WHEN EVIDENCE IS MISSING: BENCHMARKING HALLUCINATION AND ABSTENTION FROM INCOMPLETE OBSERVATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

In real-world settings, document-grounded LLM agents must successfully navigate incomplete corpora, whereas most benchmarks assume all evidence is available. We introduce a scenario-agnostic simulation framework that generates document corpora based on a structured latent world. The framework maintains full provenance for all facts, their inference chain, and in which documents they are surfaced, allowing arbitrary subsets of documents to be masked and exactly recomputing which facts remain observable and inferable when certain evidence is missing. We instantiate our framework on four human-validated enterprise scenarios, each producing approximately 1,000 documents. Based on the fact-to-document provenance, we construct question-answering and structured prediction tasks requiring retrieval, multi-document synthesis, logical inference, and abstention when evidence is missing. Evaluating nine LLMs on our tasks using RAG and CLI agent harnesses, we find that models most accurate at answering when evidence is present differ from those that abstain most reliably when it is missing.

1 INTRODUCTION

Retrieval-augmented generation grounds large language model (LLM) responses in information retrieved from external documents, supporting tasks that require knowledge not represented in a model's own weights Lewis et al. (2020). Recent benchmarks have extended this setting to large corpora of heterogeneous documents, requiring LLM agents to locate and combine evidence from multiple sources Choubey et al. (2025). Completing such tasks on said corpora can require synthesizing dispersed facts, drawing evidence-supported conclusions, and reconstructing temporal relationships between events surfaced in the documents. A key component is that agents must also recognize the need to abstain when required evidence is missing. Although it is a key capability of modern-day agents, evaluating abstention is challenging as it requires fine-grained control over what evidence is accessible and whether it is sufficient to provide an answer Wen et al. (2025).

Existing benchmarks tackle important parts of this problem already: multi-hop QA benchmarks like HotpotQA Yang et al. (2018) test reasoning across multiple documents, and benchmarks such as MuSiQue Trivedi et al. (2022) include unanswerable questions. However, evaluating the performance of agents under changing available evidence requires an additional capability: tracking how evidence removal affects the ground truth answers of benchmark items. Removing a document does not necessarily remove all its facts globally, as another document may surface the same information. Losing a required fact for inference can make a conclusion no longer inferable. Answerability must therefore be calculated relative to the visible corpus and its remaining inference dependencies.

We address this requirement by simulating a latent world prior to surfacing the facts in a corpus of documents, while retaining full fact-to-document provenance. Tracking this enables us to evaluate whether agents successfully recover supported evidence and abstain when the required evidence is missing. Our key contributions are:

(1) **Scenario-agnostic simulation framework.** We introduce a specification-driven simulation framework that generates a synthetic latent world including entities, processes, events, and facts. Using the latent world, base facts are surfaced as a corpus of documents using an LLM.
(2) **Controllable evidence visibility.** The framework retains full fact-to-document provenance and

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.

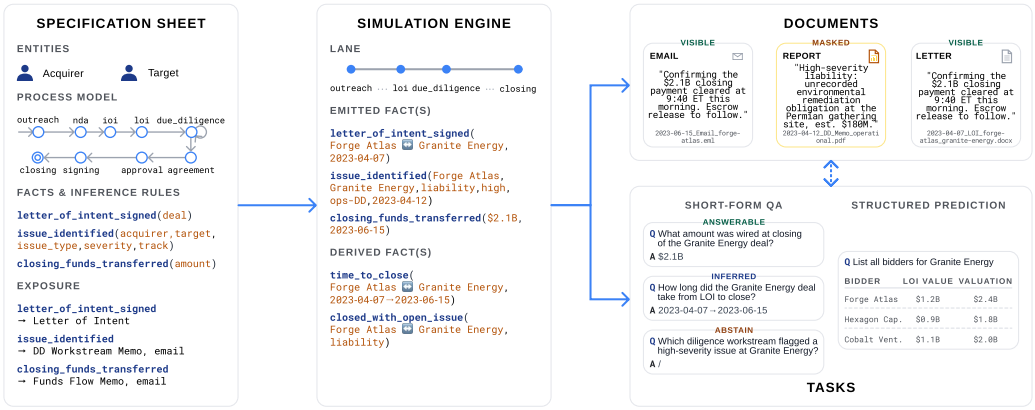

Figure 1: Overview of the framework components. Based on the *specification sheet*, a latent world is generated by the simulation engine. Facts are surfaced into documents and corresponding question answering and structured prediction tasks are generated, that are grounded in the corpus.

