REGBENCH: SOURCE-GROUNDED BENCHMARKS FOR REGULATORY CROSS-REFERENCE REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Compliance work in regulated domains requires tracking cross-references across sections and applying the resulting chain to numeric or categorical decisions. Existing benchmarks test long-document retrieval and multi-hop QA, but the benchmarks reviewed here do not isolate traversal over explicit regulatory cross-reference graphs. We introduce RegBench, a specialised benchmark comprising **827 expert-level questions grounded in 4,766 atomic-fact propositions** from DNV Ship Rules and the U.S. Basel III capital regulation, 12 CFR Part 217. Frontier systems reach only **17–49%** strict accuracy on DNV and **17–57%** on Basel, with clear tier degradation as required context scope increases, and an independent judge from a third vendor reproduces the ranking. Three published graph-RAG methods further collapse to $\leq 17\%$ strict at pilot scale. To scale such evaluations, we present a novel framework that converts regulatory cross-reference graphs into source-grounded QA items with minimal human effort. This is achieved through a graph-anchored generation pipeline that samples reference chains to synthesise scenarios, and a calibrated audit (v5d) that bounds quality without expert review of every question. Our automated audit agrees with full-set SME annotation on **99.0%** of audited items and **99.1%** of atomic facts on DNV, and transfers to 12 CFR Part 217 without tuning. Together with independently validated answer grading, these results support strict atomic-fact evaluation at benchmark scale. Code and data are available at `https://github.com/regbench/regbench`.

1 INTRODUCTION

Regulated industries — banking, maritime classification, aviation, pharmaceuticals, nuclear licensing, tax and statutory codes — are governed by codified rules (Board of Governors of the Federal Reserve System, 2024; DNV, 2025; U.S. Federal Aviation Administration, 2024; U.S. Food and Drug Administration, 2024; U.S. Nuclear Regulatory Commission, 2024; U.S. House of Representatives, Office of the Law Revision Counsel, 2024). Complex compliance checks can require tracing references across chapters and combining the cited provisions into a defensible verdict. The benchmarks reviewed below do not directly test this process on regulatory source text.

Question answering (QA) over long technical documents is the natural test of this read-and-derive capability: it requires a model to locate the relevant clauses, follow them through the document, and return an answer grounded in the cited text. But existing QA datasets miss what makes regulatory review hard: an **extensive chain of cross-references** that the answer must traverse. This chain creates two coupled sub-problems. **Challenging question generation**: each item must name an applicability anchor, traverse the regulatory graph, and resolve to a verdict that depends on every step. **Hard annotation**: an expert must verify each derived fact against verbatim source, step by step, a time-consuming process. A model can also land the right verdict while skipping or fabricating steps; strict atomic-fact conjunction grading exposes such unearned credit, at the cost of lower scores and per-step gold labels that scale only with an automated audit.

We propose **RegBench** (Figure 1), a regulatory benchmark that addresses each sub-problem directly. For question generation, a graph-anchored *generation pipeline* mirrors regulatory traversal: it samples chains from the corpus' cross-reference graph, LLM-synthesises practitioner-style scenarios over them, and encodes each item as a tuple $(\textit{scenario}, \textit{anchor}, \textit{chain}, \textit{required_facts}, \textit{verdict})$ scored by strict atomic-fact conjunction, so any missing step breaks the verdict. For annotation, a calibrated

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

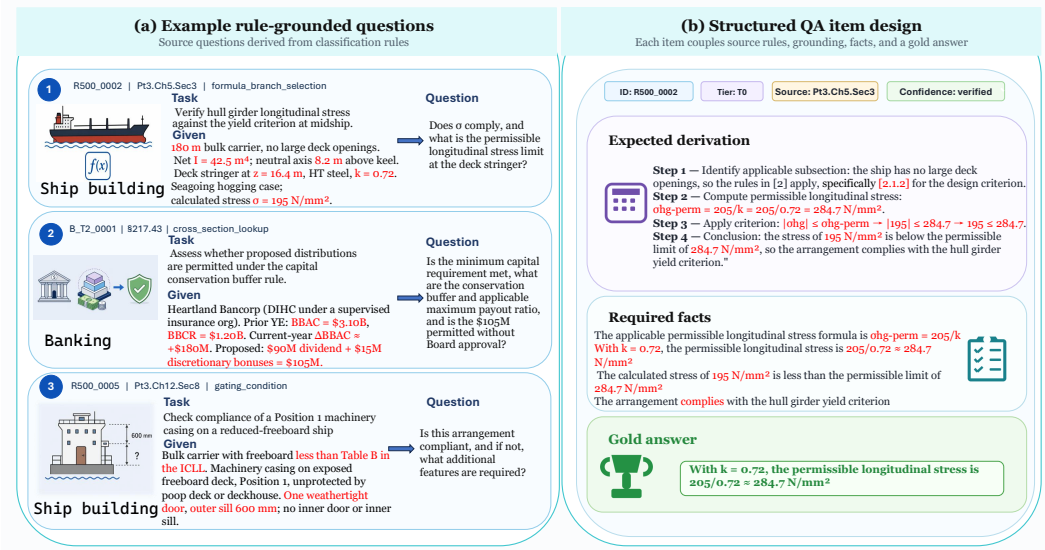

Figure 1: Overview of RegBench. The pipeline (§3) extracts the cross-reference graph from a regulatory corpus, samples chains at controlled context scope, synthesises source-grounded scenarios over those chains, and gates the released set through a calibrated selective audit (v5d, §3.3). Each released item is a tuple $(\textit{scenario}, \textit{anchor}, \textit{chain}, \textit{required_facts}, \textit{verdict})$ scored by strict atomic-fact conjunction; tiers range from local short-context facts to broader cross-reference contexts (T0–T4 on DNV, T0–T3 on Basel, where cross-Part chains exit the Part 217 corpus and are excluded by scope).

selective audit (v5d) — an LLM filter that decomposes each item into per-fact checks and routes flagged items to subject-matter experts (SMEs) for source-grounded repair — applies locked rules across splits and corpora without retuning. On the held-out DNV TEST split (disjoint from the TRAIN items used to derive v5d) it misses **0/446** facts on passed items (rule-of-three $\leq 0.67\%$), and it agrees with SME re-review on 99.0% of items and 99.1% of facts in the combined 200-item DNV pool; the Basel evidence is thinner than DNV's (Appendix B). Applying the framework to DNV Ship Rules (DNV, 2025) and 12 CFR Part 217 (Board of Governors of the Federal Reserve System, 2024) yields **827 questions / 4,766 atomic-fact propositions** (546 DNV + 281 Basel). Seven credentialed reviewers source-grounded the flagged items and sampled passes under the v5d-advisory protocol (Appendix B).

Contributions.

- We propose a **portable graph-anchored generation pipeline** that turns a regulatory corpus' `xref` graph into chain-anchored, source-grounded QA items; it ports across two regulatory regimes without retuning and extends to implicit, term-based references in civil law (§4.5).
- We design a **calibrated selective audit (v5d)** that gates release quality without per-question expert authoring; its locked rules carry over from a development split to held-out splits and a second corpus without retuning, with a **0/446** fact-level miss rate (rule-of-three $\leq 0.67\%$) on the held-out DNV TEST split.
- We construct **RegBench**, with **827 questions / 4,766 atomic-fact propositions** from DNV Ship Rules and 12 CFR Part 217. Frontier accuracy tops out at 17–57% strict with clear tier degradation, an independent cross-judge preserves 40 of 42 pairwise orderings (§4.4), and three published graph-RAG methods (ArchRAG (Wang et al., 2026a), HippoRAG (Gutiérrez et al., 2024), GraphRAG (Edge et al., 2024)) score $\leq 17\%$ strict at pilot scale (Appendix N).

Table 1: Comparison with representative benchmark families. "Long-doc / real-source input" requires both a long source document in context and authentic source text; "audited intermediate-fact keys" concerns how reference keys are derived, not how answers are scored. ✓: explicit construction/evaluation target; ○: partial or adjacent coverage; ✗: not explicit.

Benchmark	Long-doc / real-source input	Professional / regulated domain	Source-defined regulatory `xref` chain	Scenario-to-verdict reasoning	Audited intermediate-fact keys	Calibrated LLM audit / judge
HotpotQA (Yang et al., 2018)	○	✗	✗	✗	○	✗
RULER (Hsieh et al., 2024)	○	✗	✗	✗	✗	✗
UDA (Hui et al., 2024)	✓	○	✗	○	○	✗
MMLongBench-Doc (Ma et al., 2024)	✓	✗	✗	○	○	✗
LegalBench (Guha et al., 2023)	○	✓	✗	○	○	✗
FinBen (Xie et al., 2024)	○	✓	✗	○	○	✗
PHANTOM (Ji et al., 2026)	✓	✓	✗	○	○	✗
IndustryEQA (Li et al., 2025)	○	✓	✗	○	○	✗
LEGIT (Lee et al., 2026)	✗	✓	✗	✓	✓	✓
PRBench (Akyürek et al., 2026)	○	✓	✗	○	○	✓
PLawBench (Shi et al., 2026)	○	✓	✗	✓	✓	✓
JudgeBench (Tan et al., 2025)	✗	✗	✗	✗	○	✓
RegBench	✓	✓	✓	✓	✓	✓

2 Related Work

We compare RegBench with representative benchmarks across six construct axes (Table 1; per-row rationale in Appendix A). We then walk through QA-benchmark families and benchmark-validation work in turn.

QA benchmarks. HotpotQA and related benchmarks study linked evidence and compositional reasoning in open-domain, synthetic-rule, or event-centric settings (Yang et al., 2018; Trivedi et al., 2022; Tafjord et al., 2021; Li et al., 2024). Benchmarks such as LongBench shift attention to long-context retrieval, stress testing, real documents, and evidence-position effects (Bai et al., 2024; An et al., 2024; Hsieh et al., 2024; Xu et al., 2024; Hui et al., 2024; Ma et al., 2024; Liu et al., 2024). LegalBench and related specialised-domain benchmarks move evaluation into legal, financial, industrial, and other specialised settings (Guha et al., 2023; Xie et al., 2024; Ji et al., 2026; Li et al., 2025; Han et al., 2026; Xie et al., 2026). Concurrent legal and professional benchmarks evaluate open-ended reasoning with expert rubrics (Lee et al., 2026; Akyürek et al., 2026; Shi et al., 2026); these examples grade aspects of intermediate reasoning, but do not build items from the explicit cross-reference paths of a statutory corpus. Yet source-to-verdict QA over public regulatory rulebooks remains underexplored, and existing settings do not directly test explicit regulatory chain traversal with source-grounded intermediate-step reconstruction. Conventional conclusion-only grading admits *phantom credit*: a model can land the right verdict while its intermediate steps go unchecked. RegBench addresses this gap by sampling paths through rulebooks, turning them into practitioner-scope applicability questions, grounding the resulting clause-level facts in the source text, and testing the same construction across regulatory domains.

Benchmark validation and judging. Benchmark practice spans expert-authored evaluation, broad benchmark suites, and executable real-world tasks (Rein et al., 2023; Hendrycks et al., 2021; Srivastava et al., 2023; Liang et al., 2023; Mazumder et al., 2023; Jimenez et al., 2024; Mialon et al., 2023; Patwardhan et al., 2025). Judge-focused work further shows that LLM-based scoring for free-form outputs must itself be checked against objective correctness labels and consistency criteria (Tan et al., 2025; Lù et al., 2025; Wang et al., 2026b). These studies show that benchmark quality depends on scoring reliability, not just item difficulty. For regulatory source-to-verdict QA, existing protocols still under-specify trace-level validation, making derivation failures hard to localise. RegBench instead makes defects localisable at the anchor, hop, fact, or verdict level, rather than collapsing them into a single aggregate answer score.

3 THE REGBENCH METHOD

The RegBench framework is a reusable pipeline for constructing source-grounded QA benchmarks from regulatory corpora (Figure 2). This section defines the task object and framework components, then describes their instantiation on DNV Ship Rules and 12 CFR Part 217.

3.1 TASK FORMULATION

RegBench evaluates source-grounded regulatory reasoning: given a scenario and regulatory source context, a model must follow the relevant cross-reference chain and derive a numeric or categorical verdict.

A RegBench item is a tuple $(\textit{scenario}, \textit{anchor}, \textit{chain}, \textit{required_facts}, \textit{verdict})$:

- *scenario* — a self-contained natural-language situation (a ship design, a bank capital position, a transaction).
- *anchor* — the section in the regulatory corpus where reasoning starts.
- *chain* — an ordered list of source units, starting from *anchor*, that must be applied to derive the answer. The chain may remain within a short local clause, span multiple clauses in one section, or require broader cross-section / cross-Part context.
- *required_facts* — atomic facts the answer must contain. Each fact is a verifiable proposition against the verbatim source text.
- *verdict* — a numeric value or compliance decision derived from applying *chain* to *scenario*.

A model answer is **strict-correct** only if the grader verifies every proposition in *required_facts*. An answer that reaches the final verdict but omits an intermediate derivation fact is marked incorrect, since RegBench evaluates source-grounded derivation rather than conclusion-only accuracy. Strict accuracy therefore measures derivation completeness as well as reasoning: a correct but concise derivation that leaves an intermediate fact implicit is not credited.

Appendix H gives a concrete released-item record and maps its JSON fields to this tuple representation.

3.2 SOURCE GRAPH AND CONSTRUCTION PIPELINE

RegBench first maps each corpus to a directed section graph (Figure 2A). Ingestion is corpus-specific: DNV Ship Rules are converted from PDF to section-level markdown with MinerU (Wang et al., 2024), while 12 CFR Part 217 (Board of Governors of the Federal Reserve System, 2024) uses markdown prepared from the public eCFR text without PDF extraction. The shared graph interface normalises section identifiers (Pt.X.Ch.Y.Sec.Z for DNV; subpart-section-paragraph for Part 217), adds edges for explicit cross-references, and samples balanced chains stratified by required context scope. Each sampled chain fixes the intended reasoning path, defines the item's tier, and diversifies anchors across the corpus. Evaluated systems and audits receive only the markdown source (Appendix F).

Tier definition. We use a fixed context-scope tier schema: T0 = single-clause local derivation; T1 = single-section derivation requiring a table, formula, conditional branch, or multiple clauses; T2 = cross-section reasoning within the same chapter; T3 = cross-chapter reasoning within the same Part; and T4 = cross-Part or cross-volume reasoning. This schema is corpus-independent, but each corpus realizes only the tiers supported by its in-scope source: DNV uses T0–T4, while 12 CFR Part 217 uses T0–T3 because T4-style cross-Part chains leave the public Part 217 scope and are excluded. The tier labels are context-scope controls, not exact hop-count labels and not claims that same-numbered tiers have identical substantive difficulty across corpora.

For each sampled chain, an LLM generates a schema-constrained tuple from §3.1. The scenario must read like a practitioner compliance case: it specifies the relevant ship-design or bank-capital facts, names the decision to be checked, and withholds both the answer and the intended cross-reference path. We reject drafts that leak answer tokens, section identifiers, or path cues into the scenario, since such items would not reflect real compliance review. A chain-visible confidence check then removes uncertain, internally inconsistent, or cue-dependent drafts. The retained tier-diverse pool enters the v5d selective audit in §3.3. Pipeline yields are reported in Appendix D.

