchmarks (fraction of per-task binary checks passed). "Unique to DRA" rows have no APEX counterpart. Best-model CIs are 95% bootstrap.

	DRA (ours)	**APEX-v1**	**APEX-Agents**
Design			
Domains	MC only	MC + IB + Law + Med	MC + IB + Law
n tasks	70	200	480
Binary checks/task	$\sim$15 verifiers	$\sim$29 criteria	$\sim$4 criteria
Ordinal rubric?	Yes (5 criteria, 0–3)	No	No
Grading	Human SME	LLM judge	LLM judge
Binary-check pass rate (comparable metric)			
Definition	Mean verifier pass rate	Mean Score	Mean Score
Best model	60.2% [53.3, 66.8]	64.2% [$\sim$61, $\sim$67]	53.9% [$\sim$52, $\sim$56]
Task-level binary (comparable metric)			
Definition	$V \geq$ threshold	—	Pass@1 (all criteria)
V=100% / Pass@1	14.3% [7.1, 22.9]	—	24.0% [20.2, 28.0]
$V \geq 80\%$	32.9% [22.9, 44.3]	—	—
Unique to DRA			
Mean $\bar{r}$ (0–3 ordinal)	1.89 [1.70, 2.10]	N/A	N/A
ACCEPT ($V \geq 80\%$ & $\bar{r} \geq 2.5$)	15.7% [7.1, 24.3]	N/A	N/A
VRS strict (blended)	61.4 [55.1, 67.5]	N/A	N/A

and has approached you for the GTM plan. The client is a leading Indian automotive component manufacturer and now wants to expand in the EU. They want to capture 10% of the EU market share in 5 years with an initial investment of INR 500 Cr. Give a Yes or No decision to the client for entering the market using only the files provided by the client; market reports may be used from Mordor Intelligence only, no other market reports, blogs, broker research, or third-party data are allowed. Also state any assumptions taken for the analysis. The client has asked for a PowerPoint slide as a one-pager GTM plan deck; it should include Basic company information, Investment goal, Product-Market fit as 2×2 matrices, Value Proposition, Target Market, and Competitor Analysis.

Sanity Check.

Lazy AI Test. A basic AI will use other ways of entry such as a manufacturing setup or a joint venture, ignoring the note in the Excel file that restricts entry to the export-only method which flips the Go/No-Go decision to Go, the opposite of the correct answer.

Expert Test. A consultant fetching the Mordor report and the Excel data, applying the export-only entry method, and excluding the non-usable slides would reach a No Go. One correct answer exists.

Solution Logic.

USD to INR exchange rate used: spot rate for 22/04/2026.

Market size (from Mordor Intelligence report). Total EU market in 2031 $\approx$ INR 43,500 Cr; 10% target $\approx$ INR 4,350 Cr in revenue.

Channel decomposition. The market splits into OEM ($\sim$80%) and aftermarket ($\sim$20%). OEM lock-in periods are long and hard to crack within a 5-year horizon under an export-only model, so the OEM contribution can be set to effectively zero. Even on a very optimistic view, an attainable OEM share is 5% of INR 35,000 Cr = INR 1,750 Cr.

Accessible aftermarket is too small. Aftermarket $\approx$ 20% ($\sim$INR 8,700 Cr). With aggressive execution, a 15% share gives INR 1,300 Cr.

Headline. Total achievable revenue is INR 1,300 Cr (no OEM) to INR 3,050 Cr (with 5% OEM = 1,750 + 1,300 Cr).

Golden range answer. INR 1,300 Cr to INR 3,050 Cr, which is below the 10% target of INR 4,350 Cr.

Final answer. *No Go*.

This creates a hybrid information-access pattern: closed-corpus for the client's internal data, with a single specifically-authorized external source (Mordor Intelligence) for the market sizing. The agent must correctly scope its web search and must apply the export-only entry method noted in the source files rather than considering alternative entry methods that change the answer.

L.2 RCP — Relevance Compression Prompt (Service Operations)

RCP tasks supply a deliberately noisy corpus where the majority of source material is irrelevant to the analytical question. The agent must filter, locate the buried qualifying detail or contradicting footnote, distinguish root drivers from outcome metrics, and present only the signal in the deliverable. Reported

benchmarks may be directionally misleading and must be reconstructed using policy-defined corrections.

Prompt (anonymized).

A large Indian telecom company operates a 2,400-seat blended contact center. Current performance shows Service Level at 78% (target 85%), Average Handle Time (AHT) at 6.8 minutes (target 5.5), First Call Resolution (FCR) at 71%, and Cost per Call at INR 48. NPS stands at 71 (target 80). The COO must submit a Performance Benchmarking and Gap Analysis before the quarterly operations review in 10 days; a flawed analysis risks continued cost inefficiency or misallocation of improvement investments.