all inference dependencies for inferred facts, enabling dynamic document masking and automatic recomputation of the facts that remain observable or inferable. This directly updates the ground truth for our benchmark tasks without manually re-labeling after each masking change.
(3) **Multi-task benchmark and evaluation.** We instantiate the framework across four enterprise scenarios, producing 599 QA and 375 structured prediction tasks derived from the latent world. We evaluate nine LLMs on the tasks using RAG and CLI agent harnesses.

2 Related Work

Our work relates most closely to existing benchmarks on document-grounded QA, hallucination and abstention, and structured prediction from documents.

Question-answering benchmarks. Multi-hop QA benchmarks evaluate reasoning across documents Yang et al. (2018); Welbl et al. (2018); Ho et al. (2020); Trivedi et al. (2022), although some work has identified shortcuts in datasets, allowing evaluated models to answer questions without combining the intended evidence Trivedi et al. (2020). Other related benchmarks evaluate questions requiring information beyond the initial context provided Ferguson et al. (2020), changing world knowledge Vu et al. (2024); Park et al. (2024), and temporal reasoning Fatemi et al. (2024); Wang & Zhao (2024); Gruber et al. (2025); Wei et al. (2025); Kenneweg et al. (2026).

Long-context and multi-document benchmarks test how models understand and combine information across larger and more varied inputs (documents) Bai et al. (2024); Wang et al. (2024); Suri et al. (2025); Li et al. (2026). Retrieval-focused evaluations extend these settings to search (large) document collections Tang & Yang (2024), including email corpora Ryan et al. (2025), enterprise workflows Choubey et al. (2025); Sun et al. (2026), and open-ended deep research Abaskohi et al. (2025). Similarly to our approach, other works have approached the generation of evaluation data from synthetic worlds that span simulated narratives Labutov et al. (2018), encyclopedic document collections Gong et al. (2025), and enterprise communications Hamilton et al. (2026).

Our QA tasks combine retrieval and inference over simulated synthetic document corpora, where documents can be selectively masked. The framework additionally maintains fact-to-document provenance, recording which documents surface base facts and how inferred facts depend on them, allowing us to recompute answerability and ground truth of QA items automatically.

Hallucination & abstention benchmarks. Existing benchmarks assess factual correctness Lin et al. (2022); Wei et al. (2024) and verify model outputs Ravichander et al. (2025); Cheng et al. (2025), while others assess hallucination detection and attribution Li et al. (2023); Ji et al. (2025); Liu et al. (2026). Closest to our setting, context faithfulness and abstention benchmarks task models

with grounding their answer in provided evidence and abstaining when it is missing Ming et al. (2024); Song et al. (2025); Kirichenko et al. (2025). Other works have extended these settings to evidence gathering, refusal and controlled evidence removal in multi-hop reasoning and search Song et al. (2026); Xie et al. (2026); Wang et al. (2026a).

Our work focuses on abstention and intrinsic hallucination, termed by HalluLens Bang et al. (2025) as responses that contradict or lack the support from the context provided to the model. In our setting, support is assessed against the surfaced evidence/facts in the visible document corpus.

Structured prediction benchmarks. DTBench Guo et al. (2026) evaluates schema-guided document-to-table extraction. FinAuditing Wang et al. (2026b) evaluates semantic matching, relation extraction, and mathematical reasoning over semi-structured financial filings. Our structured prediction tasks ask an agent to synthesize facts and reconstruct processes and events as tables in response to a natural language query over a corpus of unstructured documents. The ground truth tables are derived from the simulated latent world and recomputed to reflect the facts, events, and processes recoverable from the visible set of documents.

3 World Generation

3.1 Specification Sheet

As outlined in Figure 1, a specification sheet (spec sheet) is provided as the input to the simulation framework. A spec sheet serves as a blueprint, passed to the simulator to produce a structured latent world. Spec sheets define scenarios across five dimensions: entity types and roles, process models, event templates, facts, and inference rules. The engine is scenario-agnostic and instantiating a new domain simply requires authoring a new spec sheet, either manually or with the help of an LLM. Below we outline the key components of a spec sheet.