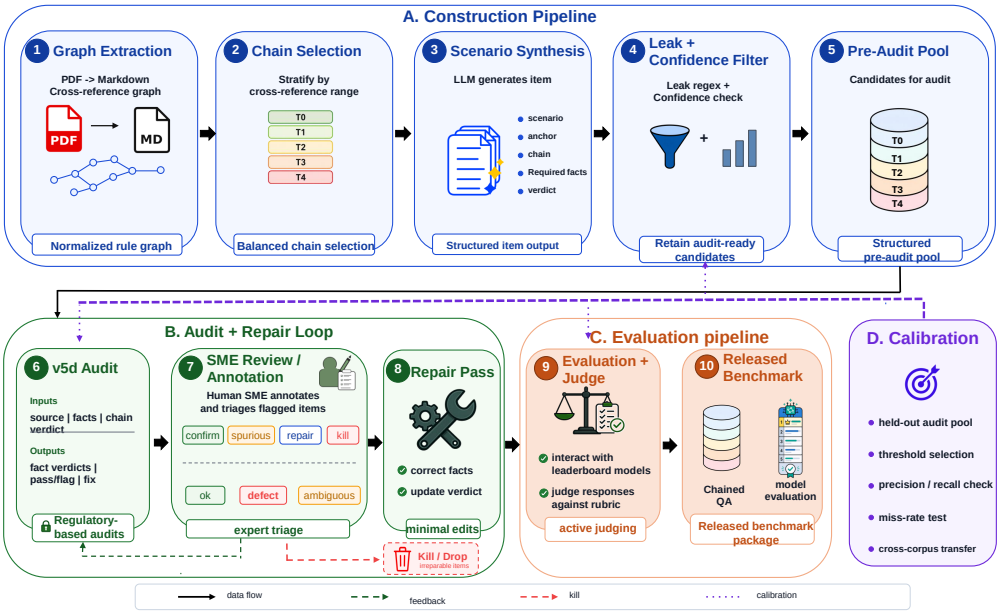

Figure 2: RegBench workflow. (A) The construction pipeline extracts a normalised rule graph, samples tier-balanced chains, synthesises structured draft items, filters leakage and unstable candidates, and forms the pre-audit pool. (B) The audit-and-repair loop applies v5d, routes flagged items to SME triage, repairs source-grounded labels when possible, and drops irreparable items. (C) The evaluation pipeline runs model evaluation and answer judging against the released labels. (D) Calibration estimates audit reliability against SME source-grounded review.

Practitioner-scope consultation (DNV). The DNV question-generation scope (eligible Parts/Chapters, practitioner-relevant chain ranges, scenario parameters such as vessel class, length, and structural element) was set in consultation with practising ship structural designers, anchored to an *independent* structural-design need scope they identified before any item was synthesised. The same scope governs both the question-generation pool and the practitioner-scope (`full_context`) evaluation path (§4.1), so the benchmark replicates real review usage rather than an LLM-internal notion of which clauses are interesting. Basel scenarios are similarly scoped to bank-capital decisions reviewed in practice but were not validated by an in-the-loop banking practitioner. We treat DNV as the realism anchor and Basel as construction-pipeline portability evidence (Appendix B).

3.3 V5D SELECTIVE AUDIT

v5d is the core quality-control mechanism in RegBench. It is a calibrated selective audit with three parts (Figure 4, Appendix E): an *evidence packet*, *locked defect rules*, and *selective routing*.

Evidence packet. For each generated candidate, v5d receives the sampled *chain*, the corresponding verbatim source clauses, and the item's *required_facts* (which encode the candidate gold answer). It decomposes the candidate label into per-fact checks and returns per-fact verdicts (*correct*, *incorrect*, *uncertain*, *hallucinated_citation*), an item-level disposition (*pass*, *flag*), and a fix recommendation (*no_fix_needed*, *fix_answer*, *fix_question*, *kill_question*).

Locked rules. The prompt reserves *incorrect* for substantive, verifiable contradictions: a required fact asserts a concrete binding element, the source contains the corresponding element, and the two materially disagree. Ambiguity is resolved conservatively: when unsure whether a fact is wrong, the auditor prefers *correct*, and an item is flagged only for a substantive *incorrect* fact or a *hallucinated_citation*, never for uncertainty alone. Three guardrails (E1–E3 here, T1–T3 in the prompt) target chart/table row-column mismatches, formula integrity, and default-vs-special-branch errors; each fires only when all its source checks hold, anchoring the audit to regulatory table coordinates, formula operands, and case-tree branches rather than DNV-specific syntax. The rules

Table 2: v5d–SME calibration on the headline slices. "Routed confirmed": precision on v5d-flagged items; "v5d missed": SME-confirmed defects v5d did not flag; item and fact agreement: v5d–SME match rates on the audited slice. Bold: the abstract's headline figure. Other slices and their aggregate: Table 5 (Appendix B).

Audit design	Pool	Items audited	Facts audited	v5d routed	Routed confirmed	v5d missed	Item agreement	Fact agreement
Full audit (pass + routed)	DNV TRAIN + TEST	200	1,117	38	100%	1	**99.0%**	**99.1%**
Routed + 20-pass sample	Part 217 audited slice	79	478	59	94.9%	0	96.2%	94.77%

were locked after SME-audited DNV TRAIN development and applied unchanged to DNV TEST and 12 CFR Part 217 (prompt, guardrails, and derivation in Appendices I, E, and M).

Selective routing. Items that pass v5d continue through the release pipeline. Flagged items are routed to SME triage for confirmation, repair, or removal. Per-scope verification routines and caught examples are detailed in Appendix E.

3.4 Audit and repair loop

The audit-and-repair loop turns v5d-screened candidates into released benchmark items (Figure 2B).

SME triage and repair. Every v5d-flagged item is source-grounded re-reviewed by an SME, who assigns an item disposition in {confirm-defect, spurious-flag, repair, kill} and per-fact labels in {OK, defect, ambiguous}. Items marked *repair* receive minimal edits to *required_facts* and/or *verdict*. Items marked *kill* are removed when the ground truth cannot be repaired against the source. Auditors see no model outputs and no leaderboard signal during review. v5d's per-fact reasoning is rendered only on items v5d flagged.

3.5 Release protocol and calibration summary

We release the post-audit-and-repair items with per-item v5d verdicts (item-level disposition and per-fact reads) and the construction, audit, and evaluation code, together with every contested SME label, the disposition ledger, and the SME-certified perturbed answer keys (Appendix J). Agreeing SME labels stay internal, so the audit slate is not exposed to future re-use (Figure 2B–C). Calibration (Figure 2D) estimates v5d's operating point on sampled slices, following sampled-validation practice; it does not certify each item of the 827-question / 4,766-fact release.

Because the DNV row audits passed and routed items alike, it estimates both agreement and the pass-set miss rate. On the held-out DNV TEST split, v5d misses **0/446** facts on passed items (rule-of-three (Hanley & Lippman-Hand, 1983) 95% upper bound $\leq 0.67\%$) and agrees with SME source-grounded re-review on 98.94% of facts. On the combined 200-item TRAIN+TEST pool it reaches **198/200 = 99.0%** item-level and **1107/1117 = 99.1%** fact-level agreement, with **97.4%** recall and **100%** precision on SME-flagged items.

Basel uses the locked prompt unchanged: its audited slice (all 59 v5d-routed items plus a 20-item v5d-pass sample) yields 76/79 = 96.2% item-level and 453/478 = 94.77% fact-level agreement, with **0/122** fact-level defects in the pass sample. We treat Basel as transfer evidence rather than a co-equal bounded-release-quality claim. The DNV pilot and selective-routing slices and the four-slice aggregate (97.9% item-level, 97.4% fact-level) are discussed with Table 5 in Appendix B.

3.6 Composition and statistics

The released promoted benchmark contains **827 questions spanning 4,766 atomic-fact propositions**, post-audit-and-repair under the §3.5 release pipeline (per-subset counts in Appendix D, Table 6).

DNV Ship Rules (546 released questions: 499 scale + 47 pilot, after 1 scale and 2 pilot items were killed; 3,104 facts post-repair) are 5-tier (T0–T4). 12 CFR Part 217 (281 questions, 1,662 facts

post-repair; 7 items killed) is 4-tier (T0–T3, cross-Part excluded by scope). After the v5d audit pass and repair, the post-audit-and-repair pool is released alongside per-item v5d audit verdicts. Every v5d-flagged item underwent SME source-grounded re-review under the protocol in Appendix B, with raw SME annotation labels and disagreement logs retained internally rather than redistributed. The tier schema (§3.1) is structural and shared across corpora. What differs is which level of each corpus's native hierarchy fills the schema's *chapter* slot (DNV Chapter; Basel Subpart of Part 217). Cross-corpus claims therefore concern the tier-degradation pattern, not absolute level. Appendix Q documents the per-corpus mapping with concrete chain examples. A representative item appears in Appendix G.

4 EXPERIMENTS

4.1 SETUP

Models evaluated. Sonnet 4.6 (Anthropic, 2026b), GPT-5.4, GPT-5.6-sol and GPT-5.6-terra (OpenAI, 2026a;b), Qwen3.6-Plus (Qwen Team, 2026), and MiniMax-M2.7 (MiniMax, 2026), each evaluated under **practitioner-scope** (`full_context`) — DNV anchor-Part expansion as identified by the consulting ship-design team (§3.2), capped at 600K characters; Basel = full Part 217. We additionally explore a **BM25-retrieval** condition (`bm25+reader`: BM25 (Robertson & Zaragoza, 2009) top-10 over the full corpus, fed to a Sonnet 4.6 reader) for **Sonnet 4.6 only**, as a probe of whether lexical retrieval can substitute for full-context loading. Closed-book runs are reported in Appendix P.

Evaluation pools. We use two complementary pools, both subsets of the released 827-Q / 4,766-fact benchmark. (i) The **scale pool** is the released DNV scale split (499-Q, excluding the 47-Q pilot) plus the full 281-Q Basel release, and is what the leaderboard (§4.2) and tier-degradation analysis (§4.3) report — seven evaluation methods $\times$ both corpora. (ii) The **pilot subset** is a 47-Q post-audit-and-repair stratified slice of the DNV release (`P50_` prefix, 10 items per tier, T4 has 7) carrying denser per-fact SME annotation; it is used in §4.3 to evaluate a broader 13-system panel that adds Anthropic's three-tier capability scale (Opus / Sonnet / Haiku) and three published graph-RAG variants (ArchRAG, HippoRAG, GraphRAG) which would be infeasible to run at scale. The pilot subset also anchors the SME-1 / SME-2 judge calibration described below and the cross-judge replay in Appendix O. All numbers under both pools come from the same Sonnet 4.6 primary judge; leaderboard answers are additionally scored by an independent cross-judge (below).

Prompts, decoding, and effort. Single-shot, no chain-of-thought; deterministic decoding (*temperature* $= 0$, *max_tokens* $= 16{,}384$); reasoning effort fixed at `medium` for every run except the v5d audit (`xhigh`) and GPT-5.6-terra (`high`). Prompt sources are listed in Appendix I and per-run efforts in Appendix J.

Grading. Atomic-fact strict conjunction: an answer is correct only if the LLM judge finds every required fact present. The primary judge is Sonnet 4.6. Because Sonnet 4.6 also generates the scenarios and is itself evaluated, every leaderboard answer is re-scored by an independent **cross-judge**, DeepSeek-v4-pro (DeepSeek-AI et al., 2026), which has no other role in the pipeline and is not evaluated; prompt and decoding are identical. Both judges are validated against expert annotation on *model outputs*. SME-1 labelled fact presence on **2,453** model-answer fact instances across 12 model–condition pairings; the primary judge agrees with SME-1 at **99.6%** (Wilson 95% CI 99.3%–99.8%, Gwet AC1 $= 0.994$ (Gwet, 2008)). A second independent annotator (SME-2), working from the same source and rubric on **2,350** labels, agrees with the primary judge at **97.5%** (AC1 $= 0.951$) and with SME-1 at **97.2%** (AC1 $= 0.954$) on the **1,341** shared labels, all in the "almost perfect" range of Landis & Koch (1977). The cross-judge agrees with SME-1 and SME-2 at 96.0% and 94.7%; its disagreements skew strict (about 3:1), so it scores every system lower, but it ranks them the same way (§4.4).

4.2 FRONTIER MODEL PERFORMANCE ON THE PROMOTED DNV RELEASE

Strict accuracy on the released DNV ($N = 499$) and 12 CFR Part 217 ($N = 281$) pools is in Table 3, under both judges. Frontier methods cluster in 36–57% under the primary judge, above

Table 3: Strict accuracy (%) on the released DNV ($N = 499$) and 12 CFR Part 217 ($N = 281$) pools under the primary judge (Sonnet 4.6) and the independent cross-judge (DeepSeek-v4-pro), ordered by DNV primary score. Wilson 95% half-widths are at most ±4.4pp (DNV) and ±5.8pp (Basel); per-cell intervals in Appendix J. `fc` = practitioner-scope `full_context`.

Model	**DNV** primary	**DNV** cross-judge	**Basel** primary	**Basel** cross-judge
GPT-5.6-sol `fc`	**49.3**	**43.7**	56.2	**52.7**
BM25+Sonnet 4.6	44.5	40.9	**56.9**	49.8
Sonnet 4.6 `fc`	44.1	41.5	52.0	47.0
GPT-5.4 `fc`	43.1	39.9	45.2	37.7
GPT-5.6-terra `fc`	40.5	36.7	49.5	43.4
Qwen3.6-Plus `fc`	35.9	32.7	35.6	28.8
MiniMax-M2.7 `fc`	16.6	15.0	17.1	16.7

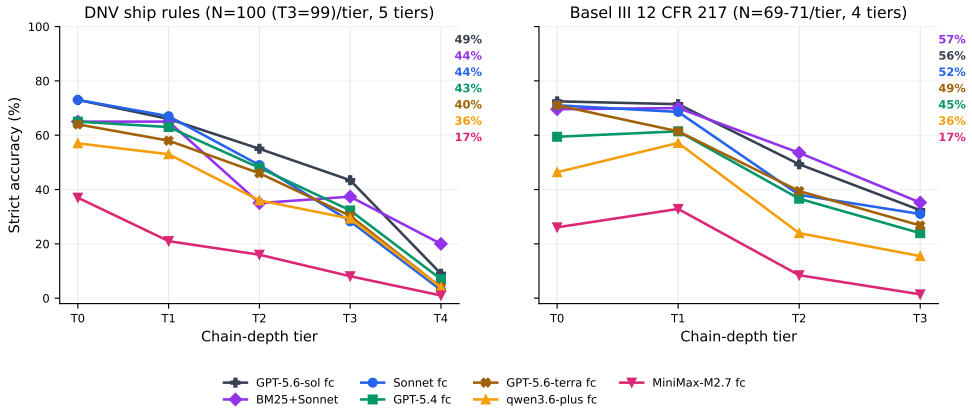

Figure 3: Strict accuracy (%) by chain-depth tier on the released DNV ($N = 499$) and 12 CFR Part 217 ($N = 281$) pools. Every method drops with tier. BM25+Sonnet overtakes Sonnet `full_context` at the hardest tiers. Per-tier denominators: DNV $\{100, 100, 100, 99, 100\}$; Basel $\{69, 70, 71, 71\}$. Right-margin labels: overall strict accuracy. `fc` = `full_context`.

MiniMax-M2.7's 17% floor. Paired McNemar (McNemar, 1947) on DNV: GPT-5.6-sol leads (49.3%) and is distinguishable from Sonnet `fc` ($p = 0.018$) and GPT-5.4 `fc` ($p < 0.001$), but not from BM25+Sonnet ($p = 0.058$); BM25+Sonnet, Sonnet `fc` and GPT-5.4 `fc` are pairwise indistinguishable ($p > 0.10$); GPT-5.6-terra is indistinguishable from all three ($p > 0.12$) and above Qwen3.6-Plus ($p = 0.023$); the leading group > Qwen3.6-Plus > MiniMax-M2.7 at $p < 0.001$. On Basel, GPT-5.6-sol and BM25+Sonnet are tied ($p = 0.92$) and the rest of the ordering recovers.