Task (precise). *Using ONLY the four provided internal files: (1) benchmark performance against industry top-quartile and average performers; (2) identify and rank the top 3 performance gaps by financial and customer impact; (3) qualitatively assess the cost and NPS impact of each gap using policy-defined relationships; (4) recommend the top 2 priority actions with expected impact and timeline.*

Important constraints. *Use ONLY the provided files. No external data or assumptions. All conclusions must reconcile across ALL four files. Reported benchmark data may be directionally misleading. True benchmarks must be reconstructed where required. Non-operational data (marketing, ESG, unrelated IT projects) must be filtered out.*

Output format (strict). *Table 1 - Performance Benchmarking & Gap Analysis, with columns [Metric, Current, Industry Average, Top Quartile, Gap vs. Top Quartile, Impact (Qualitative)] for five rows: Service Level, AHT, FCR, Cost per Call, NPS. Section A - Top 3 Gaps and Impact Analysis (≤180 words). Section B - Priority Recommendations (≤120 words). Mandatory final line:* "Top Priority: Implement AI-powered call routing and real-time coaching to address AHT and FCR gaps, driving cost efficiency and NPS improvement."

Sanity Check.

Lazy AI Test (must fail). A standard model will use the headline AHT benchmark of 5.5 min, overstate the AHT gap, treat Cost per Call as a root gap, and recommend generic fixes like hiring more agents. It will fail to apply the Appendix C correction, distinguish root drivers from outcome metrics, and use the policy-defined relationships.

Expert Test (must pass). A domain-aware solver will: (1) adjust the AHT benchmark to 6.2 min using policy, (2) identify FCR and AHT as primary drivers, (3) treat Cost per Call as a derived outcome metric, (4) use policy relationships to assess impact, and (5) recommend AI routing plus real-time coaching.

Deterministic outcome. Top gaps: FCR (11 pp), AHT (0.6 min), Service Level (10 pp). Root cause: capability gap in routing and agent enablement. Recommendation: AI routing + real-time coaching.

Solution Logic.

Decision archetype. Performance benchmarking + gap analysis.

Step 1 - reconstruct the true benchmark. Headline Top Quartile AHT = 5.5 min (Performance Deck). Policy adjustment (Appendix C, Footnote 9): true Top Quartile AHT = 6.2 min, because the headline excludes training, coaching, and complex call-handling time. Other benchmark metrics are used as reported.

Step 2 - validate current performance. Current AHT (6.8 min), FCR (∼71%), Service Level (∼78%), and Cost per Call (∼INR 48) all align with cross-site averages in the operational dataset.

Step 3 - compute true gaps. AHT gap: $6.8 - 6.2 = 0.6$ min. FCR gap: $82\% - 71\% = 11$ pp. Service Level gap: $88\% - 78\% = 10$ pp. Cost per Call gap: INR $48 - 32 = 16$ (derived outcome). NPS gap: $82 - 71 = 11$ points.

Step 4 - prioritization logic. Rank by customer impact (NPS sensitivity) and cost impact (operational efficiency). Primary operational drivers: FCR → resolution quality → NPS; AHT → efficiency → cost. Derived outcome metrics: Cost per Call, NPS. Policy relationships (Appendix D): FCR strongly influences NPS; AHT strongly influences cost.

Step 5 - top 3 ranked gaps. (i) FCR gap (11 pp) - highest customer impact; primary NPS driver. (ii) AHT gap (0.6 min) - primary cost driver. (iii) Service Level gap (10 pp) - queue performance and customer experience. Cost per Call is a derived outcome, not a root operational gap.

Step 6 - root-cause diagnosis. Evidence across policy and operational data indicates lack of advanced call routing, limited real-time coaching/agent enablement, and gaps in agent support systems. Top-performing centers exhibit AI-powered routing, strong agent enablement, and integrated self-service.

Step 7 - recommendations. Top priority: AI-powered call routing + real-time coaching. Expected directional impact: reduce AHT, improve FCR, lower Cost per Call, improve NPS - all consistent with the policy-defined relationships in Appendix D. Secondary action: strengthen the knowledge base and increase self-service deflection.

Golden answer. Top gaps: FCR (11 pp), AHT (0.6 min), Service Level (10 pp). Root cause: capability gap in routing and agent enablement. Recommendation: AI routing + real-time coaching.

Verifier checks include whether the agent's filter produced the correctly scoped subset, whether out-of-scope filler (marketing, ESG, unrelated IT) appears in the response, whether the Appendix C policy correction is applied to the AHT benchmark, and whether Cost per Call and NPS are correctly characterized as derived outcome metrics rather than as root gaps to attack directly.

L.3 SCP — Structural Compliance Prompt (Cost Optimization)

SCP tasks include explicit schema requirements that must be satisfied for the response to be considered valid. They test the agent's ability to follow precise structural instructions while conducting substantive analysis. A correct number inside a malformed envelope counts as a hard failure on the Format & Deliverability criterion; a wrong number inside a valid envelope partially passes FD.

Prompt (anonymized).