Process model. The process model is a probabilistic finite automaton of an entity's lifecycle in a scenario. Starting off in an initial process, each process defines weighted and state-dependent transitions to the next until a terminal state is reached. To ensure more realistic scenarios can be modeled, the framework also supports branching/concurrency (running event tracks within a process in parallel) and repetition of processes.

Process lane. A process lane is one independent simulated instantiation of the process model for a set of entities and attributes, which are passed down to its processes and events (see Figure 2). A lane consists of a sequence of individual processes/steps, their events and facts, and its completion state. Examples of a lane include an investment deal, a new hire, or a product launch. The latent world and its corresponding document count scale by adding more lanes, with lane-specific entities resampled for each lane alongside entities shared across all lanes.

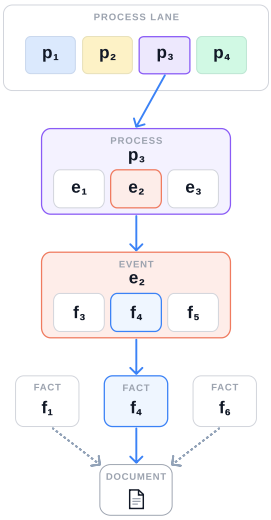

Figure 2: Simulation hierarchy: Process lanes consist of processes that contain events. Events emit facts that are surfaced in documents.

Process. A process is a single stage within its lane that emits a set of events (e.g., 'Due Diligence' in an M&A lane). Events are timestamped within their process' temporal window.

Event templates. Event templates define types of events that can occur in a process, along with the facts that the event emits.

Facts. A fact is an instantiation of a logical predicate that is considered true in the simulated world. Base facts are directly emitted by event templates and surfaced in documents. Inferred facts on the other hand, are only produced through inference rules and are not directly surfaced in documents (see subsection 3.3).

Inference rules. Inference rules synthesize facts to produce new inferred facts. Inferred facts test whether agents can combine information scattered across documents to derive conclusions not explicitly stated.

3.2 Simulation Engine

The simulator uses a spec sheet along with three auxiliary inputs to deterministically generate a world: a random seed, a start year, and the number of process lanes. The engine then proceeds to instantiate the declared entities, split across two types: localized lane entities that are resampled for new lanes (e.g., the acquired company in M&A process), and shared entities that persist across lanes (e.g., the acquiring company across multiple M&A processes). Every lane is assigned a start time, and the engine walks the process model until it hits a terminal state.

Each process consists of its unique set of event templates. Templates can run in sequence or in concurrent tracks and they can be configured to trigger either by a certain probability, once per entity-role type, or a sampled number of times within the process. Event templates can also depend on the current lane state, for example occurring only after one high-severity issue has occurred. When a template fires it produces events that emit one or more base facts, that are assigned a timestamp.

Emitted events update the latent lane state with certain fact types (e.g., enumerating the number of issues). The engine then dynamically shifts transition weights based on the latent lane state (a lane with severe issues is more likely to move from the due diligence phase into the renegotiation or termination phase). Loop caps ensure a process cannot be revisited an infinite number of times. This ensures lanes following the same spec can take different paths that are conditioned on their previously emitted facts.

The generated world records lanes, processes, events and facts. All facts have stable identifiers, predicates, arguments, and lane membership. Base facts additionally record timestamps and their generating events. Inferred facts instead record the generating rule, identifiers of their supporting facts, and start and end timestamps.

Forward chaining. After all base facts are emitted for a lane, the engine applies inference rules in specification order to produce inferred facts, allowing later rules to use facts inferred by earlier rules. Inference rules combine multiple facts to compute what facts must hold, even though they may not be explicitly emitted by an individual event. Inference rules derive facts by joining multiple predicates under value and temporal constraints. For example, a rule identifies a superseded bid when a later bid for the same entity is lower in value than the previous bid. These rules provide the ground truth for evaluating multi-document synthesis where agents must recover supporting facts from (multiple) documents and synthesize them to infer conclusions that are not explicitly surfaced.

3.3 Document Generation

The document generation stage occurs after the simulation engine completes latent world generation. The spec's exposure model lists which predicates each document type may carry. For every document type, the engine collects matching facts in a lane and groups them into documents with a greedy clustering pass. Facts are grouped based on temporal proximity and shared processes, events, entities, or topics. Each document is then assigned a date, a source, and a list of base facts it surfaces.