4.3 Stable degradation across reasoning tiers

Strict accuracy shows clear tier degradation on both corpora (Figure 3): every method drops ≥ 25pp from its T0 to its hardest tier (Sonnet `fc` 73→3% on DNV; 71→31% on Basel), with the largest tier-to-tier change in every row downward. Nine of the fourteen rows are strictly monotonic. The other five exhibit small T0→T1 (Basel) or T2→T3 (DNV) upticks (≤10.8pp) within Wilson-95% noise. Rankings shift across tiers: BM25+Sonnet trails Sonnet `full_context` at T0/T1 but overtakes it at the hardest DNV tiers (+9pp at T3, +13pp at T4) and at every Basel tier from T1 onwards, suggesting retrieval scales better than full-context as chain-depth grows. Basel `full_context` is the entire Part 217 (tier-invariant), so the pattern holds independently of DNV's variable context size.

A separate 13-system pilot panel (Appendix N) extends the comparison to Anthropic's capability scale and three graph-RAG variants (Wang et al., 2026a; Gutiérrez et al., 2024; Edge et al., 2024). The panel surfaces a ≈19pp median phantom-credit gap between conclusion-only and strict scoring, i.e. derivation detail that conclusion-only scoring does not require. The graph-RAG methods score 0–17% strict; two exceed the 14.9% MiniMax pilot score by 2.1pp.

4.4 Robustness to the model judge

We test whether the leaderboard is robust to the choice of model judge, and to the other models used in construction. The central result is that **the ranking is the same whichever model judges**: the cross-judge scores every system lower (by 0.4–7.5pp) yet preserves 40 of the 42 pairwise orderings across the two corpora (Kendall $\tau = 0.905$ on each), and both exceptions are pairs that are statistically tied under either judge (BM25+Sonnet vs Sonnet `fc` on DNV, $p > 0.10$; GPT-5.6-sol vs BM25+Sonnet on Basel, $p \geq 0.48$). The primary judge also shows no preference for its own model's answers: under it, a model from another vendor leads DNV (GPT-5.6-sol, $p = 0.018$ against Sonnet `fc`). The construction side is independent by design: chains are extracted mechanically with no LLM in chain selection, the v5d audit runs a different vendor's model (GPT-5.4) over every Sonnet-generated item, and SMEs verify flagged items against source. Three further checks find no generator self-preference (Sonnet 4.6 still leads on items generated by DeepSeek-v4-pro), no dependence on how answer keys are segmented, and no inflation of audit agreement from SMEs seeing v5d (97.3% item / 98.9% fact agreement where v5d output was hidden). Protocols and per-system results are in Appendix R (Table 15).

4.5 Beyond explicit citations: a civil-law pilot

The released corpora cite by address ("§217.100(b)(1)"); civil codes also refer implicitly, e.g. the term *lease* points to the provision that defines it. Implicit references differ from citations in form, not kind: both resolve against a document-derived index and become graph edges, so chain sampling, generation and the v5d audit are unchanged. We extended graph construction to four government-published English civil codes (6,630 provisions) with a corpus-specific deterministic parser, and two legal professionals verified the extracted edges on stratified samples (Appendix S, Table 18). Implicit references (any reference naming no address: defined terms, institutional terms, relative pointers) outnumber citations 1.7:1 to 4.5:1, and 1,236 of 2,189 two-hop article-to-article chains pass through an implicit edge, so a citation-only graph would miss most of each code's reference structure; resolution errors, concentrated in one sub-article descriptor class, create no false article-to-article edges. On 50 PRC Civil Code items from the unmodified pipeline and the locked v5d prompt, all reviewed by four practising lawyers and judges, v5d raised fact-level doubts on 9, and 35 of the other 41 (85%, 95% CI 72–93%) were judged usable as generated. Strict accuracy leaves headroom (Sonnet 4.6: 62% with the code in context, 46% closed-book; GPT-5.4: 38% / 30%), but a widely reproduced civil code is partly answerable from memory (DNV closed-book: 0%), so we treat it as evidence that the construction method carries over, not as a scored split (worked chains in Appendix S).

5 Conclusion

RegBench is the first public benchmark to evaluate the reasoning trace of regulatory cross-reference traversal over long-document contexts — anchor, chain, atomic facts, verdict — against verbatim source. Around this construct we build a graph-anchored generation pipeline and a calibrated semi-automated audit (v5d) that ports without retuning across DNV and 12 CFR Part 217, achieving 0/446 fact-level miss rate on passed items and 99.0%/99.1% item/fact agreement on the 200-item DNV pool, at ∼one-fifth per-item SME re-review cost. Frontier methods reach 17–57% strict with clear tier degradation, three published graph-RAGs collapse to $\leq 17\%$, and BM25+Sonnet retrieval leads at the hardest tiers. The ranking does not depend on the models in the loop: an independent judge reproduces it, and the construction method extends to implicit references in civil law. We discuss limitations in Appendix C; next, we will expand the benchmark's scale and breadth and augment v5d towards a fully automated annotation pipeline.

AI use statement

Generative AI models are documented components of the RegBench artefact and of its evaluation protocol rather than incidental tools: benchmark items are LLM-synthesised over sampled cross-reference chains (§3.2), the v5d selective audit is an LLM audit whose flags route items to expert repair or removal (§3.3–§3.4), and model answers are graded by an LLM judge (§4.1). Model identities, versions, decoding settings, reasoning effort, and verbatim prompts are given in §4.1, Appendix I, and Appendix J. In the terms of the ICLR 2027 AI Policy for Authors, we used generative

AI tools to *generate synthetic data sets* (item synthesis), *clean and reformat dataset* (the v5d audit and repair loop), *implement methods*, *design or provide feedback on research methodology or experiments*, *support qualitative and thematic data analysis*, and *interpret results*. We have not used generative AI tools to help develop theoretical models or conceptual frameworks, propose or refine hypotheses, or assist with translation; formulating mathematical claims, providing critical ingredients for proving them, and assisting in the writing of proofs are not applicable to this work. Additionally, we used generative AI tools to draft parts of the paper, edit the paper to improve readability, summarise and analyse existing literature, create or modify scientific figures, format references, and propose keywords. We have reviewed all AI-assisted work. Every LLM-synthesised item was released only after the calibrated v5d audit and source-grounded re-review by credentialed subject-matter experts described in §3.3–§3.5 (calibration in Table 2); the LLM judge was validated against expert labels on model outputs (§4.1); AI-assisted code, text, figures, and reference entries were reviewed and revised by the authors, and reported numbers were recomputed from the released artefact. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

Ethics statement

RegBench is built exclusively from regulatory text: 12 CFR Part 217 is taken from the public eCFR, and the DNV Ship Rules are freely available after DNV registration. The DNV source text is not redistributed; released items carry section identifiers so users can re-fetch and re-extract it themselves (licensing and hosting terms in Appendix L). Scenarios describe hypothetical ship designs and bank capital positions; the dataset contains no personal or sensitive information (Appendix K). No human-subjects research was conducted: expert annotation was source-grounded review of synthetic items by credentialed domain reviewers, who saw no model outputs and no leaderboard signal during item review (§3.4). All expert reviewers (the DNV ship-design SMEs and the finance-credentialed Basel reviewers) are members of the research team. The two legal professionals who verified the civil-law reference graphs and the four practising lawyers and judges who reviewed the PRC pilot items (§4.5) are external to the team. The benchmark is intended for evaluating source-grounded reasoning; it is not regulatory advice and should not be used to automate compliance decisions or to train compliance-decision systems, as recorded in the Croissant responsible-AI fields (Appendix L). The work received no external funding, and the authors declare no competing interests.

Reproducibility statement

The evaluated models, decoding settings, reasoning-effort levels, context conditions, and grading protocol are specified in §4.1, with the full prompt templates in Appendix I and a per-run reasoning-effort table in Appendix J. The released artefact (`https://github.com/regbench/regbench`; also in the supplementary material) contains the 827-item / 4,766-fact pool, the locked v5d audit prompt and driver (Appendix M), per-item v5d verdicts, the judge prompt and a stand-alone strict-conjunction grader, the construction and evaluation source code, the PDF-to-markdown ingestion driver (Appendix F), MLCommons Croissant metadata (Appendix L), and a manifest with file checksums and model-version metadata (Appendix J). For independent checking of the calibration and scoring claims it also contains the SME-vs-v5d and SME-vs-judge disagreement logs, the disposition ledger, per-fact judge verdicts for every evaluated system, and the SME-certified perturbed answer keys with a script that recomputes Table 16 (Appendix J). The GPT-5.6 runs follow the same protocol (§4.1); the cross-judge re-scoring of all leaderboard answers and the other robustness checks are described in §4.4 and Appendix R, the three-judge pilot replay in Appendix O, and the civil-law pilot protocol in Appendix S. The expert annotation protocol and all calibration statistics are given in §3.5 and Appendix B; a worked released item and its JSON field mapping appear in Appendices G and H; a datasheet is provided in Appendix K.

REFERENCES

Mubashara Akhtar, Omar Benjelloun, Costanza Conforti, Luca Foschini, Pieter Gijsbers, Joan Giner-Miguelez, Sujata Goswami, Nitisha Jain, Michalis Karamousadakis, Satyapriya Krishna, et al. Croissant: A metadata format for ML-ready datasets. *Advances in Neural Information Processing Systems*, 37:82133–82148, 2024.

Afra Feyza Akyürek, Advait Gosai, Chen Bo Calvin Zhang, Vipul Gupta, Jaehwan Jeong, Anisha Gunjal, Tahseen Rabbani, Maria Mazzone, David Randolph IV, Mohammad Mahmoudi Meymand, Gurshaan Chattha, Paula Rodriguez, Diego A. Mares Buendia, Pavit Singh, Michael Liu, Subodh Chawla, Peter Cline, Lucy Ogaz, Ernesto Gabriel Hernández Montoya, Zihao Wang, Pavi Bhatter, Marcos Ayestaran, Bing Liu, and Yunzhong He. PRBench: Large-scale expert rubrics for evaluating high-stakes professional reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 42297–42325, 2026. doi: 10.18653/v1/2026.acl-long.1958. URL https://aclanthology.org/2026.acl-long.1958/.

Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. L-Eval: Instituting standardized evaluation for long context language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14388–14411, 2024.

Anthropic. Claude Haiku 4.5 System Card, October 2025. URL https://assets.anthropic.com/m/99128ddd009bdcb/original/Claude-Haiku-4-5-System-Card.pdf.

Anthropic. Claude Opus 4.6 System Card, February 2026a. URL https://www.anthropic.com/claude-opus-4-6-system-card.

Anthropic. Claude Sonnet 4.6 System Card, February 2026b. URL https://www.anthropic.com/claude-sonnet-4-6-system-card.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. LongBench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 3119–3137, 2024.

Basel Committee on Banking Supervision. Basel III: A global regulatory framework for more resilient banks and banking systems. Technical report, Bank for International Settlements, Basel, Switzerland, December 2010. URL https://www.bis.org/publ/bcbs189.htm.

Board of Governors of the Federal Reserve System. 12 CFR Part 217: Capital adequacy of bank holding companies, savings and loan holding companies, and state member banks (Regulation Q). U.S. Government Publishing Office, eCFR, 2024. URL https://www.ecfr.gov/current/title-12/chapter-II/subchapter-A/part-217.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling

Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. DeepSeek-V4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

DNV. DNV rules for classification of ships. DNV Group AS, 2025. Available via the DNV Rules and Standards Explorer at https://standards.dnv.com/explorer/RU-SHIP (free DNV registration required).

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph RAG approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92, 2021.

Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. LegalBench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in Neural Information Processing Systems*, 36:44123–44279, 2023.

Bernal J Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. *Advances in Neural Information Processing Systems*, 37:59532–59569, 2024.

Kilem Li Gwet. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48, 2008.

Hongyong Han, Wei Wang, Gaowei Zhang, Mingjie Li, and Yi Wang. CoralVQA: A large-scale visual question answering dataset for coral reef image understanding. *Advances in Neural Information Processing Systems*, 38, 2026.

James A Hanley and Abby Lippman-Hand. If nothing goes wrong, is everything all right?: interpreting zero numerators. *JAMA*, 249(13):1743–1745, 1983.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the real context size of your long-context language models?, 2024. URL https://arxiv.org/abs/2404.06654.

Yulong Hui, Yao Lu, and Huanchen Zhang. UDA: A benchmark suite for retrieval augmented generation in real-world document analysis. *Advances in Neural Information Processing Systems*, 37:67200–67217, 2024.

Lanlan Ji, Dominic Seyler, Gunkirat Kaur, Manjunath Hegde, Koustuv Dasgupta, and Bing Xiang. PHANTOM: A benchmark for hallucination detection in financial long-context QA. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2026. URL https://openreview.net/forum?id=5YQAo0S3Hm.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues?, 2024. URL https://arxiv.org/abs/2310.06770.

J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. *Biometrics*, pp. 159–174, 1977.

Jinu Lee, Kyoung-Woon On, Sophia Simeng Han, Arman Cohan, and Julia Hockenmaier. Evaluating legal reasoning traces with legal issue tree rubrics. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3299–3322, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.150. URL https://aclanthology.org/2026.acl-long.150/.

Ruosen Li, Zimu Wang, Son Q Tran, Lei Xia, and Xinya Du. MEQA: A benchmark for multi-hop event-centric question answering with explanations. *Advances in Neural Information Processing Systems*, 37:126835–126862, 2024.

Yifan Li, Yuhang Chen, Anh Dao, Lichi Li, Zhongyi Cai, Zhen Tan, Tianlong Chen, and Yu Kong. IndustryEQA: Pushing the frontiers of embodied question answering in industrial scenarios, 2025. URL https://arxiv.org/abs/2505.20640.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models, 2023. URL https://arxiv.org/abs/2211.09110.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

Xing Han Lù, Amirhossein Kazemnejad, Nicholas Meade, Arkil Patel, Dongchan Shin, Alejandra Zambrano, Karolina Stanczak, Peter Shaw, Christopher Pal, and Siva Reddy. AgentRewardBench: Evaluating automatic evaluations of web agent trajectories. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=fQcUZMPIvu.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, et al. MMLongBench-Doc: Benchmarking long-context document understanding with visualizations. *Advances in Neural Information Processing Systems*, 37: 95963–96010, 2024.

Mark Mazumder, Colby Banbury, Xiaozhe Yao, Bojan Karlaš, William Gaviria Rojas, Sudnya Diamos, Greg Diamos, Lynn He, Alicia Parrish, Hannah Rose Kirk, et al. DataPerf: Benchmarks for data-centric AI development. *Advances in Neural Information Processing Systems*, 36:5320–5347, 2023.

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

Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: a benchmark for general AI assistants, 2023. URL https://arxiv.org/abs/2311.12983.

MiniMax. MiniMax M2.7: Early echoes of self-evolution, March 2026. URL https://www.minimax.io/news/minimax-m27-en.

OpenAI. GPT-5.4 Thinking System Card, March 2026a. URL https://openai.com/index/gpt-5-4-thinking-system-card/.

OpenAI. GPT-5.6 System Card, July 2026b. URL https://deploymentsafety.openai.com/gpt-5-6.

Tejal Patwardhan, Rachel Dias, Elizabeth Proehl, Grace Kim, Michele Wang, Olivia Watkins, Simón Posada Fishman, Marwan Aljubeh, Phoebe Thacker, Laurance Fauconnet, et al. GDPval: Evaluating AI model performance on real-world economically valuable tasks. *arXiv preprint arXiv:2510.04374*, 2025.

Qwen Team. Qwen3-Max: Just scale it, September 2025. URL https://qwen.ai/blog?from=research.latest-advancements-list&id=241398b9cd6353de490b0f82806c7848c5d2777d.