> ***Context and stakes.*** *A listed Indian consumer-durables manufacturer has experienced a 320 bps margin decline over the last two quarters. The Board has mandated an immediate cost-optimization program focused on manufacturing efficiency. The COO must present a validated cost-reduction plan within 5 days; the analysis will be directly reviewed by the Board Finance Committee. Critical constraint: the committee will only accept outputs that strictly follow the prescribed reporting structure. Any deviation in format will result in outright rejection, regardless of analytical correctness.*
>
> ***Task.*** *Using ONLY the provided data files (Excel, PDF, and Assumptions): (1) identify the true operational manufacturing cost per unit; the reported cost includes embedded adjustments that must be identified and excluded, and these adjustments are not explicitly labeled and may appear in notes, footnotes, or appendix sections. (2) Apply a 12% cost reduction ONLY on the operational cost base. (3) Compute: true operational unit cost, reduced unit cost, annual cost before optimization, annual cost after optimization, absolute savings, and percentage savings.*
>
> ***Critical analytical requirements.*** *Reconcile inconsistencies across files (Excel vs. PDF vs. Assumptions). Identify and exclude non-operational cost components. Do NOT assume the "Total" value in the Excel file is final. Use ONLY the provided annual production volume. No external assumptions allowed.*
>
> ***Artifact requirement (strict SCP, hard failure if violated).*** *Return ONLY a valid JSON object with the exact structure below. No explanations, no comments, no additional keys, no missing keys, no reordered keys.*

```
{
  "cost_analysis": {
    "unit_cost_reported": number,
    "unit_cost_operational": number,
    "unit_cost_reduced": number
  },
  "annual_metrics": {
    "annual_cost_before": number,
    "annual_cost_after": number,
    "absolute_savings": number,
    "percentage_savings": number
  },
  "decision": {
    "recommendation": "ACCEPT" or "REJECT",
    "justification_flag": "MEETS_TARGET" or "DOES_
  }
}
```

> ***Formatting rules.*** *All monetary values → INR Crores (2 decimal places). Percentages → 2 decimal places. JSON must be strictly valid and machine-parseable. Keys must appear in exact order. No trailing commas.*
>
> ***Decision rule.*** *If percentage savings $\geq$ 10% → ACCEPT, else → REJECT.*
>
> ***Hidden complexity.*** *A portion of the overhead cost in the Excel file includes a non-operational allocation that is explained only in the PDF appendix; the reported total cost therefore overstates the true operational cost.*

Sanity Check.

> **Lazy AI Test.** The model will use INR 110 without adjustment OR will fail the JSON format check.
>
> **Expert Test.** The model will remove the INR 7 non-operational overhead, compute correctly, and emit strict JSON in the prescribed key order.

Solution Logic.

> Reported unit cost = INR 110. Remove non-operational overhead = INR 7 (cross-referenced from the PDF appendix). True operational cost = INR 103.
>
> Reduced unit cost = 103×0.88 = INR 90.64.
>
> Annual cost before = $103 \times 1{,}500{,}000$ = INR 154.5 Cr. Annual cost after = $90.64 \times 1{,}500{,}000$ = INR 135.96 Cr.
>
> Absolute savings = INR 18.54 Cr; percentage savings $\approx$ 12%.
>
> **Decision: ACCEPT** (savings $\geq$ 10% threshold).

The structural compliance test is independent of the analytical answer. The verifier layer parses the response as JSON and checks key ordering and datatypes before any numeric value is inspected; a malformed envelope is graded as a hard FD failure even when the underlying analytics are correct.

L.4 LDP — Latent Decomposition Prompt (Operations Research)

LDP tasks state a final objective but require the agent to infer the intermediate variables, coefficients, or sub-problems that must be solved before the final answer can be computed. The decomposition itself is the test: a passing response derives each latent quantity from the provided data, then formulates and solves the underlying optimization problem before computing the headline number.

Prompt (anonymized).

An American metal-fabrication firm has four product portfolios catering to four different industries - Aerospace, Automotive, Defense, and Electronics. Each product, regardless of industry, must undergo a standard lifecycle of end-to-end production across four departments (Drilling, Milling, Turning, Assembly), not necessarily in sequence; each department may have underlying sub-steps. The firm was established in 2010 and has four strong vendor relationships (`Vendor_abc`, `Vendor_def`, `Vendor_ghi`, `Vendor_jkl`) relied upon to achieve desired production.

The CEO wants to know the maximum total overall contribution that can be generated from the product portfolio in Year 2014, together with a clear Go / No-Go decision evaluation.

Use `Product-Vendor Time per Unit Details.xlsx` to extract the production time for each (product-industry, department) combination by choosing the minimum time applicable across vendors. Use `Historical Contribution Per Unit.xlsx` to derive the average contribution (in USD/unit) for each industry bucket for Year 2014, computed by averaging the per-unit contribution across the past 4 years (2010, 2011, 2012, 2013). Use `Department Sub-Activity Constraint.xlsx` to derive the maximum total hours available for each department, computed by summing the hours available for each sub-step within that department.