The document renderer is conditioned on the scenario description, the document definition, the involved entities, the date, and the exposed facts (each assigned one of four surfacing strengths). The planner then transforms each fact and its level into an instruction about how a fact should appear in a document, which is passed to an LLM to write the document as an email, letter, report, or spreadsheet. The following validation stage checks recorded evidence spans for all assigned facts and document-format constraints (e.g., the given email must be written from the sender's perspective). Failed documents are regenerated, up to a fixed number of attempts.

Document masking. In practice, agents often only have access to parts of the relevant evidence. Our framework simulates this by masking documents from the agent. Masking withholds a fraction of documents from the corpus, which hides evidence of certain base facts. The underlying facts remain true; however, a base fact becomes unobservable when no unmasked document surfaces it.

Document masking operates within each lane and supports two strategies: (1) window masking hides a contiguous run of documents in time (simulating missing temporal blocks of documents), and (2) random masking hides documents uniformly at random.

Fact visibility is propagated through relevant inferred facts. A base fact is visible if at least one unmasked document exposes it, while an inferred fact is visible if all of the prerequisite facts are visible. Hiding a single base fact can make multiple downstream facts uninferable. This setup ensures fact visibility is inferred and every QA item is labeled unanswerable when none of its matching facts remain observable or inferable from the corpus.

4 TASKS

All tasks are derived directly from the latent world produced by the simulation engine. Ground truth answers for QA items and tables for structured prediction items are computed programmatically using the document visibility. We generate four task types across our two categories: question answering (1) and structured prediction (3). Structured prediction consists of the Synthesis, Process Reconstruction, and Event Reconstruction tasks. Samples are provided in Appendix E.

4.1 QUESTION ANSWERING

Question-Answer items cover both base facts surfaced in documents and inferred facts produced by the specification's inference rules. Answers include entity names, numerical quantities, dates, temporal intervals, and categorical values. We only retain QA items whose matching facts yield exactly one canonical answer in the full latent world, thus excluding ambiguous questions. An item is defined as answerable after masking iff at least one matching fact is observable or inferable from the visible document corpus; otherwise, the expected answer is abstention. In this setup, agents must recover factual information, synthesize supporting evidence where inference is required, and distinguish between cases of sufficient and insufficient evidence.

4.2 STRUCTURED PREDICTION

Structured prediction items require agents to produce tables with pre-specified columns in response to a natural language query. Table cells can contain textual values, numerical quantities, dates, or Boolean indicators. Queries are scoped either to an individual lane or across the entire scenario. Synthesis tasks collect and aggregate information, such as bidders and their valuations for a company acquisition surfaced in documents, and can also include inferred facts. Process Reconstruction lists observable processes alongside the dates of their first and last visible documents. Event Reconstruction counts event occurrences by template within each lane; here, an event is considered observable when at least one of its emitted facts is surfaced and visible. Masking can have different effects depending on the task. It may leave individual cells undefined, or change counts and inference indicators. The evaluated agents are expected to submit the requested table, or an empty table when no rows match the input query.

5 VALIDATION

5.1 PROGRAMMATIC VALIDATION

Before generation, we validate each spec sheet against a JSON schema and check process transitions and reachability, entity references, event predicates and arguments, document exposure mappings, and inference rule inputs. QA templates are checked to ensure they reference base or inferred fact types and use valid fields for constructing questions and defining their ground truth answers. A separate simulation-based check flags inference rules that do not fire in any sampled process lane.

During document generation, automated checks verify that each assigned fact has a recorded evidence span matching the generated text and that the overall document meets its formatting constraints. Failed document generations are retried up to a retry limit. After generation, coverage checks identify base facts missing from the fact-to-document mappings and flag entity names that

are absent in their respective lane documents. In combination, these checks assess the structural consistency of the latent world and its fact coverage.

5.2 Human Validation

To ensure scenario realism, we asked four subject-matter experts with professional experience in the corresponding domains or with the relevant data sources to review each spec sheet and one lane produced from that specification, separate from the final corpora. We used their feedback to refine the specifications, making the scenarios, documents, and supporting evidence more consistent with real-world workflows.