Qwen Team. Qwen3.6-Plus: Towards real world agents, April 2026. URL https://qwen.ai/blog?id=qwen3.6.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. GPQA: A graduate-level Google-proof Q&A benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 4(1-2):1–174, 2009. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

Yuzhen Shi, Huanghai Liu, Yiran Hu, Gaojie Song, Xinran Xu, Yubo Ma, Tianyi Tang, Li Zhang, Qingjing Chen, Di Feng, Wenbo Lv, Weiheng Wu, Kexin Yang, Sen Yang, Wei Wang, Rongyao Shi, Yuanyang Qiu, Yuemeng Qi, Jingwen Zhang, Xiaoyu Sui, Yifan Chen, Yi Zhang, An Yang, Bowen Yu, Dayiheng Liu, Junyang Lin, Weixing Shen, Bing Zhao, Charles L. A. Clarke, and Hu Wei. PLawBench: A rubric-based benchmark for evaluating LLMs in real-world legal practice. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10067–10116, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.458. URL https://aclanthology.org/2026.acl-long.458/.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023.

Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. ProofWriter: Generating implications, proofs, and abductive statements over natural language. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 3621–3634, 2021.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Y. Tang, Alejandro Cuadron, Chenguang Wang, Raluca Ada Popa, and Ion Stoica. JudgeBench: A benchmark for evaluating LLM-based judges, 2025. URL https://arxiv.org/abs/2410.12784.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

U.S. Federal Aviation Administration. Title 14 of the Code of Federal Regulations: Aeronautics and space. U.S. Government Publishing Office, eCFR, 2024. URL https://www.ecfr.gov/current/title-14.

U.S. Food and Drug Administration. Title 21 of the Code of Federal Regulations: Food and drugs. U.S. Government Publishing Office, eCFR, 2024. URL https://www.ecfr.gov/current/title-21.

U.S. House of Representatives, Office of the Law Revision Counsel. United States Code. U.S. Government Publishing Office, 2024. URL https://uscode.house.gov/.

U.S. Nuclear Regulatory Commission. Title 10 of the Code of Federal Regulations: Energy. U.S. Government Publishing Office, eCFR, 2024. URL https://www.ecfr.gov/current/title-10.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, et al. MinerU: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*, 2024.

Shu Wang, Yixiang Fang, Yingli Zhou, Xilin Liu, and Yuchi Ma. ArchRAG: Attributed community-based hierarchical retrieval-augmented generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(19):15868–15876, 2026a. doi: 10.1609/aaai.v40i19.38619.

Yidong Wang, Yunze Song, Tingyuan Zhu, Xuanwang Zhang, Zhuohao Yu, Hao Chen, Chiyu Song, Qiufeng Wang, Zhen Wu, Xinyu Dai, Yue Zhang, Cunxiang Wang, Wei Ye, and Shikun Zhang. TrustJudge: Inconsistencies of LLM-as-a-judge and how to alleviate them. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=4uPyOCeN6U.

Edwin B. Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927.

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

Tianchi Xie, Minzhi Lin, Mengchen Liu, Yilin Ye, Changjian Chen, and Shixia Liu. InfoChartQA: A benchmark for multimodal question answering on infographic charts. *Advances in Neural Information Processing Systems*, 38, 2026. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/c7bd72f33e4b0ce16d2bec720d1e1597-Abstract-Datasets_and_Benchmarks_Track.html.

Xiaoyue Xu, Qinyuan Ye, and Xiang Ren. Stress-testing long-context language models with lifelong ICL and task haystack. *Advances in Neural Information Processing Systems*, 37:15801–15840, 2024.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2369–2380, 2018.

A Related-work comparison rationale

Table 4 gives the row-level rationale for Table 1. The six marks in each row follow the Table 1 column order: long-document / real-source input, professional or regulated domain, source-defined regulatory `xref` chain, scenario-to-verdict reasoning, audited intermediate-fact keys, and calibrated LLM audit / judge.

Benchmark	Marks	Rationale
HotpotQA (Yang et al., 2018)	○ ✗ ✗ ✗ ○ ✗	Wikipedia multi-hop QA with sentence-level supporting facts, adjacent to real-source QA and trace supervision. It is not a professional regulatory benchmark, does not construct items from regulatory cross-reference chains, and does not evaluate source-to-verdict compliance reasoning or calibrated LLM audit.
RULER (Hsieh et al., 2024)	○ ✗ ✗ ✗ ✗ ✗	A synthetic long-context stress test with controllable sequence length and task complexity. It probes long-context behaviour, but not real regulatory sources, professional-domain rule application, source-defined cross-reference traversal, trace validation, or audit.
UDA (Hui et al., 2024)	✓ ○ ✗ ○ ○ ✗	A real-world document-analysis and RAG benchmark with expert QA over raw documents, including adjacent professional domains. Its questions can require complex document grounding, but the construction is not based on regulatory `xref` chains and does not enforce RegBench-style atomic-fact trace validation or calibrated audit.
MMLongBench-Doc (Ma et al., 2024)	✓ ✗ ✗ ○ ○ ✗	A real long-PDF document-understanding benchmark with expert-labelled questions and cross-page evidence. It is adjacent to scenario reasoning and evidence validation, but it is not a professional or regulated-domain benchmark, and it does not target regulatory cross-reference chains or calibrated audit.
LegalBench (Guha et al., 2023)	○ ✓ ✗ ○ ○ ✗	A legal-reasoning benchmark authored by legal experts, including rule application and legal judgement tasks. It partially overlaps with source-grounded legal reasoning, but items are not generated from a public regulatory rulebook's explicit `xref` graph and are not graded by strict required-fact reconstruction.
FinBen (Xie et al., 2024)	○ ✓ ✗ ○ ○ ✗	A broad financial benchmark that includes a regulations-QA subset and other financial tasks. It is professional-domain and partially adjacent to regulatory QA, but it is not a chain-anchored source-to-verdict benchmark and does not use source-defined regulatory `xref` traversal or atomic-fact conjunction.
PHANTOM (Ji et al., 2026)	✓ ✓ ✗ ○ ○ ✗	A financial long-context QA hallucination-detection benchmark over source documents. It explicitly targets professional-domain source faithfulness, but its construct is faithful-vs-hallucinated QA rather than regulatory rule-chain application with clause-level required facts.
IndustryEQA (Li et al., 2025)	○ ✓ ✗ ○ ○ ✗	An embodied QA benchmark for safety-critical industrial scenarios. It is professional-domain and adjacent to scenario-level safety reasoning, but its evidence is visual and embodied rather than regulatory source text; it does not define a regulatory `xref` chain or source-fact audit protocol.
LEGIT (Lee et al., 2026)	✗ ✓ ✗ ✓ ✓ ✓	Legal judgment prediction with reasoning traces scored against issue-tree rubrics derived from court decisions. It supplies expert intermediate keys and validates its LLM judge against two groups of licensed lawyers (Krippendorff's α up to 0.74), but the input is case facts without the statutory text, items are not built from a rulebook's cross-reference graph, and scoring awards partial credit (about half of the points for the final order).
PRBench (Akyürek et al., 2026)	○ ✓ ✗ ○ ○ ✓	Open-ended professional deliverables (legal and finance) graded with large expert-written rubrics and weighted partial credit. Its LLM judge is selected by agreement with two experts on 101 tasks (Cohen's $\kappa \approx 0.60$, on par with expert–expert agreement). Reference texts are prepended for only a subset of tasks, rubrics mix required facts with instruction-following, practical-utility, and uncertainty-handling criteria, and no item requires traversing explicit cross-references through a codified rulebook.
PLawBench (Shi et al., 2026)	○ ✓ ✗ ✓ ✓ ✓	Rubric-scored simulation of real-world legal-practice workflows. Most items (750 of 850) are case analyses from facts to a legal conclusion, with expert reference answers that list the required case facts, reasoning steps, and statutes, and its LLM judge is selected by correlation with three legal experts. Its inputs are short task materials adapted from real cases rather than a statutory corpus to be traversed, and rubric scoring is weighted rather than a conjunction over source-derived intermediate facts.
JudgeBench (Tan et al., 2025)	✗ ✗ ✗ ✗ ○ ✓	A benchmark for evaluating LLM-based judges using objectively labelled answer pairs. It directly targets judge reliability and is adjacent to factual correctness checking, but it is not a document, professional-regulatory, cross-reference, or source-to-verdict benchmark.
RegBench	✓ ✓ ✓ ✓ ✓ ✓	RegBench explicitly uses public regulatory source text, professional compliance scenarios, source-defined `xref` chains, scenario-to-verdict reasoning, strict required-fact reconstruction, and a calibrated LLM audit calibrated against SME source-grounded re-review.

Table 4: Rationale for the comparison marks in Table 1. Each mark tuple follows the column order of Table 1; ✓ indicates an explicit construction/evaluation target, ○ indicates partial or adjacent coverage, and ✗ indicates that the property is not explicit.

B AUDIT CALIBRATION PROTOCOL AND DETAILED RESULTS

Calibration measures the v5d operating point rather than adding another construction step. We report calibration in two registers: (1) a *primary* fact-level-blind miss rate on v5d-passed items, and (2) *anchored* item-level and fact-level agreement on the full audit pool, where SMEs see v5d's item-level disposition as an advisory hint and v5d's cited clauses are visible during disputed-item re-review. The two registers carry different evidential weight; we keep them separate.

Table 5: v5d–SME calibration on all four disjoint audit slices (full version of Table 2). "Routed confirmed" is precision on the v5d-flagged subset; "v5d missed" counts SME-confirmed defects v5d failed to flag; "Item agreement" and "Fact agreement" are v5d–SME match rates on the audited slice. The bolded 99.0% / 99.1% on DNV TRAIN+TEST is the abstract's headline calibration figure. The Aggregate row sums numerators and denominators across body rows; dashes mark quantities not available for that audit design.

Audit design	Pool	Items audited	Facts audited	v5d routed	Routed confirmed	v5d missed	Item agreement	Fact agreement
Full audit (pass + routed)	DNV pilot	49	277	9	77.8%	1	93.9%	98.2%
Full audit (pass + routed)	DNV TRAIN + TEST	200	1,117	38	100%	1	**99.0%**	**99.1%**
Routed + 20-pass sample	Part 217 audited slice	79	478	59	94.9%	0	96.2%	94.77%
Selective routing only	DNV selective audit split	46	271	46	100%	–	100%	94.5%
Aggregate	union of audit slices	374	2,143	152	96.7%	2	97.9%	97.4%

DNV protocol. Calibration uses a held-out 100-question DNV TEST split, fully SME-audited end-to-end. SME-1 (session A) and SME-3 (session B), two of the three DNV ship-design SMEs, independently annotate every item (60 each with a 20-item shared overlap for inter-annotator agreement). The annotation interface shows v5d's item-level verdict as an advisory hint; v5d's per-fact reasoning is rendered only on flagged items, so per-fact judgements on v5d-passed items are independent of audit flagging at the fact level. The combined DNV audit pool spans 200 items: this 100-Q TEST split plus a 100-Q SME-3 development subset on which the rules were originally derived. For the development subset, a pre-rejudge canonical audit was followed by a fresh SME re-review of v5d/canonical disagreements with v5d's cited source clauses visible. The cycle is evidence-exposed but not deferential: on an SME-1 pilot disputed slice, SMEs overruled v5d-suspect verdicts after checking the source and independently flagged additional defects among facts v5d had labelled correct.

Primary miss-rate estimate. On the held-out DNV TEST split, v5d's miss rate on the 446 facts of v5d-passed items is **0/446** (rule-of-three (Hanley & Lippman-Hand, 1983) 95% upper bound $\leq 0.67\%$). On the reviewer-A v5d-passed Basel sample, the miss rate is **0/122** fact-level defects (Basel-specific rule-of-three $\leq 2.5\%$). On DNV, per-fact judgements on v5d-passed items are made without v5d's per-fact reasoning visible (interface contract), so the per-fact-blind property is structural, not promised; the Basel review interface showed v5d's per-fact verdicts and reasons on every fact, so the Basel figure is not per-fact blind (Table 17).

Anchored DNV agreement. On TEST, v5d's per-fact reads vs SME source-grounded re-review agree on 561/567 = 98.94% facts; inter-annotator $\kappa = 0.77$ item / 0.905 fact on the 20-item shared overlap. Across the combined 200-item DNV audit pool, item-level agreement with final SME annotation is **198/200 = 99.0%**, fact-level agreement is **1107/1117 = 99.1%**, recall **38/39 = 97.4%**, precision **38/38 = 100%**. The disputed-slice re-review is empirically non-deferential: on an SME-1 pilot disputed slice, SMEs overruled three of v5d's nine suspect verdicts after consulting the cited clauses and independently flagged six additional defects among facts v5d had labelled correct (pre-rejudge snapshot detail in Appendix D).

Cross-corpus protocol (12 CFR Part 217). The locked v5d prompt is applied to the 288-item Basel pool at the same matched pipeline as DNV TEST (xhigh reasoning effort, broad source loading). Four finance-credentialed reviewers (A–D) audit the Basel v5d outputs. The v5d-flagged set is redistributed across reviewers B–D at 23/22/22 slots, with four shared items (one per tier, T0–T3) rated by all three for inter-annotator agreement. A separate 20-item stratified sample (5 per tier, T0/T1/T2/T3) of v5d-passed items is audited by reviewer A, paralleling the DNV TEST v5d-pass-set audit, to extend the v5d-pass audit onto Basel.

Basel cross-corpus port. We frame Basel as evidence that the audit ports to a second regulatory corpus without retuning, not a co-equal bounded-release-quality claim: the Basel evidence is thinner than the DNV evidence and we do not pool the pass-set miss rates as a single corpus-agnostic bound. v5d flags 59/288 Basel items at a tier-monotonic rate; SME re-review by reviewers B–D confirms **56/59 = 94.9%** as fact-level defective (49 repair, 7 kill, 3 spurious flags). On the audited slice (59 v5d-flagged + 20 v5d-passed), item / fact agreement with v5d is **76/79 = 96.2% / 453/478 = 94.77%**. B/C/D inter-annotator stability on the 4-item shared overlap is Gwet's AC1 = 0.54 (small-n stability stat, not transfer evidence: C↔D 4/4, B↔C and B↔D 2/4 each — B more lenient than the C/D consensus).

Secondary slices and aggregate. The selective-routing slice (DNV selective audit split) reports agreement on the routed subset only — v5d-passed items are not exhaustively SME-reviewed — so its 100% item agreement and 94.5% (256/271) fact agreement are computed over a slice that is enriched for defects by construction and are not directly comparable to the full-audit rows (DNV pilot and DNV TRAIN+TEST). The 15 fact-level mismatches there all go in one direction (v5d marked the fact `correct`/`uncertain`, SME marked it `defect`/`ambiguous`), reflecting v5d's design conservatism: even within items v5d flags, individual facts can still be under-flagged. The Aggregate row of Table 5 sums numerators and denominators across the four disjoint audit slices (366/374 = 97.9% item-level; 2,088/2,143 = 97.4% fact-level). We report this as a supplementary cross-corpus count.

Cross-vendor protocol (pilot-scope). A vendor-swap of the v5d audit on the 49-question dense pilot replays the locked v5d prompt with Claude Sonnet against the GPT-5.4-xhigh primary. Sonnet runs at `medium` reasoning effort (deterministic decoding, *temperature* $= 0$). The asymmetry — primary at `xhigh`, secondary at `medium` — is a deliberately weaker secondary-compute regime. The replay tests whether v5d's flag-set survives a weaker compute regime under a different vendor, not whether the two engines are equivalent on defect-detection sensitivity. The replay agrees on 43/49 dispositions (Cohen's $\kappa = 0.45$); all 6 disagreements are GPT-5.4-flag / Sonnet-pass, so Sonnet's flag-set is a subset of the primary's — consistent with compute-dependent sensitivity, not vendor idiosyncrasy.