Create a mathematical model that maximizes total overall contribution for 2014, computed as the sum-product of unit counts and per-unit contributions, by product industry.

Strict do-nots. *Do NOT browse the web for any information; rely only on the internal files provided. Use the production-time-per-unit only from the named Excel file. Use the per-industry 2014 contribution only from the named Excel file. Use the per-department maximum hours only from the named Excel file. Unit counts for Aerospace, Defense, Automotive, and Electronics must be integer and non-negative. The minimum total contribution to qualify as a Go decision is USD 200. Do not hallucinate or fabricate datapoints; strictly adhere to the business logic and inputs provided.*

Output. *Produce a Word document named `Maximum Dollar Contribution 2014` containing a clear Go / No-Go evaluation and the maximum total overall contribution in USD, rounded to the nearest dollar.*

Sanity Check.

Lazy AI Test. The prompt embeds several failure surfaces that defeat a basic LLM: (i) three complex Excel files where the relationships between inputs are not explicit and require strong relational reasoning to be discovered; (ii) the presence of confusing or side-tracking data points that derail an LLM which does not carefully scope the inputs to use; (iii) the need to invoke a proper linear-integer-programming library to derive the maximized contribution that drives the Go/No-Go decision; (iv) the need for clear stepwise aggregation logic to produce a stress-validation answer.

Expert Test. A single business-logic path leads, step by step, to the unique correct mathematical answer; any incorrect logic or missing step produces a wrong answer.

Solution Logic.

Decision archetype. Go / No-Go decision.

Let X_1, X_2, X_3, X_4 be the integer non-negative number of units to be produced for Aerospace, Automotive, Defense, and Electronics, respectively.

Objective. Maximize the total 2014 contribution $X_1c_1+X_2c_2+X_3c_3+X_4c_4$, where c_i is the average per-unit contribution (USD/unit) for industry i in 2014, derived as the average of the 2010-2013 per-unit contributions from `Historical Contribution Per Unit.xlsx`.

Constraints (department-hour budgets, derived as sums over sub-activities from `Department Sub-Activity Constraint.xlsx`).

- Drilling: $3X_1 + 7X_2 + 4X_3 + 0X_4 \leq 70$.
- Milling: $0X_1 + 2X_2 + 4X_3 + 6X_4 \leq 80$.
- Turning: $3X_1 + 4X_2 + 0X_3 + 5X_4 \leq 90$.
- Assembly: $4X_1 + 6X_2 + 5X_3 + 3X_4 \leq 100$.
- $X_1, X_2, X_3, X_4 \in \mathbb{Z}_{\geq 0}$.

Solve as a linear integer program. If the maximum total contribution exceeds USD 200, the decision is Go; otherwise No-Go.

Golden range answer. A Word document containing a rounded dollar value; the acceptable numeric final output is USD 290-310.

The decomposition test is whether the agent correctly identifies the latent variables (X_i), derives the per-industry contribution coefficients from the historical-averaging rule, derives the department-hour right-hand sides from the sub-activity sums, formulates the LP with the correct integer/non-negativity constraints, and only then computes the headline number. Several frontier agents in our evaluation correctly identify the LP structure but mis-derive at least one coefficient from the source files, producing a confident-looking wrong answer.

L.5 FSP — Failure-Sensitive Prompt (Cost Optimization)

FSP tasks construct a precision point where a single mis-pulled value or mis-applied formula invalidates the entire recommendation. The trap is deliberately built into the source materials, typically as a stale or placeholder value that contradicts a live external authority, and the verifier layer detects whether the agent caught it.

Prompt (anonymized).

Context. *You are a Supply Chain Strategy Consultant advising a global fast-fashion conglomerate. The client is undergoing a Zero-Based Budgeting (ZBB) review for FY2026. The CSCO needs to make a final procurement decision regarding their highest-volume ocean freight lane: Shenzhen (Yantian) to Rotterdam. The client must choose between signing a "Fixed Annual Contract" with a 3PL carrier, or floating their volume on the "Spot Market," which carries a volatile fuel surcharge.*

Task. *Calculate the Total 2026 Projected Freight Cost (in USD) for the Shenzhen-to-Rotterdam lane under both the Fixed Contract and the Spot Market options, and recommend the most cost-effective routing strategy.*

Workflow. *(1) Review* `Global_Lane_Volumes_FY26.csv` *and isolate the annual TEU (Twenty-foot Equivalent Unit) volume specifically for the Shenzhen-to-Rotterdam lane. (2) Review* `Ocean_Carrier_Metrics.csv` *to determine the Base Spot Rate and the Fuel Consumption factor (tons of fuel burned per TEU) for this specific lane. (3) Calculate the Spot Market Fuel Surcharge: read* `2026_Freight_Sourcing_Policy.txt` *carefully, determine the correct price per ton for Marine Fuel, and multiply by the total tons of fuel required for this lane's annual volume. (4) Add Base Spot Cost to Fuel Surcharge to obtain the Total Spot Market Cost. (5) Compare against the Fixed Annual Contract cost.*

Constraints & deliverables. Web search required: *adhere strictly to the fuel pricing policy. If a live rate is mandated, you must search the live web for the current USD price of the specified fuel index and cite your exact source.* Format & decision commit: *output a structured ZBB Memo containing Total Annual TEU Volume for the target lane; Total Fixed Contract Cost; Total Spot Market Base Cost (excluding fuel); Total Spot Market Fuel Surcharge Cost; Total Spot Market All-In Cost; and a definitive final recommendation written exactly as* `DECISION: SIGN FIXED CONTRACT` *or* `DECISION: USE SPOT MARKET`.