As the ground truth labels for our benchmark items are derived from the latent world, their grounding depends on whether the generated documents surface their assigned facts faithfully. We evaluated this through a human annotation study involving five annotators, each reviewing the same 200 documents across the four scenarios. Each scenario contributed 45 randomly sampled documents to which we added an additional five negative-control documents presented exclusively with distractor facts. Annotators were tasked to select each candidate fact supported by the presented document. Across all document-fact annotations, annotators selected 77.6% of assigned facts correctly and rejected 98.9% of distractor statements successfully, with inter-annotator agreement of Fleiss' $\kappa = 0.765$. We provide the complete annotation protocol, annotation interface, agreement statistics, and scenario breakdown in Appendix F.

6 Experiments

6.1 Setup

Corpora. The dataset consists of four scenarios: Mergers & Acquisitions (M&A), Employee Lifecycle, Sales Lifecycle, and Venture Capital (VC) Fundraising. Each corpus contains approximately 1,000 documents spanning emails, letters, reports, and spreadsheets, and also including scenario-specific document subtypes (e.g. due-diligence reports). For our experiments we mask 204, 202, 218, and 196 documents, respectively. Across these scenarios, we evaluate 974 tasks: 599 question answering and 375 structured prediction tasks. A detailed breakdown is provided in Table 1.

Table 1: Dataset summary statistics. Document counts are reported before masking. QA denotes question answering, and SP denotes structured prediction.

		# Documents					# Tasks	
Scenario	**Lanes**	**Total**	**Emails**	**Letters**	**Reports**	**Spreadsheets**	**QA**	**SP**
Mergers & Acquisitions	20	1021	675	148	161	37	217	104
Employee Lifecycle	8	1019	898	7	67	47	149	75
Sales Lifecycle	19	1087	804	53	230	0	126	136
VC Fundraising	27	1002	844	49	104	5	107	60

Models. We evaluate our tasks across the four scenarios on nine models in total. We evaluate six closed-source models: Opus 5, Sonnet 5, and Haiku 4.5 from Anthropic and the GPT-5.6 family (Sol, Terra, Luna) from OpenAI. We also evaluate three open-weight models across a variety of model sizes and authors: DeepSeek-V4-Flash (284B), GPT-OSS-120B, and Qwen3.6-27B. Detailed configuration parameters are provided in Appendix A.

Harnesses. We evaluate each model under two harnesses with an 80 turn limit: RAG Lewis et al. (2020) is based on retrieving from an indexed corpus and CLI is based on exploring files directly through shell commands. Both harnesses access the same visible document corpus (excl. masked documents) for the relevant scenario.

The RAG harness provided a retrieve tool that accepted a text query and a requested number of results k, using cosine similarity search on an index of document chunks. Documents were split into paragraph-aware chunks of approximately 2000 characters with 200 character overlap, and

each chunk was embedded using OpenAI's `text-embedding-3-small`. Each query returned the five highest-scoring chunks by default, but we allowed the agents to request between one and twenty. Results included the document ID, chunk ID, and similarity score.

The CLI harness provided a bash tool to execute shell commands inside an isolated Docker container, with all visible documents for each scenario mounted in a read-only directory. Each command returned its output, error messages, and exit status.

Both harnesses provided a dedicated submission tool. For QA tasks, agents submitted an answer or indicated abstention. For structured prediction tasks, agents submitted table rows, or an empty table when no rows matched the query. The full prompts used for each combination of harness and task are provided in Appendix C.

Judging. We evaluate responses deterministically against ground truth reference answers and tables, accepting answers exclusively through the submission tools. Submitted responses are normalized by case, whitespace, punctuation, dates, and numeric representations. The full normalization and matching procedure is outlined in Appendix A.

For answerable QA items, we report correct answers, false refusals (explicit abstention) and incorrect answers. Incorrect answers include both incorrect submitted answers and terminal model failures (turn/context limits or no submission). For unanswerable QA items we report correct abstentions, hallucinations (non-abstaining submission), and terminal model failures. Each rate is calculated within each answerability group and pooled across scenarios.

For the structured prediction tasks, we report cell precision, recall and F_1 separately for the three subsets: Synthesis, Process Reconstruction, and Event Reconstruction. Within each subset, we average task-level precision and recall over all tasks across all four scenarios and compute F_1 as their harmonic mean.