C LIMITATIONS

Benchmark, not solver. RegBench is positioned as a benchmark, not a method that closes the regulatory-reasoning gap. The released gap (17–49% strict on DNV, 17–57% on Basel, clear tier degradation across all evaluated systems) is the open problem the benchmark exposes; v5d is a calibrated quality gate for benchmark construction, not a deployment-grade reasoning system.

Construct scope. RegBench operationalises a deliberately narrow construct: traversal of explicit cross-references in verbatim source text. The civil-law pilot (§4.5) shows the construction extends to implicit, term-based references, but the released items contain none. It does not test implicit obligations, open-textured interpretation, versioning, or adversarial ambiguity, and the audit removes exception-heavy and table-arithmetic items. The two corpora cover marine engineering and financial regulation; transfer to pharmaceutical, aviation, tax, or non-English regimes is not demonstrated. The construction pipeline and v5d prompt apply unchanged across corpora, so broader coverage is bounded by audit budget rather than methodological rewrite.

Scale and transfer. The release contains 827 questions / 4,766 atomic-fact propositions, which is modest relative to concept-recall benchmarks. Source-grounded chain construction with SME audit

is expert-intensive per item, and we traded scale for verified quality. Per-tier counts (100/tier $\times$ 5 tiers DNV) are adequate for the degradation analyses in §4.3/§4.3; cross-corpus comparison rests on pattern, not absolute level. The Basel evidence in §3.5 is meaningfully thinner than DNV (smaller audited slice, smaller v5d-pass audit, small inter-annotator overlap) and is treated as evidence that the audit ports to a second corpus, not as a co-equal bounded-release-quality measurement.

D CONSTRUCTION YIELD AND PRE-AUDIT BASELINE

Table 6: Released RegBench composition. Item and atomic-fact counts per subset, with the subset's role in the paper. Items dropped during SME re-review are noted in the role column. The release totals 827 items / 4,766 atomic-fact propositions.

Subset	Items	Atomic facts	Role
DNV scale (`R500_/F500_`)	499	2,835	scale evaluation, audit calibration (TRAIN/TEST splits); 1 item (F500_0129) killed during SME re-review
DNV dense pilot (`P50_`)	47	269	dense judge-calibration subset (12 model–cond. pairings); 2 items killed during SME re-review
12 CFR Part 217	281	1,662	cross-corpus transfer demonstration; 7 items killed during Basel audit
Total	**827**	**4,766**	

DNV Ship Rules. Graph extraction from the DNV PDFs yields 182,582 cross-reference chains for the T2–T4 tiers; T0/T1 candidates are sampled from section-local sources. Of 681 generated items, the confidence gate rejects 169 (24.8%), leaving 512; the leak regex rejects 1 further item (`F500_0151`) that the gate had passed, leaving 511; we trim 11 items (2.2%) to balance per-tier counts at 100 each, yielding a 500-item pre-audit pool ($681 - 169 - 1 - 11 = 500$). The leak regex's overall rejection rate on generated items is 0.15%.

From audit to release. Table 7 gives the v5d flags on each audited slice and how SMEs disposed of them. On the full DNV pool v5d flags 84 of the 500 scale items (16.8%) and 9 of the 49 pilot items; on Basel it flags 59 of 288 (20.5%). Twelve Basel candidates were removed before the audit (300 $\rightarrow$ 288). Across the 837-item pre-audit pool (500 DNV scale, 49 pilot, 288 Basel) SMEs removed 10 items as non-repairable (1 DNV scale item, `F500_0129`; 2 pilot items; 7 Basel items), leaving the 827 released (499 + 47 + 281). A typical E1 repair: in `R500_0036` a required fact read $C_{YR} = -1$ for load case BSR-1P, whereas DNV Pt3.Ch4.Sec2 Table 11 gives $C_{YR} = 1$; the -1 belongs to the adjacent C_{YG} row. The SME corrected the fact against the table and the item was released.

Table 7: v5d flags and SME dispositions on the audited slices. A DNV flag counts as confirmed if either TEST annotator marked the item defective; requiring both gives 37/38 (the annotators split on `R500_0433`).

Audited slice	Items	v5d-flagged	Confirmed defective	Repaired	Removed	Spurious
DNV TRAIN + TEST	200	38	38/38 (100%)	—	—	0
12 CFR Part 217 (locked prompt)	288	59 (20.5%)	56/59 (94.9%)	49	7	3

Pre-audit calibration baseline. To anchor the v5d calibration row in Table 2, an SME (SME-3) conducts a blinded source-backed audit on a 100-question stratified development subset of the pre-audit pool. On this subset the pre-audit measurements are 526/564 = 93.3% fact-level GT accuracy [Wilson 95% CI 90.9–95.1% (Wilson, 1927)]; 88/100 strict-correct; 12 items with at least one fact-level fail flag. These are the inputs the audit operates on.

Pre-rejudge snapshot of the combined DNV audit pool. A pre-rejudge snapshot of v5d-vs-canonical-SME on the combined 200-item DNV audit pool gives item-level agreement $191/200 = 95.5\%$, recall $31/33 = 93.9\%$, precision $31/38 = 81.6\%$. Source-grounded re-review of the nine snapshot disagreements confirmed seven v5d-flag/SME-pass items as fact-level defective — defects

the initial audit had missed and v5d had caught — which raises the post-rejudge agreement reported in §3.4 to $198/200 = 99.0\%$. The two residual disagreements are R500_0129 (v5d-pass / SME-flag, OCR-damaged source — a confirmed v5d miss) and F500_0051 (SME-3 noted partial topical overlap; all six facts SME-pass and v5d-pass, excluded from the recall denominator). The snapshot characterises initial canonical-audit error rather than v5d shortfall.

E V5D FAILURE-MODE GUARDRAILS (E1, E2, E3)

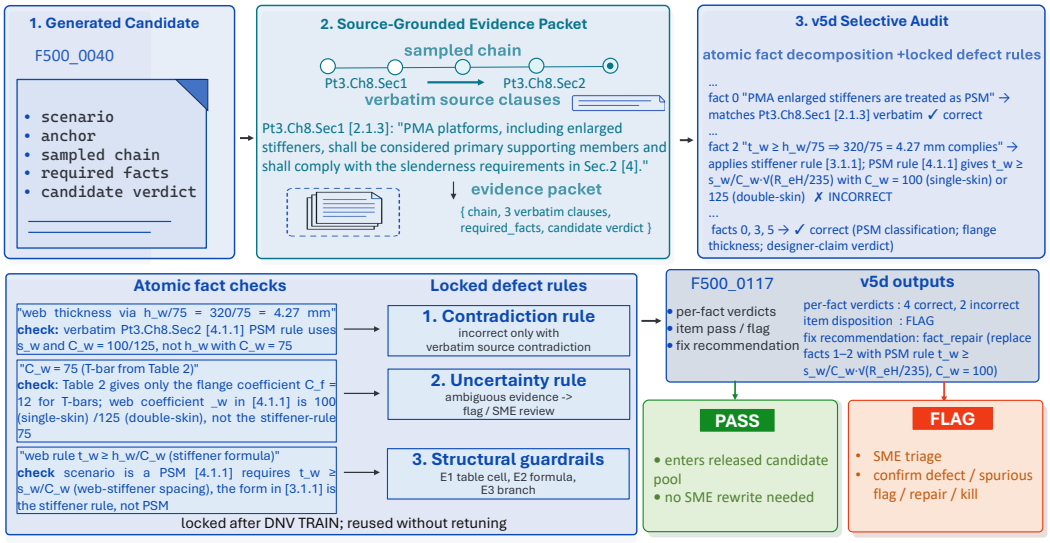

Figure 4: v5d selective audit design. v5d assembles chain-selected source evidence for each generated candidate, decomposes the gold label into atomic fact checks, applies locked source-grounded defect rules, and selectively routes flagged items to SME triage and repair before release.

The locked v5d audit prompt (§3.3) carries three failure-mode guardrails (E1: chart / table row-column mismatch; E2: formula integrity; E3: default-vs-special branch error). Table 8 documents each scope, its verification routine, and a caught example from the released DNV pool.

Failure scope	What can go wrong	Verification routine	Caught example
E1. Chart / table row-column mismatch	A required fact reads a value from a regulatory table but tracks the wrong row or column for the scenario (wrong load case, wrong ship-type notation, wrong material grade).	Re-derive table coordinates from the scenario parameters; compare the value at the correct *(row, column)* with the value claimed in the fact.	`R500_0036` (Pt3.Ch4.Sec2, T0): fact reads $C_{YR}{=}-1$ from Table 11 for the BSR-1P fatigue load case; the source has $C_{YR}{=}1$ for BSR-1P — the -1 value belongs to the separate C_{YG} row (gravity component), not C_{YR}.
E2. Formula integrity	A required fact asserts a derivation that violates the rule's formula structure: wrong operand, wrong coefficient source, wrong rule branch substituted into a formula slot.	Symbolically reconstruct the derivation from the source formula; compare operands and coefficient inputs against the fact's expression.	`F500_0040` (Pt3.Ch8.Sec1, T2): for a PMA enlarged stiffener treated as a primary supporting member, the fact uses the stiffener web rule $t_w \geq h_w/75 \cdot \sqrt{R_{eH}/235}$; the source's PSM web rule is $t_w \geq s_w/C_w \cdot \sqrt{R_{eH}/235}$ with $C_w{=}100$ (or 125 for double skin), not the stiffener coefficient.
E3. Default-vs-special branch error	A required fact applies the wrong branch of a default / special-case case tree, where the scenario unambiguously selects the other branch and the branches yield materially different answers.	Walk the rule's case tree; check that the scenario unambiguously selects the cited branch. If the scenario leaves the branch choice ambiguous, note the authoring ambiguity rather than treating it as a fact defect.	`F500_0137` (Pt5.Ch2.Sec8, T4): for a container ship, the fact applies the generic harbour-bending factor $f_{har}{=}0.5$ from Pt3.Ch5.Sec2; the source explicitly says "unless otherwise defined in Pt.5," and Pt5.Ch2.Sec3 [2.3.1] defines $f_{har\text{-}M}{=}0.9$ for container ships — the special branch reverses the compliance result.

Table 8: The three failure-mode guardrails of the v5d audit rule-set (§3.3). Each scope is anchored to a structural primitive of regulatory source text. The same three rules apply unchanged to 12 CFR Part 217 because the same primitives appear in the Basel corpus: regulatory tables (e.g., risk-weight grids), regulatory formulas (e.g., capital ratios), and regulatory case structures (e.g., method-1 / method-2 alternatives).

Worked example: conditional rules. Regulatory clauses are often piecewise ("if ..., otherwise if ..."). Graph nodes are sections, not branches, so a piecewise clause needs no special handling in the graph: each item's scenario pins the parameters that select one branch, and selecting that branch is itself a required fact. DNV Pt3.Ch12.Sec10 [2.1.1]–[2.1.2] sets the minimum freeing-port area for a bulwark well with four conditions: $A = 0.7 + 0.035\,\ell_b$ for $\ell_b \leq 20$ m and $A = 0.07\,\ell_b$ for $\ell_b > 20$ m; an increase of 0.004 m^2 per metre of well per 0.1 m of average bulwark height above 1.2 m (and a permitted decrease below 0.9 m); a 50% increase for ships with no sheer (linear interpolation for sheer below standard); and half the area for wells on superstructure decks. Released item `R500_0006` (T0) describes a freeboard-deck well 28 m long with a 1.5 m bulwark and standard sheer. Its required facts are: the $\ell_b > 20$ m branch applies, giving 1.96 m^2; the height branch is triggered (1.5 m > 1.2 m); the excess is three 0.1 m increments of $0.004 \times 28 = 0.112\,m^2$, a correction of 0.336 m^2; and the minimum area is 2.296 m^2 per side. The sibling item `R500_0117` (24 m, 1.5 m, no sheer) exercises the no-sheer branch as well, ending at $1.968 \times 1.5 = 2.952\,m^2$. A model that follows the chain correctly but applies the $\ell_b \leq 20$ m formula, or omits the no-sheer increase, fails the item; E3 targets exactly this default-versus-special-branch error.

F SOURCE-INGESTION PIPELINE

Source ingestion has two corpus-specific paths. For DNV Ship Rules, we use MinerU (Wang et al., 2024) to extract structured markdown from regulatory PDFs. For 12 CFR Part 217, we use markdown prepared from the public eCFR text and bypass PDF extraction. The source text fed to evaluated systems and to the v5d audit is markdown in both cases; chart and figure fidelity is therefore bounded by MinerU's rendering only for DNV PDF-derived items, which is a known limitation for DNV items whose *required_facts* reference design-load or dimensional charts. Per-corpus configuration consists of (i) a section-identifier regex and (ii) an anchor-expansion rule for `full_context` assembly; generation, audit, and judge prompts transfer without per-corpus rewriting.

G SAMPLE ITEM

We render one released DNV item end-to-end so a reader who never opens the artefact sees what an item looks like. Item `F500_0015` (DNV Ship Rules, Pt3.Ch12.Sec10, T2) is a freeing-port and hatch-cover compliance check on a 140 m general cargo vessel.

Scenario (verbatim). *"A 140 m general cargo vessel has a freeboard deck with a forward well formed by bulwarks. The bulwark in way of the forward well is 25 m long and has an average height of 1.5 m. The ship has no sheer. The designer proposes to fit freeing ports on each side of the forward well with a total area of 2.80 m^2 per side. The securing bolts for the internal watertight hatch covers in the cargo hold are spaced at intervals determined by reference to the bolt area requirements for ordinary packed hatch covers. The designer needs to confirm: (1) Is the proposed freeing port area of 2.80 m^2 per side sufficient for the forward well, and if not, what is the minimum required area? (2) Where in the rules would the designer find the net bolt area requirements per bolt for the ordinary packed hatch covers securing those internal hatches?"*

Cross-reference chain. `Pt3.Ch12.Sec10` → `Sec1 [3.3.1]` → `Sec3 [4.1.2]` → `Sec2 [1.3]` (depth 3, T2: cross-section within the same chapter Pt3.Ch12).

Required atomic facts (gold derivation).

1. Base freeing-port area for $\ell_b = 25$ m is $A = 0.07 \times 25 = 1.75\,m^2$.
2. Bulwark height of 1.5 m exceeds 1.2 m by 0.3 m, requiring an increase of $0.004 \times 25 \times 3 = 0.30\,m^2$, giving an adjusted base area of 2.05 m^2.
3. No-sheer condition requires a 50% increase, giving a minimum required area of 3.075 m^2 per side.
4. The proposed 2.80 m^2 per side is insufficient and does not comply.
5. The net bolt-area requirements per bolt for ordinary packed hatch covers are found in the access-openings section of the same chapter (Sec2 [1.3]).

6. The cross-reference to bolt-area scantlings is reached via the internal-hatch securing requirements, which explicitly direct to the Sec2 [1.3] provisions.

Expected verdict. The proposed 2.80 m^2/side is insufficient; the minimum required area is **3.075 m^2 per side**. Net bolt-area requirements: Pt3.Ch12.Sec2 [1.3].

Strict scoring. A model answer is correct only if a Sonnet 4.6 judge verifies all six required facts present in the answer text. An answer that lands the 3.075 m^2 verdict but omits e.g. fact 2 (bulwark-height adjustment) is graded *incorrect*, since RegBench grades source-grounded derivation, not conclusion-only accuracy. The full JSON record for this item (including `annotator_grounding` with the start-clause / end-clause / chain-summary / expected-derivation fields) is the corresponding line of `data/dnv.jsonl` in the released artefact, keyed by `id == "F500_0015"`; Appendix H maps each JSON field onto the formal tuple from §3.1.