Sanity Check.

Lazy AI Test. A standard LLM will read the policy text file, lazily grab the $400.00 internal marine-fuel placeholder, calculate a deflated fuel surcharge of $3,240,000, and arrive at a Total Spot Cost of $8,190,000, causing it to incorrectly recommend `DECISION: USE SPOT MARKET` and exposing the client to massive market loss.

Expert Test. An experienced supply-chain consultant would extract the correct 4,500 TEU volume from the noise, correctly sequence the base freight and the 1.8× fuel multiplier, adhere to the strict exception policy overriding the $400 baseline, fetch the live VLSFO market rate, and mathematically prove that the volatile Spot Market exceeds the $9.9M Fixed Contract ceiling.

Solution Logic.

External data required. The agent must search the live web for the current "VLSFO Global 20 Ports Average" price (typically published by Ship&Bunker or similar maritime indices). In mid-2026 this fluctuates around $600-$650 USD per metric ton.

Step-by-step trace.

- Filter lane to Shenzhen → Rotterdam; Annual volume = 4,500 TEUs.
- *Fixed Contract Cost:* $4{,}500 \times \$2{,}200 = \$9{,}900{,}000$.
- *Spot Market Base Cost:* $4{,}500 \times \$1{,}100$/TEU $= \$4{,}950{,}000$.
- *Spot Market Fuel Surcharge (the FSP trap):* fuel per TEU = 1.8 tons; total fuel $= 4{,}500 \times 1.8 = 8{,}100$ tons. The agent must reject the $400 decoy placeholder in the policy file and fetch the live VLSFO average. Assuming $620/ton: $8{,}100 \times \$620 = \$5{,}022{,}000$.
- *Spot Market All-In Cost:* $\$4{,}950{,}000 + \$5{,}022{,}000 = \$9{,}972{,}000$.
- *Compare and decide:* $\$9.90\text{M} < \$9.97\text{M} \Rightarrow$ `DECISION: SIGN FIXED CONTRACT`.

Golden answer range. Fixed Cost: exactly $9,900,000. Spot Base Cost: exactly $4,950,000. Total Spot Cost: varies between $9.6M and $10.5M depending on the live VLSFO price pulled. Decision: assuming VLSFO is trading above $612/ton (which it historically does), the decision must be `DECISION: SIGN FIXED CONTRACT`.

The trap is whether the agent treats the provided policy file as authoritative or independently verifies the live VLSFO index. Several frontier agents in our evaluation accept the $400 placeholder without checking, which inverts the decision and produces a confident but wrong recommendation. The verifier layer checks both the numerical outputs and the final decision string for an exact-match comparison against the golden range.

M Detailed SME Rubric Definitions

Each response is scored on five reasoning criteria by a Subject Matter Expert (SME) with relevant management-consulting expertise. Each criterion is scored on the integer scale 0 = absent or seriously flawed, 1 = poor, 2 = adequate, 3 = excellent. The full ordinal rubric describing how each of the four scores is awarded for each criterion is provided in Appendix M.1 below; the brief dimension definitions that appear in Table 3 of the main paper are reproduced and elaborated here for completeness:

- **DI — Data Integrity**: Whether facts, numbers, citations, and references are accurate. A 0 indicates fabricated or seriously mis-stated data i.e.,

the response asserts something that is verifiably false or invented.

- **AR — Analytical Rigor**: Whether the reasoning chain is sound, sufficiently deep for the question, and free of logical gaps or hand-waving. A 3 means the agent shows the steps and the steps are correct.
- **RF — Relevance & Focus**: Whether the response addresses the asked question without irrelevant content, scope drift, or filler. A 0 indicates the response largely answered a different question or padded itself with off-topic material.
- **EP — Execution Precision**: Whether requested operations like calculations, transformations, filtering, structural construction, are performed correctly. A 0 indicates the agent attempted the right operation but executed it wrong.
- **FD — Format & Deliverability**: Whether the output is presented as a usable MC deliverable: appropriate layout, completeness, readability, professional tone. A 0 indicates an unusable artifact (truncated, malformed, missing sections).

The five criteria are designed to capture distinct dimensions of reasoning quality, but in practice they correlate substantially across our dataset (mean off-diagonal $\rho \approx 0.60$; see Section G.1).