Compute. All experiments were conducted on a server operating Ubuntu 24.04 LTS with two 128-core AMD EPYC 9754 processors, 1.5 TB of DDR5 memory, and four NVIDIA H200 GPUs. All open-weight models were served using vLLM on this machine and all closed-weight models were accessed via their respective provider APIs.

6.2 Results

Strong performance on answerable questions does not imply reliable abstention. Using the CLI harness, Qwen 3.6 leads answerable accuracy at 81.6% but only abstains correctly on 27.9% of unanswerable questions. Opus 5 shows the opposite, abstaining correctly on 77.2% of unanswerable questions, while struggling with answerable accuracy at 43.2%. Under the RAG harness, the leaders differ again: GPT-OSS leads answerable accuracy and struggles with correct abstention, whereas Sonnet 5 has the highest correct abstention and lowest hallucination rate, and only performs moderately on answerable questions. The difference is not explained by frequent abstention on answerable questions. Even Opus 5 using the CLI harness falsely refuses 6.0% of answerable items, while submitting incorrect or no answers on 50.8%.

Questions requiring inference expose larger performance differences. As expected, across both harnesses every model performs worse on questions about inferred facts than on directly surfaced base facts. However, the performance gap varies drastically among models. Under CLI, Opus 5 falls from 55.4% accuracy to 9.1% after one inference step and drops to 0% after two. We observe similar results for Opus 5 under the RAG harness. Conversely, Qwen 3.6 falls from 84.6% to 75.8% after one inference step, and 57.9% after two under CLI. A full inference-depth breakdown is presented in Table 8.

Harness effects vary across models and include terminal failures. Neither harness consistently improves both answerable accuracy and correct abstention. Switching to RAG from CLI increases answerable accuracy for five of the nine models, while only improving abstention for three. For example, Opus 5's correct abstention rate drops from 77.2% to 57.4%, whereas DeepSeek-V4-Flash's rate increases from 38.2% to 45.6%. Harness differences also affect task completion. Within the bounds of our evaluation, Qwen 3.6 terminates without an answer on 56.6% of unanswerable ques-

Table 2: Question answering results. For answerable items, Incorrect includes incorrect answers and terminal model failures. For unanswerable items, Incorrect denotes terminal model failures without a submitted answer. Under both harnesses, the models with the highest answerable accuracy differ from those that abstain most reliably on unanswerable items. **Bold** denotes the best score among all models for each harness.

		Answerable			Unanswerable		
Harness	**Model**	**Correct**	**False Refusal**	**Incorrect**	**Correct**	**Hallucination**	**Incorrect**
CLI	Opus 5	.432	.060	.508	**.772**	**.228**	**.000**
	Sonnet 5	.499	.058	.443	.618	.382	**.000**
	Haiku 4.5	.743	.015	.242	.404	.559	.037
	GPT-5.6 Sol	.540	**.002**	.458	.228	.772	**.000**
	GPT-5.6 Terra	.501	**.002**	.497	.250	.750	**.000**
	GPT-5.6 Luna	.495	**.002**	.503	.316	.662	.022
	DeepSeek-V4-F	.775	.006	.218	.382	.559	.059
	GPT-OSS-120B	.758	.004	.238	.243	.706	.051
	Qwen3.6-27B	**.816**	.004	**.179**	.279	.449	.272
RAG	Opus 5	.436	.035	.529	.574	.426	**.000**
	Sonnet 5	.568	.060	.371	**.625**	**.375**	**.000**
	Haiku 4.5	.698	.030	.272	.412	.588	**.000**
	GPT-5.6 Sol	.557	.002	.441	.213	.787	**.000**
	GPT-5.6 Terra	.570	**.000**	.430	.184	.816	**.000**
	GPT-5.6 Luna	.499	**.000**	.501	.257	.743	**.000**
	DeepSeek-V4-F	.691	.037	.272	.456	.544	**.000**
	GPT-OSS-120B	**.734**	.011	**.255**	.162	.757	.081
	Qwen3.6-27B	.672	**.000**	.328	.037	.397	.566

tions under RAG, compared to 27.2% under CLI. Therefore, Qwen 3.6's lower hallucination rate under RAG does not indicate more reliable abstention. A detailed breakdown of failure modes is provided in Table 11.