H TUPLE-FIELD MAPPING FOR A RELEASED ITEM

This appendix maps the released JSON record onto the formal tuple from §3.1:

$$(\textit{scenario}, \textit{anchor}, \textit{chain}, \textit{required_facts}, \textit{verdict}).$$

We render the mapping for two items: a DNV Ship Rules example (`F500_0015`, the worked item from Appendix G) and a 12 CFR Part 217 example (`B_T2_0000`). The same JSON schema applies in both corpora, so the mapping ports unchanged.

Common dataset metadata (all items). `id`, `domain` (`dnv_ru_ship` or `basel_12cfr217`), `tier` (0–4), `format` (`explanation` or `multiple_choice`), and `tested_pattern` (e.g. `multi_hop_navigation`, `cross_section_lookup`). These fields support stratification and analysis but are not part of the formal tuple.

Table 9: Tuple-field mapping for the DNV worked item `F500_0015` (Pt3.Ch12.Sec10, T2): formal tuple components in §3.1 mapped to the released JSON schema fields, with the released value excerpted for each field.

Tuple field	JSON path	Released value (excerpt)
scenario	`question_text` (+ `scenario_parameters`)	"A 140 m general cargo vessel … freeing port area of 2.80 m^2 per side …" (full text in Appendix G)
anchor	`source_section`	`Pt3.Ch12.Sec10`
chain	`chain.{start, end, path, depth, edges_raw}`	`Pt3.Ch12.Sec10` → `Sec1 [3.3.1]` → `Sec3 [4.1.2]` → `Sec2 [1.3]`; depth 3; cross-section within same chapter
required_facts	`required_facts` (ordered list of 6)	6 atomic facts: base-area formula; bulwark-height adjustment; no-sheer increase; sufficiency verdict; bolt-area cross-reference; chain-summary fact (verbatim list in Appendix G)
verdict	implied by `required_facts`	3.075 m^2/side minimum; proposed 2.80 m^2 insufficient; bolt-area requirements at Pt3.Ch12.Sec2 [1.3]
(derivation)	`annotator_grounding.{start_clause, end_clause, chain_summary, expected_derivation}`	SME-anchored clause spans + verbatim derivation chain (used for audit re-review, not for model input)
(quality filters)	`leak_check`	confidence-gate result released as a JSON object on each item; per-item v5d audit verdicts are released alongside the pool, while raw SME re-review labels are retained internally (§3.5)

Example 1 — DNV `F500_0015` (T2, Pt3.Ch12.Sec10).

Example 2 — Basel `B_T2_0000` (T2, Pt217.SubpartE.Sec217.132). A Basel-side T2 item on an nth-to-default credit-derivative capital calculation. The same JSON schema applies (Table 10); the only corpus-specific difference is in `chain.kinds`, which uses `same_subpart` / `cross_subpart` edges instead of DNV's `within_chapter` / `cross_chapter` (the schema-slot binding is documented in Appendix Q).

Table 10: Tuple-field mapping for the Basel worked item `B_T2_0000` (Pt217.SubpartE.Sec217.132, T2). Same JSON schema as Table 9; corpus-specific differences are confined to `chain.kinds` edge labels (`same_subpart` / `cross_subpart`).

Tuple field	JSON path	Released value (excerpt)
scenario	`question_text`	"A Board-regulated institution acting as an advanced approaches bank has entered into an nth-to-default credit derivative … five wholesale obligors with RBC requirements of $1.2M, $0.9M, $2.1M, $0.7M, $1.5M …"
anchor	`source_section`	`Pt217.SubpartE.Sec217.132`
chain	`chain.{start, end, path, depth, kinds, raws}`	`Sec217.132` → `Sec217.135` → `Sec217.142`; depth 2; `kinds = [same_subpart, same_subpart]`; `raws = [§217.135, §217.142(m)]`
required_facts	`required_facts` (ordered list of 5)	5 atomic facts: SFA/SSFA treatment under §217.142(l)(1); ordered RBC ranking and first-to-default exposure selection ($0.7M); no-mitigant condition; standardized counterparty-credit-risk approach for the unrecognised purchased derivative; non-SPE wholesale risk-weight selection
verdict	implied by `required_facts`	First-to-default exposure drives capital; applicable RBC $0.7M; purchased OTC derivative risk-weighted under wholesale CCR rules
(derivation)	`annotator_grounding.*`	SME-anchored clause spans across SubpartE
(quality filters)	`leak_check`	confidence-gate result released as a JSON object on each item; v5d audit verdicts and reviewer-B/C/D dispositions are retained internally (§3.5)

Schema portability. The two examples differ only in (i) the `domain` value, (ii) corpus-specific `chain.kinds` labels (DNV uses chapter/part edges; Basel uses subpart edges), and (iii) the per-corpus structural unit that fills the schema's *chapter* slot (DNV Chapter vs Basel Subpart, cf. Appendix Q). The tuple fields themselves — *scenario*, *anchor*, *chain*, *required_facts*, *verdict* — are corpus-independent and resolve to the same JSON paths in both items, which is what lets the construction pipeline and the v5d audit transfer unchanged across DNV and Basel.

I PROMPT TEMPLATES

The locked v5d audit prompt is released verbatim at `prompts/v5d_audit.md` in the GitHub release. The generation, scenario-synthesis, confidence re-score, and answer-grading judge prompts are reproduced inline in the released pipeline source (`pipeline/generation/` and `pipeline/judge/`); a stand-alone reference implementation of the strict atomic-fact-conjunction grader ships at `examples/grade_strict.py`. The full assembled v5d prompt is also reproduced in the supplementary material.

J REPRODUCIBILITY MANIFEST

Leaderboard with intervals. Table 11 gives the counts and Wilson 95% intervals behind Table 3. Judge replies that could not be parsed after re-querying are scored incorrect, as for every system (0–4 per system and corpus).

Table 11: Strict accuracy counts and Wilson 95% intervals (%) behind Table 3.

	DNV ($N = 499$)		**Basel** ($N = 281$)	
Model	Primary	Cross-judge	Primary	Cross-judge
GPT-5.6-sol `fc`	246 [44.9–53.7]	218 [39.4–48.1]	158 [50.4–61.9]	148 [46.8–58.4]
BM25+Sonnet 4.6	222 [40.2–48.9]	204 [36.7–45.2]	160 [51.1–62.6]	140 [44.0–55.6]
Sonnet 4.6 `fc`	220 [39.8–48.5]	207 [37.2–45.9]	146 [46.1–57.7]	132 [41.2–52.8]
GPT-5.4 `fc`	215 [38.8–47.5]	199 [35.7–44.2]	127 [39.5–51.0]	106 [32.3–43.5]
GPT-5.6-terra `fc`	202 [36.3–44.8]	183 [32.6–41.0]	139 [43.7–55.3]	122 [37.7–49.3]
Qwen3.6-Plus `fc`	179 [31.8–40.2]	163 [28.7–36.9]	100 [30.2–41.3]	81 [23.8–34.4]
MiniMax-M2.7 `fc`	83 [13.6–20.2]	75 [12.2–18.4]	48 [13.1–21.9]	47 [12.8–21.5]

The released artefact contains: (i) the 827-question / 4,766-fact promoted pool; (ii) the v5d audit prompt verbatim (`prompts/v5d_audit.md`); (iii) per-item v5d audit dispositions on the audited subset of the released pool (`data/v5d_verdicts.jsonl`; full pilot coverage and the DNV calibration audit slice; per-fact reads regenerable by re-running the audit driver with the locked prompt — see `data/v5d_verdicts.README.md`); (iv) the answer-grading judge prompt and the strict-conjunction grader as a stand-alone reference implementation (`examples/grade_strict.py`); (v) the construction, audit, and evaluation pipeline source code (cross-reference graph extraction, scenario synthesis, v5d audit driver, baseline runners, judge); (vi) the DNV PDF→markdown ingestion driver with a vendored MinerU build; (vii) per-tier sample items rendered in Appendix G and Appendix H; (viii) a reproducibility manifest with file checksums and model-version metadata (`MANIFEST.json`); (ix) MLCommons Croissant (Akhtar et al., 2024) v1.1 metadata at `artifacts/croissant.json`. The artefact also contains: (x) SME-vs-v5d and SME-vs-judge disagreement logs, with per-item SME labels and reviewer rationale on every contested label (`artifacts/sme_stats/disagreements/`); (xi) the flag-to-disposition ledger (repair / kill / spurious) and the 837→827 kill ledger (`artifacts/sme_stats/disposition_ledger.json`); (xii) per-fact primary- and cross-judge verdicts for every evaluated system, with full answer text for the GPT-5.6 runs; (xiii) the SME-certified perturbed answer keys of Appendix R with their re-judged verdicts. Agreeing SME labels and the answer text of the five frozen systems are retained internally to preserve audit-slate integrity (§3.5); DNV source PDFs are not redistributed but are available after free DNV registration (Appendix K). The double-blind benchmark release page is hosted at `https://regbench.github.io/`.

Reasoning-effort table. Reasoning effort is held fixed at `medium` for every run in the paper, with two exceptions: the v5d audit pipeline (Appendix M) runs GPT-5.4 at `xhigh`, and GPT-5.6-terra is evaluated at `high`. The other leaderboard runs (§4.2), including GPT-5.6-sol, the primary answer-grading judge, the GPT-5.4 judge replay (Appendix O), and the cross-vendor v5d replay (Appendix B) use `medium`; the DeepSeek-v4-pro cross-judge runs at the vendor's default reasoning setting, since reasoning tokens share its output budget, and replies that exhaust the budget before emitting a verdict are re-queried with a larger one. We do not boost effort beyond these settings: the leaderboard reports each model at the compute regime an end-user would reasonably pay for, not at saturation.

K DATASHEET

We follow the Datasheets-for-Datasets format (Gebru et al., 2021).

Motivation. Created to enable evaluation of LLMs on source-grounded multi-hop regulatory reasoning, a capability not isolated by existing benchmarks (§2). Created by the RegBench authors; no external funding is disclosed at submission. Anonymous in the submission version.

Composition. 827 instances (questions); each instance is a tuple of `(scenario, anchor, chain, required_facts, verdict)` (§3.1). 546 from DNV Ship Rules (R500_, F500_,

P50_ prefixes), 281 from 12 CFR Part 217 (B_ prefix). 4,766 atomic-fact propositions across all questions. No missing instances or hidden splits beyond the public release. No personal or sensitive content (scenarios involve hypothetical ship designs and bank capital positions).

Collection. Scenarios are LLM-synthesised (Sonnet 4.6 at $T{=}0.3$) over chains sampled from the cross-reference graphs of two public regulatory corpora (DNV Ship Rules; 12 CFR Part 217). DNV source PDFs are obtained from `https://standards.dnv.com/explorer/RU-SHIP` and parsed with MinerU (Wang et al., 2024); 12 CFR Part 217 uses markdown prepared from the public eCFR text. Section identifiers are normalised in both corpora. All items pass v5d audit; items with non-repairable defects (10 total) were killed during SME re-review and are not in the release. The pipeline timeline is 2026-Q1 to 2026-Q2.

Pre-processing / cleaning / labelling. The markdown source used for each corpus is preserved alongside the released items so source-ingestion and DNV chart-fidelity issues (§F) are reproducible. The annotation protocol is documented in Appendix B: blinded, source-backed re-solve with first-faulty-step localisation; SMEs see v5d's item-level disposition as an advisory hint on flagged items but never see v5d's per-fact reasoning on v5d-passed items (interface contract that supports the structurally-blind 0/446 miss rate measurement). Raw SME annotation labels and the annotation interface itself are retained internally rather than released, to preserve audit-slate integrity for future benchmark versioning.

Uses. Intended for evaluation of LLMs and retrieval systems on source-grounded multi-hop regulatory reasoning, analysis of degradation with required regulatory context scope, and auditing benchmark-construction pipelines for regulatory corpora. Not intended for: training compliance-decision systems; certifying any model for production regulatory use.

Distribution. The double-blind benchmark release page is `https://regbench.github.io/`. The dataset is released under **CC-BY-NC 4.0** via Hugging Face Datasets and a long-term Zenodo DOI. Each release is a frozen versioned snapshot (`v1.0.1`); errata batched quarterly as point releases. Source-corpus PDFs are NOT redistributed: the DNV PDFs are gated behind a free DNV registration, and section identifiers in the released items map deterministically to those PDFs so downstream users can re-fetch and re-extract the markdown the pipeline operates on; 12 CFR Part 217 markdown is prepared from the public eCFR text. The released GitHub repository ships a vendored MinerU build and a section-level extraction driver so the DNV markdown is reproducible from a registered user's downloaded PDFs without further configuration.

Audit disagreement ledger. The release repository includes the aggregated SME disagreement ledger (`artifacts/sme_stats/disagreements/`): one file for judge–SME disagreements in answer grading and one each for v5d–reviewer disagreements in the DNV and Basel item audits. Each record gives the contested label, its direction, the visibility condition under which it was made, and reviewer notes where available; companion files report per-slice agreement with Wilson 95% intervals and the complete flag-to-disposition ledger.

Maintenance. Maintained by the authors for at least 24 months post-publication. Contact via the release repository. Errata batched quarterly. Canary items embedded in the candidate pool support post-release leakage monitoring. On detection of leakage above a published threshold, a saturation refresh is triggered. The release repository is hosted at `https://github.com/regbench/regbench`. The GitHub account is project-dedicated and untraceable to author identity: registered to an anonymous email, carrying no author name, institution, contact, or profile information, hosting no other repositories, and with all commit-author metadata anonymised. Following the link reveals only the benchmark artefact; double-blind review is preserved end-to-end.

L CROISSANT METADATA, LICENSE, AND HOSTING

The release ships an MLCommons Croissant v1.1 metadata descriptor at `artifacts/croissant.json`, validated against the official MLCommons Croissant validator. Both the core fields (`name`, `description`, `license`, `version`, `distribution`,

per-pool record-set schemas with field-level types) and the Responsible-AI (RAI) extension fields are populated: `rai:dataCollection` (LLM-synthesised over public regulatory cross-reference graphs), `rai:dataAnnotationProtocol` (two-stage v5d audit + SME source-grounded re-review; protocol detail in §3.4), `rai:dataAnnotationAnalysis` (the calibration numbers reported in this paper), `rai:dataLimitations` (mirrors Appendix C), `rai:dataSocialImpact` (intended for evaluation; not for training compliance-decision systems), `rai:dataBiases` (English-only; two-corpus coverage; English-language regulatory regimes), `rai:personalSensitiveInformation` (none).

M How the v5d locked rules were derived

This appendix documents how the locked v5d audit prompt (§3.3, Appendix E) was constructed. The construction was an SME-driven prompt-tuning loop on a closed development pool, frozen before any downstream split was touched. We describe the loop, the development pool, the version progression, the knob-by-knob ablation, and the freeze criterion. Code, prompts, and per-version verdicts are released so the derivation is reproducible end-to-end.

Method (one-paragraph summary). SMEs source-grounded re-reviewed every item in a held-out development pool and flagged each defective `required_fact` together with a short free-text reason summary (e.g., on `R500_0036`: "Table 11 row swap — fact reads $C_{YR}{=}-1$ for BSR-1P, but the source has $C_{YR}{=}1$ for BSR-1P; the -1 value belongs to the separate C_{YG} row"). We grouped reason summaries by the structural primitive of regulatory source they touched — table cells, formula expressions, and case-tree branches — and distilled three recurring failure scopes (E1/E2/E3 in the paper, T1/T2/T3 in the prompt) that together accounted for the substantive defects in the pool. Each scope was promoted to an explicit pattern guardrail in the audit prompt: a verdict of `incorrect` on that scope is admissible only when ALL of the scope's pattern triggers fire precisely (Appendix E). Prompt versions were measured against the SME labels with three metrics in priority order — recall on SME-flagged items, item-level agreement, and flag rate — and a version was retained only if it strictly improved at least one without regressing the others. The v5d locked configuration is the assembly that survived this loop. After lock, no clause of the prompt was altered for any downstream application.