The annotation protocol pairs each ordinal score with a free-text justification guided by keyword prompts. SMEs were drawn from a recruited pool of management consultants (former MBB, Big Four Strategy, and Tier-2 firm consultants), and each (prompt $\times$ agent) cell was graded by exactly one SME. We discuss the implications of this protocol for inter-rater reliability and statistical inference in Section 5 of the main paper.

M.1 Ordinal Scoring Rubric per Dimension

Table 18 reproduces the criterion-by-criterion 0-3 scoring rubric used by SMEs and QC reviewers during annotation. Each cell describes the qualitative bar for the indicated score on the indicated dimension. The rubric is held constant across all 70 prompts and all three agents.

The Score-0 cell on every dimension is the canonical auto-reject trigger under the ACCEPT rule (Equation 3). For DI and EP the Score-0 condition is explicit (fabricated data; fundamental math errors), so an SME observing those signatures should set the score to 0 and a QC reviewer should confirm rather than soften. For AR, RF, and FD the Score-0 bar is qualitatively stricter (*seriously* flawed, *dominated by noise*, *unusable*); QC reviewers are instructed to upgrade Score-0 entries to Score-1 when the SME's free-text justification describes Score-1 behaviour, both to save the row from being rejected for the wrong reason and to surface rubric–justification mismatches.

N Annotation Quality Control Protocol

Every (prompt $\times$ agent) cell graded by a primary SME is independently reviewed by a second SME drawn from a non-overlapping QC pool. The QC pass is a verification rather than a re-annotation: the QC reviewer checks whether the primary SME's scoring is defensible against the rubric and the evidence (prompt, solution logic, response text, output files, and citations), not whether the QC reviewer would have scored the same. Two thoughtful SMEs may defensibly score a 0-3 ordinal differently; QC intervenes only when the primary score or justification contradicts the rubric, contradicts the response or file evidence, or is internally inconsistent.

N.1 QC Actions

For each verifier and each meta-criterion the QC reviewer records one of three actions:

- **Confirm** — the primary SME's score and justification stand. The QC cell is left empty (or marked "OK" if the workbook requires).
- **Edit** — the primary SME's entry is incorrect. The QC reviewer writes the exact replacement (corrected 0/1 verifier or 0-3 meta-criterion score, plus a complete 2-4-sentence replacement justification matching the rubric cell for the new score) along with a one-line auditable reason. The replacement overwrites the primary entry verbatim downstream.
- **Reject/Return** — the row is not salvageable by surgical edit. Triggered by: more than three verifier edits on a single row; any fabricated citation; missing scores on any verifier; output file missing or placeholder when the prompt required one; or multiple meta-criteria with score–justification mismatches.

N.2 Verifier QC: Coverage and Depth

Every verifier on every row receives a coverage check (verify the primary SME's 0/1 is consistent with the response, file, and citation evidence). A

Table 18: Per-dimension 0-3 ordinal scoring rubric. Reproduced from the SME Annotation Guideline (Section 5.1).

Criterion	Score 0 (Fail)	Score 1 (Poor)	Score 2 (Acceptable)	Score 3 (Strong)
DI — Data Integrity & Source Discipline	Completely ignores source constraints; fabricated data or sources.	Multiple factual errors; vague citations; use of 1-2 unauthorized sources.	Mostly accurate with minor (1-2) errors; primary data from permitted sources; 10-20% sourcing gaps.	100% adherence to authorized sources; all facts verified; no hallucinations.
AR — Analytical Rigor	Illogical or self-contradicting; fails to identify the core task; no inference attempted.	Gaps in the logic; no evidence-backed conclusions; misses interdependencies or key variables.	Sound reasoning with minor gaps; identifies most key variables; clear logic chain but has small leaps.	Strong logic with cause-effect; identifies "hidden" variables; MECE decomposition of the problem.
RF — Relevance & Focus	Response dominated by noise; includes forbidden content.	50-60% relevant; 10-20% scope creep.	70-80% relevant and within scope; low tangential content (<20%).	90%+ relevant; excludes out-of-scope content.
EP — Execution Precision	Fundamental math errors; wrong units; black-box output.	Correct approach but results off by 10%+; partial unit normalization; gaps in methodology.	Minor computational errors (<5%); all units normalized; most calculation steps shown.	100% accurate calculations; unit discipline; complete audit trail.
FD — Format & Deliverability	Ignores format; unusable output; no actionable insights.	Partial format compliance; lack of specificity in recommendations.	Follows format with minor (10-20%) deviations; actionable but some interpretation needed.	100% compliance with format/constraints; actionable and executive-ready recommendations.

subset of verifiers receives full-rigour re-derivation rather than consistency-checking, in priority order:

1. *Final-answer verifier* (always re-derived; verified against the output file when one exists).
2. *Trap verifier* (the verifier tied to the prompt's embedded cognitive trap; always re-derived).
3. *Numeric verifiers with tight tolerances* (3-5 recomputed per row; more if SME scores look suspiciously uniform).
4. *Citation-dependent verifiers* (any verifier whose pass condition names a specific source).
5. *Output-file verifiers* (any verifier requiring the output deliverable; verified inside the file directly).

N.3 Citation Validation

For every citation-dependent verifier, the QC reviewer opens the cited source and confirms three things: the cited URL or document resolves and matches the named source; the source actually supports the specific claim attributed to it (not a nearby claim or a paraphrase that reshapes the figure); and the source is within the authorized-source list named in the prompt. Any failure on these three checks is recorded as an Edit on the affected verifier with reason "citation invalid: [specific issue]". A fabricated citation list (sources that do not exist, or quotations not present in the sources) triggers Reject/Return: fabricated citations are surfaced to the rubric owner rather than silently corrected.

N.4 Meta-Criterion QC: Three Checks

For each of the five meta-criteria, the QC reviewer runs three checks in order. **Check A (rubric–justification match):** for the score the SME assigned, the rubric cell text and the SME's justification should describe the same outcome; mismatches indicate the justification supports a different score than the one given. **Check B (verifier coherence):** Execution Precision should track the numeric-verifier pass rate (a high pass rate paired with EP = 0/1 or low pass rate paired with EP = 3 is suspect); Data Integrity & Source Discipline should track the primary-source/trap verifier outcome and any citation-validation failures from the previous subsection. **Check C (overall-comment consistency):** the SME's free-text overall comment should be tonally and factually consistent with the five scores; comments that contradict a score (for example, "reasoning is aligned" paired with DI = 0) are flagged.

N.5 Score-0 Rule

Any meta-criterion at 0 triggers automatic REJECT of the response under the ACCEPT rule (Equation 3), so Score-0 entries receive priority QC scrutiny. A justification that describes a Score-1 outcome ("multiple factual errors") paired with a Score-0 entry (rubric cell: "fabricated data; no

adherence to sources") is the canonical edit case: the correct QC action is Edit to 1, both saving the row from being rejected for the wrong reason and surfacing the rubric–justification mismatch.

N.6 Star Ratings

In addition to the item-level QC, the QC reviewer records six holistic star ratings per response: one combined rating across all verifiers, and one per meta-criterion. The star ratings are recorded post-QC (after any edits) and provide a second, coarser quality signal independent of the underlying ordinal scores. They are not used in the primary VRS or ACCEPT computation reported in this paper but are retained as a per-cell quality covariate for future analysis.

O Evaluation Infrastructure Code

The full evaluation infrastructure, including agent adapters (Claude/OpenAI/Gemini), result storage, diagnostic tooling, and task specification format, is available at:

https://anonymous.4open.science/status/dra-response-gen-4288

The 70-prompt corpus will be released separately.

Key infrastructure components:

- `csv_loader.py`: Task batch dispatcher with file resolution, multi-agent dispatch, and result aggregation.
- `adapters/claude_adapter.py`: Claude Opus 4.6 adapter with tool-use support.
- `adapters/openai_adapter.py`: o3-deep-research adapter with Containers API integration for file output tasks.
- `adapters/gemini_adapter.py`: Gemini deep-research adapter with Interactions API, local code execution, and format-aware file generation instructions.
- `results_store.py`: Merge-on-write result storage supporting partial reruns without data loss.
- `diagnose_run.py`: Automated failure categorization and per-agent cost/success reporting.

References

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Xavier Bouthillier, Pierre Delaunay, Mirko Bronzi, Assya Trofimov, Brennan Nichyporuk, Justin Szeto, Nazanin Mohammadi Sepahvand, Edward Raff, Kanika Madan, Vikram Voleti, Samira Ebrahimi Kahou, and Christopher Pal. 2021. Accounting for variance in machine learning benchmarks. *Proceedings of Machine Learning and Systems*, 3:747–769.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021a. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borber, Jason Gong, Kaiyang Cai, Jiayi Patel, and 1 others. 2021b. FinQA: A dataset of numerical reasoning over financial data. In *EMNLP*.

Jacob Cohen. 1988. *Statistical Power Analysis for the Behavioral Sciences*, 2nd edition. Lawrence Erlbaum Associates.

Janez Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *JMLR*, 7:1–30.

Thomas G Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923.

Zhiwei Du and 1 others. 2025. Deepresearch bench: Benchmarking deep research agents. *arXiv*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Collin Burns. 2024. Length-controlled AlpacaEval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

Bradley Efron. 1979. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1–26.

Jiawei Gu, Xuhui Yan, Zhichao Shi, Lingfeng Si, Zhangyue Li, Lixin Qin, Xiang Feng, and Shanshan Xia. 2024. A survey on llm-as-a-judge. *arXiv:2411.15594*.

Neel Guha, Julian Nyarko, Daniel E Ho, and 1 others. 2024. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *NeurIPS*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021a. Measuring massive multitask language understanding. *ICLR*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. Measuring mathematical problem solving with the MATH dataset. *NeurIPS*.

Sture Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2023. A survey on hallucination in large language models. *arXiv:2311.05232*.