Development pool. The dev pool is the 148-question DNV TRAIN audit: 48 items from the SME-1 design pilot plus 100 items from a subsequent SME-3 source-backed audit, with 16 SME-flagged positives and 132 SME-passed negatives. Every dev item carries (a) a per-fact label set, (b) an item-level verdict, and (c) for flagged facts, an SME free-text reason summary citing the offending source primitive. The pool is disjoint from DNV TEST, DNV selective audit split, 12 CFR Part 217, and all model-leaderboard splits used in §4.

Version progression. Five generations of the prompt were tried before lock; intermediate versions are kept in the public registry for ablation reproducibility (`filter_dev/filter_prompts.py`, `PROMPTS` dict).

- **v1 (baseline neutral).** Flag any `required_fact` not directly verifiable in source. Recall 100%, item agreement 38%, flag rate ∼60% — catches everything but over-flags paraphrase nuance.
- **v2 (conservative).** Flag only on explicit verbatim contradictions. Recall fell to 62%; agreement rose to 80%. Trades too much recall for the precision gain.
- **v3 (chain-of-verdicts with mandatory source-span citation, broader source loading, `multi_uncertain` aggregation).** Recall returned to 100% on the 16-positive set with 76% agreement. v3 was used as the held-line baseline for v5 development.
- **v5 family (modular knobs).** Decomposed the v3 line into four orthogonal knobs — (1) tightened `incorrect` verdict (substantive contradictions only), (2) clause-numbering / sub-bracket convention, (3) JSON output schema, (4) strong conservative-bias replacement, (5) E1/E2/E3 pattern guardrails distilled from the SME reason summaries — and ablated their combinations as v5a/b/c/d (Table 12).

Knob-by-knob ablation on TRAIN. v5d combines all knobs and was the configuration locked for downstream use. The ablation registry (`filter_prompts.py`) lets a re-runner reproduce each variant against the dev pool.

Tag	Composition	Flags	Recall	Prec.	Agree
v1	Aggressive base, narrow source	108	100%	14.8%	37.8%
v2	Conservative base, narrow source	34	62.5%	29.4%	79.7%
v3	Chain-of-verdicts, broad source (native item-verdict)	41	81.2%	31.7%	79.1%
v5a	Tightened-`incorrect` base only (knobs 2+3)	24	75.0%	50.0%	89.2%
v5b	v5a + conservative-bias replacement (knobs 2+3+1)	25	81.2%	52.0%	89.9%
v5c	v5a + E1/E2/E3 pattern guardrails (knobs 2+3+5)	22	75.0%	54.5%	90.5%
v5d	**All knobs (2+3+1+5)**	**26**	**81.2%**	**50.0%**	**89.2%**

Table 12: Prompt-version ablation on the 148-question DNV TRAIN dev pool (49 SME-1 pilot + 100 SME-3 source-backed audit; 16 dev-time SME-flagged positives), using each version's native item-level verdict directly. v5a/b/c are intermediate ablations retained for reproducibility but not used downstream. v5d was locked over v5c because the conservative-bias knob recovers one true positive (recall 75%→81%) while losing 4.5pp precision and 1.3pp agreement: the recall gain is a defensible trade for the cross-corpus transfer where false positives would burn SME re-review time on Basel. Numbers are reproducible from `filter_dev/results/results_v{1,2,3,5a,5b,5c,5d}.json` with `analyze_custom.py`. Note: §3.3 reports v5d's *post-rejudge* canonical numbers on the SME-3 100-Q subset (17/17 precision, 17/18 = 94% recall on substantive defects), which are the calibration figures used in the main results; the table here uses the dev-time pre-rejudge labels to preserve apples-to-apples comparison across versions.

From SME reason summaries to pattern guardrails. The three pattern guardrails in v5d (T1 table row/column mismatch, T2 formula integrity, T3 default-vs-special branch; verbatim text in Appendix E) are not authored from first principles — they are the structural distillation of the SME free-text reason summaries on the TRAIN dev pool. Each guardrail records the conjunctive trigger conditions a defect on that scope had to satisfy in the SME write-ups: e.g. T1's three-part trigger (table present in source, scenario-matching row+column locatable, source-cell value materially different from fact's claim) is the minimal ALL-of-the-following the SMEs themselves required before they would call a table-cell mismatch a real defect rather than a paraphrase concern. Promoting these conditions into the prompt as ALL-must-hold checks is what lifts agreement against pre-rejudge dev labels from ∼79% (v3 native) to ∼89% (v5d), at the cost of a small recall hit that the post-rejudge SME-3 re-review later closes back to 17/18 = 94% on substantive defects.

Freeze criterion and lock point. v5d was locked when (i) the four-knob combination strictly dominated each three-knob ablation on at least one of {recall, precision, agreement} without regression on the others, (ii) the SME-flagged-defect recall on TRAIN had stabilised across three consecutive perturbations of the conservative-bias and pattern-guardrail wordings (no further reword changed the verdict on more than one item), and (iii) the flag rate was within the 15–25% band that downstream SME re-review could absorb for the 837-item pre-audit candidate pool (10 items were subsequently SME-killed, leaving the 827-item released set). After lock, the prompt text, the auditor model (GPT-5.4 at `model_reasoning_effort=xhigh`), the structured-output schema (`schema_v5d.json`), and the source-loader caps (140K bytes per section, 500K total) were frozen and applied unchanged to DNV TEST, the DNV selective audit split, the 12 CFR Part 217 pool, and the cross-vendor Sonnet 4.6 replay. No clause of the prompt was tuned against any of those splits.

What the loop does NOT claim. The loop tunes the prompt *against the dev pool's SME labels*; it does not produce a prompt that is provably optimal in any global sense. We rely on three external checks to detect over-fitting: (a) downstream agreement on a held-out TEST pool with a different SME-pair (§3.4), (b) cross-corpus transfer to 12 CFR Part 217 with no retuning (§4), and (c) a

cross-vendor replay against Claude Sonnet 4.6 to detect prompt-model entanglement (§3.4). The *construction* of the prompt is SME-supervised; the *evaluation* of the locked prompt is on splits no one in the loop has seen.

Reproducibility pointers. The locked v5d audit prompt and its driver are released:

- Verbatim locked prompt: `prompts/v5d_audit.md` in the GitHub release, which reproduces the four-block assembly used at release; also reproduced in the supplementary material.
- Audit driver: `pipeline/audit/auto_gt_audit.py` (codex / OpenAI-compatible CLI invocation, structured-output schema enforcement) with prompt templates in `pipeline/audit/gt_audit_prompts.py`.

The prompt-version intermediates (v1, v2, v3, v5a/b/c) and the SME consensus dev labels they were tuned against are retained internally to preserve audit-slate integrity (§3.5); the ablation summary in Table 12 reports the per-version numbers needed to motivate v5d's lock point.

N PILOT PERFORMANCE

This appendix expands §4.3: the same 47-Q post-audit-and-repair DNV pilot, evaluated over 13 systems (Table 13, Figure 5). The panel spans Anthropic's three-tier capability scale (Opus 4.6 (Anthropic, 2026a) / Sonnet / Haiku 4.5 (Anthropic, 2025)), four frontier model families (Sonnet, GPT-5.4, Qwen3.6-Plus and Qwen3-Max (Qwen Team, 2025), MiniMax) under multiple context conditions (`full_context`, BM25, ChainRAG), and three published graph-RAG variants (ArchRAG, HippoRAG, GraphRAG). *ChainRAG* is a custom RegBench baseline (released alongside the artefact): BM25 retrieves the top-5 anchor sections, three cross-reference regexes follow exactly one hop into the referenced sections, and the concatenation (anchor + 1-hop referents, capped at 400K characters) is fed to the reader. It is included as a structural baseline that follows explicit references while retaining clause text. For each system we report three quantities: per-fact rate (lenient: average across required facts), conclusion-only rate (lenient: just the final verdict fact), and strict accuracy (the RegBench scoring rule: every required fact present). The phantom-credit gap is conclusion − strict: the share of answers whose verdict is credited but whose required derivation detail is not.

Table 13: Pilot performance on the 47-Q post-audit DNV pilot. Graph-RAG rows are highlighted.

System (47-Q post-audit DNV pilot)	Per-fact	Conclusion	**Strict**	Phantom credit
Opus 4.6 (oracle)	87.1%	73.9%	**55.3%**	18.6pp
BM25+Sonnet 4.6	82.5%	83.0%	**53.2%**	29.8pp
Sonnet 4.6 (`fc`)	74.3%	61.7%	**51.1%**	10.6pp
Haiku 4.5 (oracle)	79.2%	68.1%	**46.8%**	21.3pp
GPT-5.4 (`fc`)	78.8%	59.6%	**40.4%**	19.1pp
BM25+GPT-5.4	79.2%	60.9%	**38.3%**	22.6pp
Qwen3.6-Plus (`fc`)	78.4%	55.3%	**38.3%**	17.0pp
GPT-5.4 (ChainRAG)	76.6%	63.8%	**36.2%**	27.7pp
Qwen3-Max (`fc`)	63.2%	38.3%	**19.1%**	19.1pp
HippoRAG (Gutiérrez et al., 2024)	58.0%	48.9%	**17.0%**	31.9pp
GraphRAG (Edge et al., 2024)	43.1%	29.8%	**17.0%**	12.8pp
MiniMax-M2.7 (`fc`)	37.5%	19.1%	**14.9%**	4.3pp
ArchRAG (Wang et al., 2026a)	21.2%	17.0%	**0.0%**	17.0pp

Phantom-credit gap. Across the panel, the phantom-credit gap (conclusion − strict) has median $\approx$ 19pp and ranges from 4.3pp (MiniMax) to 31.9pp (HippoRAG). Conclusion-only scoring reports higher scores by this margin on the pilot. We read the gap as uncredited derivation detail rather than as a count of undeserved correct verdicts: it includes verdicts reached without the required chain, but also correct, concise derivations that omit an intermediate fact, which strict atomic-fact conjunction marks incorrect by design (§3.1). The answer-key sensitivity study (Appendix R) shows that how facts are segmented does not drive the scores, but it does not separate these two cases.

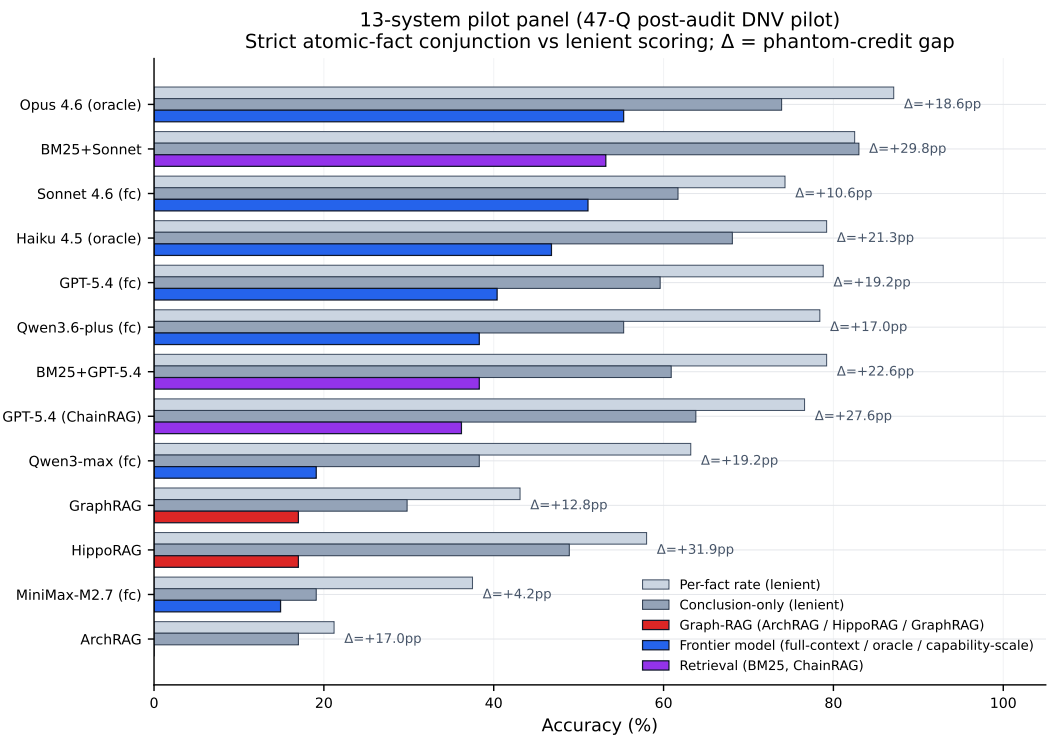

Figure 5: Pilot performance: per-fact, conclusion-only, and strict atomic-fact conjunction across 13 systems; Δ on the right is the phantom-credit gap.

Graph-RAG failure. Three published graph-RAG methods score near the bottom of the strict ranking: ArchRAG (Wang et al., 2026a) 0%, HippoRAG (Gutiérrez et al., 2024) 17.0%, and GraphRAG (Edge et al., 2024) 17.0%. The latter two exceed MiniMax's 14.9% pilot score, while all three trail BM25 retrieval over the same corpus (53.2% with a Sonnet reader). Their retrieval representations and readers differ, so this pilot does not isolate the cause of the gap. Exact numerics, clause IDs, and formula branches remain possible failure points to test in a controlled ablation.

Scope and caveats. The panel is scoped to the 47-Q DNV post-audit pilot to keep the 13-system comparison tractable. The pilot result is informative but not benchmark-wide: graph-RAG behaviour at the released 499-Q DNV scale and on Basel is not measured here, and the BM25+Sonnet 4.6 advantage we observe at the hardest DNV tiers (§4.3) is shown only against the four scale-pool comparators in this paper. Extending graph-RAG and additional retrieval methods to the full release is left as future work. Per-fact, conclusion, and strict numbers come from the same Sonnet 4.6 judge used in §4.2 and pass the SME calibration in §4.1 / Appendix O.

O Cross-judge stability

Beyond the full-leaderboard cross-judge (§4.4), we replay a third judge vendor on the 47 released pilot items: GPT-5.4 at medium effort and DeepSeek-v4-pro each replace Sonnet 4.6 across all 12 model–condition pairings, with the same prompt. Throughout, an answer is strict-correct when the judge marks every required fact present, the definition used for the primary judge. On the 3,295 fact verdicts shared by the three judges (primary-judge verdicts as recorded at pilot annotation time), pairwise agreement is 95.5% (Sonnet–DeepSeek), 94.7% (Sonnet–GPT-5.4) and 95.8% (DeepSeek–GPT-5.4); per pairing, GPT-5.4 agrees with the primary judge at 89.6–97.9%. GPT-5.4's disagreements with the primary judge are mostly strict (149 strict vs 26 lenient), yet it preserves the strict ordering of all pairings (Kendall $\tau = 1.000$ over the 60 untied pairs), and the three graph-RAG variants score 14.9%, 10.6% and 0.0% strict under both replacement judges, confirming their collapse. Ordinal claims

therefore survive a change of judge vendor, while absolute fact-presence rates remain calibrated to the primary judge.

P MEMORISATION RULE-OUT (CLOSED-BOOK)

We rule out regulation-memorisation as a confound for RegBench performance by evaluating models without source text. Closed-book strict accuracy on the 47-Q post-audit-and-repair DNV pilot is **0%** across the two Anthropic model families we evaluated (Sonnet 4.6, Haiku 4.5): no model strict-solves a single item from the scenario alone (the original closed-book run covered the 49-Q pre-kill pool with the same 0% rate; restricting to the released 47-Q pool leaves the rate unchanged). Source use, not regulation memorisation, drives the leaderboard rates reported in §4.2. Scenarios are LLM-synthesised after the source-corpus snapshots, so item-level contamination is precluded by construction; latent pattern memorisation (e.g., a model having seen Basel-style capital-instrument fact patterns during pre-training) is not addressed by closed-book and remains an open post-release leakage-monitoring item. Per-model breakdowns, cross-vendor extension to non-Anthropic families, and the closed-book run logs appear in the supplementary material.

Q TIER-SCHEMA INSTANTIATION PER CORPUS

The tier schema in §3.1 is corpus-independent in form: it labels chains by the topological scope of cross-references they traverse (single-clause; single-section requiring table/formula/branch; cross-section in same chapter; cross-chapter in same Part; cross-Part / cross-volume). What differs across corpora is which level of the corpus's native hierarchy fills each schema slot. This appendix documents that mapping so cross-corpus tier comparisons are read against a shared structural definition.

Hierarchy mapping. Table 14 gives the per-corpus instantiation.

Table 14: Tier-schema instantiation per corpus. Each row maps a corpus-independent schema slot to the structural unit of the corpus that fills it, so cross-corpus tier comparisons read against a shared structural definition.

Schema slot	DNV Ship Rules	12 CFR Part 217
Section	Section (e.g., `Sec5`)	Section (e.g., `217.132`)
Chapter	Chapter (e.g., `Ch12`)	Subpart (e.g., `SubpartE`)
Part	Part (e.g., `Pt3`)	Part 217 (entire in-scope corpus)

In this paper, the Basel corpus is the U.S. federal banking regulation *12 CFR Part 217* (Regulation Q) (Board of Governors of the Federal Reserve System, 2024), a multi-Subpart rule implementing the Basel III framework (Basel Committee on Banking Supervision, 2010) rather than a single section. Sections are atomic in both corpora; the schema's *chapter* slot is filled by a DNV Chapter or a Basel Subpart of Part 217.

Concrete chain examples (released items).

- **T2 (cross-section, same chapter).**
  - DNV `F500_0012`: `Pt3.Ch12.Sec10` → `Sec1` → `Sec3` — all within `Pt3.Ch12`.
  - Basel `B_T2_0000`: `Pt217.SubpartE.Sec217.132` → `Sec217.135` → `Sec217.142`; `kinds = same_subpart, same_subpart`.
- **T3 (cross-chapter, same Part).**
  - DNV `F500_0080`: `Pt3.Ch10.Sec4` → `Pt3.Ch4.Sec6` → `Pt3.Ch4.Sec3` — crosses `Ch10`→`Ch4` within `Pt3`.
  - Basel `B_T3_0075`: `Pt217.SubpartF.Sec217.207` → `Sec217.210` → `Pt217.SubpartA.Sec217.2`; `kinds = same_subpart, cross_subpart`.

- **T4 (cross-Part / cross-volume).**
  - DNV `F500_0132`: `Pt3.Ch13.Sec1` → `Pt3.Ch8.Sec2` → `Pt3.Ch3.Sec6` → `Pt5.Ch2.Sec2` — crosses `Pt3`→`Pt5`.
  - Basel: not populated. T4 chains would leave Part 217, which is the in-scope corpus, and are excluded (§3.6).

Encoding in the released artefact. The chain object on each released item carries a `kinds` field whose values name the corpus-specific edge type traversed at each hop (`within_section`, `same_subpart`, `cross_subpart` for Basel; `within_chapter`, `cross_chapter`, `cross_part` for DNV), so the schema-slot mapping is recoverable directly from the data without consulting this table.

Reading cross-corpus tier results. Because the schema slots bind to different native hierarchy levels, two items at the same tier in different corpora are matched on *topological scope* (e.g., both are cross-section within the chapter-equivalent unit), not on absolute clause count, source-text length, or domain difficulty. Cross-corpus claims in this paper therefore report shapes (clear tier degradation as tier increases; per-tier model rankings) rather than direct absolute comparisons; absolute levels are reported within each corpus.

R INDEPENDENCE CHECKS: PROTOCOLS AND FULL RESULTS

Table 15: Robustness checks on model judging and construction. Each row states a way the models in the pipeline could bias the leaderboard, the test applied, and the outcome.

Threat	Test	Outcome
Scores depend on the judge	Cross-judge re-scores every answer of 7 systems on both corpora (5,460 answers)	40/42 pairwise orderings preserved; both swaps between tied systems
Judge favours its own model	SME fact labels (2,453) split by answering model	Lenient errors 0.29% (2/693) on Sonnet answers vs 0.09% (1/1,092) elsewhere; Fisher $p = 0.56$
Generator favours its own model	65 SME-audited items generated by DeepSeek-v4-pro (13 per tier), cross-judge scored	Sonnet 4.6 still leads (56.9%) ahead of Qwen3.6-Plus (52.3%), GPT-5.6-sol (49.2%), GPT-5.4 (44.6%)
Keys fit one decomposition	143 DNV and 281 Basel keys re-segmented and certified interchangeable by SMEs; answers and judge held fixed	No significant shift (DNV $\lvert\Delta\rvert \leq 2.8$pp, all $p \geq 0.42$; Basel all 95% CIs include 0); Basel ranking identical, DNV ranking changes only within the tied top three
Audit agreement reflects SMEs seeing v5d	Agreement restricted to DNV slices where v5d output was hidden	97.3% item (145/149, development pool); 98.9% fact (561/567, held-out TEST)

Cross-judge re-scoring. DeepSeek-v4-pro re-scores every leaderboard answer (7 systems on both corpora, 5,460 answers) from the frozen model outputs, with the judge prompt, required facts and decoding held identical to the primary pass; no evaluated model is re-run. Replies whose reasoning exhausted the output budget before any verdict was emitted are re-queried with a larger budget, so coverage is complete. Relative to the primary judge, every system moves down, by 0.4 to 7.5pp (Table 3). On each corpus 20 of the 21 pairwise orderings are preserved (Kendall $\tau = 0.905$). The two exceptions are BM25+Sonnet vs Sonnet `fc` on DNV (primary 222 vs 220 correct, cross-judge 204 vs 207; $p > 0.10$ under both judges) and GPT-5.6-sol vs BM25+Sonnet on Basel (primary 158 vs 160, discordant items 43 vs 45, $p = 0.92$; cross-judge 148 vs 140, $p = 0.48$).

Self-preference of the primary judge. We split the 2,453 SME-1 fact labels by the model that wrote the answer. A *lenient* judge error (the judge credits a fact the expert marked absent) is the self-favouring direction and can only occur where the judge says present. Normalised by that opportunity, the lenient rate is 0.29% (2/693) on answers written by Sonnet 4.6 (full-context, oracle, and BM25

reader) and 0.09% (1/1,092) on the other nine pairings, a difference of 0.20pp (Fisher exact $p = 0.56$). The judge's remaining six disagreements with SME-1 are strict.

Cross-judge validation. On the same pilot labels the cross-judge agrees with SME-1 at 96.0% (2,274/2,369) and with SME-2 at 94.7% (2,134/2,254). Its disagreements are mostly strict: 70 strict vs 25 lenient against SME-1 and 91 vs 29 against SME-2.

Generator swap. To test whether the generating model favours itself, DeepSeek-v4-pro replaced Sonnet 4.6 as the scenario generator in the unchanged pipeline. The resulting items were audited by SMEs against source, giving 65 items (13 per tier), and the panel was scored by the cross-judge: Sonnet 4.6 37/65 (56.9%), Qwen3.6-Plus 34/65 (52.3%), GPT-5.6-sol 32/65 (49.2%), GPT-5.4 29/65 (44.6%). Sonnet 4.6 therefore remains competitive on items it did not write. Absolute scores are higher than on the release because these items are easier: required facts are shorter (84 vs 115 characters on average) and fewer (5.02 per item vs 5.63–5.88).

Answer-key sensitivity. With model answers and the judge (DeepSeek-v4-pro) held fixed, only the answer key varies. On DNV, DeepSeek-v4-pro proposed at most one revision per released item and four SMEs adjudicated each proposal (accept/reject, nothing pre-selected); we analyse the 143 items whose revision an SME certified as interchangeable with the released key (a merge, split or rewording that preserves the requirement). On Basel, every one of the 281 items received one balanced operation: *recompose* (same requirements and fact count, new wording and boundaries), *compress* (merge two adjacent facts) or *dilute* (split one fact); SMEs certified all 281 perturbed keys as interchangeable in a randomised, unlabelled side-by-side comparison, one reviewer per item. Table 16 gives the results. On DNV item-level strict verdicts agree on 660/715 (92.3%) system-item pairs and the rank correlation is $\rho = 0.821$ ($\tau = 0.778$), the single discordant pair lying inside the top-three group that is already statistically tied; on Basel the ranking is unchanged ($\rho = \tau = 1$) and item-level agreement is 91.6%, 91.9% and 94.1% for the three operations.

Table 16: Strict accuracy (%) under the released key and an SME-certified interchangeable key, judge and answers held fixed. Δ with paired 95% CI; exact McNemar p.

	DNV (143 items)				**Basel** (281 items)			
System	Released	Perturbed	Δ [95% CI]	p	Released	Perturbed	Δ [95% CI]	p
Sonnet 4.6 `fc`	39.2	38.5	$-0.7\,[-4.8, +3.4]$	1.000	47.0	46.3	$-0.7\,[-4.1, +2.7]$	0.839
BM25+Sonnet 4.6	37.8	38.5	$+0.7\,[-3.8, +5.2]$	1.000	49.8	51.6	$+1.8\,[-2.0, +5.5]$	0.458
GPT-5.4 `fc`	38.5	37.8	$-0.7\,[-5.6, +4.2]$	1.000	37.7	40.6	$+2.8\,[-0.4, +6.1]$	0.134
Qwen3.6-Plus `fc`	27.3	30.1	$+2.8\,[-2.3, +7.9]$	0.424	28.8	28.8	$\pm 0.0\,[-3.0, +3.0]$	1.000
MiniMax-M2.7 `fc`	14.0	14.0	$\pm 0.0\,[-3.9, +3.9]$	1.000	16.7	16.7	$\pm 0.0\,[-2.4, +2.4]$	1.000

Audit agreement by visibility of v5d output. Table 17 separates the v5d–SME agreement figures by what annotators could see. Only the DNV slices contain labels made without v5d's output; the Basel reviewers saw v5d's per-fact verdicts and reasons, and the DNV development-pool facts include 49 re-reviewed with v5d's cited clauses visible. An SME "uncertain" is counted as disagreement (counting it as agreement moves 1,107 to 1,108). The development pool is also the data from which v5d's rules were derived, which is why the held-out TEST rows are the primary evidence.

Table 17: v5d–SME agreement by slice and by whether v5d's output was visible to the annotator.

Level	Slice	Agreement	v5d output visible
Fact	DNV held-out TEST (567 facts)	561/567 = 98.9%	No
Fact	DNV development pool + TEST (1,117 facts)	1,107/1,117 = 99.1%	No, except 49 re-reviewed facts
Fact	Basel audited slice (478 facts)	453/478 = 94.8%	Yes
Item	DNV pilot + TRAIN (149 items)	145/149 = 97.3%	No
Item	DNV held-out TEST (100 items)	99/100 = 99.0%	Yes (item-level disposition)
Item	Basel audited slice (79 items)	76/79 = 96.2%	Yes

Table 18: Reference-graph construction on four civil codes: resolution accuracy on stratified samples verified by legal professionals (counts in Table 19). Edge loss: share of true article-to-article edges lost to implicit-resolution errors.

Code	Provisions	Implicit : numeric	Explicit correct	Implicit correct	Edge loss
Japan	836	4.1 : 1	100%	100%	0%
Taiwan	1,376	4.5 : 1	100%	97.7%	0%
Spain	1,915	2.8 : 1	100%	88.9%	0%
Germany	2,503	1.7 : 1	97.8%	80.3%	2.8%
Pooled	6,630		99.4%	91.6%	0.9%

Table 19: Verification sample sizes and correct counts behind Table 18.

	Explicit references		Implicit references	
Code	Correct / sampled	Accuracy	Correct / sampled	Accuracy
Japan	45/45	100%	75/75	100%
Taiwan	44/44	100%	43/44	97.7%
Spain	45/45	100%	32/36	88.9%
Germany	44/45	97.8%	57/71	80.3%
Pooled	178/179	99.4%	207/226	91.6%

S CIVIL-LAW PILOT: PROTOCOL AND WORKED CHAINS

Graph construction. The four instruments are the Japanese Civil Code with the Act on Land and Building Leases, the Taiwanese Civil Code, the Spanish Código Civil and the German BGB, each in its government-published English edition. Edge loss in Table 18 is judged at article level, the granularity of graph nodes; judged at sentence level it is 1.8% pooled (5.6% for the BGB). Each code is parsed by a corpus-specific, deterministic parser that reads the instrument's structure (Book, Title, Article, paragraph) and its reference conventions; no model is used at any stage of extraction. Numeric citations resolve as in the released corpora. Implicit references resolve two classes of term against a document-derived index: *declared terms*, which the code defines ("usufruct is a real right . . . "), and *institutional terms*, which name a contract type or legal institution regulated by a specific provision ("lease", "gift contract"). Nodes are articles, so a mis-resolved sub-article descriptor either lands on the article its citation already names or becomes a self-loop, and a container-valued target (a whole Book or Title) is rejected before chain enumeration. Two legal professionals verified the resulting edges on a stratified sample of each code, judging only what refers to what (Table 18); resolution errors concentrate in the sub-article descriptor class ("sentence 3", "subsection (5)"), where the parser attached the pointer to its containing article rather than to the provision cited just before it.

A chain reachable only through a term (Louisiana Civil Code). Louisiana is a U.S. civil-law jurisdiction whose English-language code is published by the state legislature, so every provision can be checked without translation. Art. 2716 reads: *"A lease granted by a usufructuary terminates upon the termination of the usufruct. The lessor is liable to the lessee for any loss caused by such termination, if the lessor failed to disclose his status as a usufructuary."* A generated item: a woman holding a usufruct over a New Orleans warehouse leases it to a storage company for ten years at $4,000 a month, describing herself in the lease as "the owner", and dies six years in; the tenant claims the remaining four years against the bare owners. To apply Art. 2716 a model must first establish what a usufruct is, and that provision (Art. 535, a real right of limited duration on the property of another) sits 2,181 articles away, in a different Book of the code. The word *usufruct* is the only address on that path; no citation appears along it, so a citation-following extractor cannot build the chain. With the definition established, the second sentence of Art. 2716 decides the case: the lease ends with the usufruct, and because the lessor described herself as the owner, the tenant's recourse lies against her estate, not the bare owners.

A mixed chain (PRC Civil Code). An uncle gives his married niece a Beijing apartment worth RMB 3.2 million; the written document records the gift and her acceptance but says nothing about whether the apartment is hers alone, and on divorce her husband claims it as community property.

Art. 1062 classifies property acquired during marriage as community property and cites Art. 1063 for the exceptions, among them property assigned to one spouse by a gift contract. Whether this document is such a contract is not settled in the Book on Marriage and Family; the definition is Art. 657, in the Book on Contracts. One hop follows a citation and the next follows a term.

End-to-end PRC pilot. We generated 50 items from the PRC Civil Code with the unmodified pipeline and audited them with the locked v5d prompt, without retuning. Four practising lawyers and judges with PRC expertise reviewed all 50 items against the code. v5d marked at least one fact uncertain or incorrect on 9 of the 50 items (18%); of the remaining 41, experts judged 35 (85%, 95% CI 72–93%) usable exactly as generated, a complete census of that stratum. With judge and output budget held fixed, strict accuracy is 62% (code in context) and 46% (closed-book) for Sonnet 4.6, and 38% and 30% for GPT-5.4. The closed-book scores show that a widely reproduced civil code is partly answerable from memory, unlike the DNV rules (0% closed-book, Appendix P); this is why the legal domain is reported as a pilot of the construction method, with no leaderboard.