Pranab Islam, Anand Kannappan, Rik Goyal, and 1 others. 2023. Financebench: A new benchmark for financial question answering. *arXiv:2311.11944*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. 2024. SWE-bench: Can language models resolve real-world GitHub issues? *ICLR*.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. *EMNLP*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *ACL*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv:2207.05221*.

Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. 2023. Benchmarking cognitive biases in large language models as evaluators. *arXiv preprint arXiv:2309.17012*.

Klaus Krippendorff. 2011. Computing Krippendorff's alpha-reliability. *Departmental Papers (ASC)*.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *EMNLP*.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2024. Dawn of LLM hallucinations. *arXiv preprint arXiv:2311.01862*.

Minghao Li, Feifan Song, Bowen Yu, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. 2023a. API-Bank: A comprehensive benchmark for tool-augmented LLMs. *EMNLP*.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023b. AlpacaEval: An automatic evaluator of instruction-following models. *arXiv preprint arXiv:2305.14387*.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2023. Holistic evaluation of language models. *TMLR*.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023a. Is your code generated by ChatGPT really correct? rigorous evaluation of large language models for code generation. *NeurIPS*.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, and 1 others. 2024. Agentbench: Evaluating llms as agents. *ICLR*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023b. G-Eval: NLG evaluation using GPT-4 with better human alignment. *EMNLP*.

Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin Clement, Dawn Drain, Daxin Jiang, Duyu Tang, and 1 others. 2021. CodeXGLUE: A machine learning benchmark dataset for code understanding and generation. *NeurIPS Datasets and Benchmarks*.

Yubo Ma, Zhibin Cao, Leila Tavabi, and 1 others. 2024. SciAgent: Tool-augmented language models for scientific reasoning. *arXiv preprint arXiv:2402.11451*.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. *EMNLP*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *ACL*.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2024. Gaia: A benchmark for general ai assistants. *ICLR*.

OpenAI. 2024. SWE-bench Verified: Resolving real-world GitHub issues at scale. *OpenAI Technical Report*.

OpenAI. 2025. Browsecomp: A simple challenge for browsing agents. *OpenAI Research*.

Arjun Panickssery, Samuel R Bowman, and Shi Feng. 2024. LLM evaluators recognize and favor their own generations. *arXiv preprint arXiv:2404.13076*.

Shishir G Patil, Tianjun Zhang, Xin Wang, and Joseph E Gonzalez. 2023. Gorilla: Large language model connected with massive APIs. *arXiv preprint arXiv:2305.15334*.

Siddharth Patwardhan and 1 others. 2025. Gdpval: Evaluating llms with expert deliverables. *arXiv*.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, and 1 others. 2024. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. *ICLR*.

Shlomo S Sawilowsky. 2009. New effect size rules of thumb. *Journal of Modern Applied Statistical Methods*, 8(2):597–599.

Akash Sharma and 1 others. 2025. Researchrubrics: Expert rubric evaluation for research agents. *arXiv*.

Charles Spearman. 1904. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, and 1 others. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *TMLR*.

Bertie Vidgen and 1 others. 2025. Apex: Assessing professional expertise. *Mercor Research*.

Bertie Vidgen and 1 others. 2026. Apex-agents: Evaluating deep research agents on professional tasks. *Mercor Research*.

Jiarui Wang and 1 others. 2025. Profbench: Benchmarking llms on professional knowledge work. *arXiv*.

Peiyi Wang, Lei Li, Liang Chen, Dawei Cai, Zefan Niu, and 1 others. 2024a. Large language models are not fair evaluators. *ACL*.

Xiaoxuan Wang, Ziniu Hu, Pan Lu, Yanqiao Zhu, Jieyu Zhang, Satyen Subramaniam, Arjun R Loomba, Shichang Zhang, Yizhou Sun, and Wei Wang. 2024b. SciBench: Evaluating college-level scientific problem-solving abilities of large language models. *ICML*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, and 1 others. 2024c. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. *NeurIPS*.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, and 1 others. 2023. The rise and potential of large language model based agents: A survey. *arXiv:2309.07864*.

Zihao Xu and 1 others. 2025. Researcherbench: Benchmarking research agents on scientific tasks. *arXiv*.

Fengji Zhang, Bei Chen, Yue Zhang, Jacky Keung, Jin Liu, Daoguang Zan, Yi Mao, Jian-Guang Chen, and Wenqiang Chen. 2023a. RepoCoder: Repository-level code completion through iterative retrieval and generation. *EMNLP*.

Yue Zhang, Yafu Li, Leyang Cui, and 1 others. 2023b. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv:2309.01219*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *NeurIPS*.

Lucia Zheng, Neel Guha, Brandon R Anderson, Peter Henderson, and Daniel E Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the CaseHOLD dataset of 53,137+ legal holdings. *ICAIL*.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, and 1 others. 2024. Webarena: A realistic web environment for building autonomous agents. *ICLR*.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. 2021. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. *ACL*.