Table 3: Structured prediction results, reporting precision (Prec), recall (Rec), and F_1 score. The GPT-5.6 model family achieves the highest F_1 in every category under both harnesses, while process and event reconstruction scores remain substantially lower than synthesis scores. **Bold** denotes the best score among all models for each harness.

		Synthesis			Process			Events		
Harness	**Model**	**Prec**	**Rec**	**F_1**	**Prec**	**Rec**	**F_1**	**Prec**	**Rec**	**F_1**
CLI	Opus 5	.728	.813	.768	**.339**	.293	.314	.000	.000	.000
	Sonnet 5	.730	.812	.769	.154	.134	.143	.142	.045	.068
	Haiku 4.5	.765	.829	.795	.070	.067	.068	.055	.020	.029
	GPT-5.6 Sol	.748	**.865**	.802	.331	**.302**	**.316**	**.292**	**.064**	**.105**
	GPT-5.6 Terra	**.768**	.856	**.810**	.317	.299	.308	.281	.055	.093
	GPT-5.6 Luna	.742	.841	.788	.292	.282	.287	.276	.060	.099
	DeepSeek-V4-F	.735	.849	.788	.141	.133	.137	.065	.023	.034
	GPT-OSS-120B	.622	.767	.687	.029	.036	.032	.137	.047	.070
	Qwen3.6-27B	.737	.799	.767	.037	.033	.035	.107	.031	.048
RAG	Opus 5	.739	.818	.777	.318	.294	.305	.000	.000	.000
	Sonnet 5	.758	.825	.790	.097	.087	.092	.000	.000	.000
	Haiku 4.5	.740	.809	.773	.074	.066	.069	.024	.005	.008
	GPT-5.6 Sol	**.766**	**.868**	**.814**	**.336**	**.320**	**.328**	**.361**	**.106**	**.164**
	GPT-5.6 Terra	**.766**	.861	.811	.242	.225	.233	.079	.010	.018
	GPT-5.6 Luna	.738	.850	.790	.178	.178	.178	.269	.033	.059
	DeepSeek-V4-F	.744	.842	.790	.135	.127	.131	.038	.013	.019
	GPT-OSS-120B	.684	.794	.735	.067	.070	.069	.044	.012	.019
	Qwen3.6-27B	.545	.575	.559	.035	.030	.032	.052	.009	.016

The GPT-5.6 model family leads structured prediction performance. The GPT-5.6 model family achieves the highest F_1 score in every structured prediction category under both harnesses. Terra leads Synthesis under CLI, while Sol leads the remaining harness and task categories in F_1 score. However, Process and Event Reconstruction results peak at F_1 scores of .328 and .164 respectively. Sol's Event Reconstruction precision of .361 under RAG demonstrates partial success on the task, although recall remains low at .106.

Synthesis scores conceal a gap between empty and populated tables. While models achieve substantially higher scores on Synthesis, it is important to point out that this task combines constructing populated tables with returning empty tables. Of the 297 Synthesis tasks, 162 have empty ground truth tables. Restricting the evaluation to the remaining 135 nonempty tasks lowers the best F_1 score to .698, compared to .814 when considering all tasks. We report a full breakdown of results in the appendix in Table 9.

6.3 Ablation Studies

Oracle evidence ablation. We rerun the 463 answerable QA items with each harness restricted to the unmasked documents surfacing the question's supporting base facts, as determined through provenance and inference dependencies. All other settings remain the same. Accuracy improves in 15 of the 18 model and harness combinations, with Qwen 3.6 under RAG leading with a 12.1 percentage point (pp) increase. We find that restricting access to just the supporting documents alone does not resolve the remaining QA errors. Full oracle results appear in the appendix in Table 5.

Relaxed visibility ablation. We rescore the existing QA responses under a relaxed visibility rule that treats inferred facts as visible when at least 50% of their prerequisite facts are visible. This reclassifies 46 previously unanswerable questions as answerable. We analyze balanced accuracy, which changes at most 2.7 percentage points. Performance rankings after the intervention change only slightly with one model swap per harness, suggesting relative model performance is largely stable under this change. Full results appear in the appendix in Table 7.

7 Conclusion

We introduced a scenario-agnostic simulation framework that generates document corpora from a structured latent world. Fact-to-document provenance and inference dependencies enable document masking and automatic recomputation of ground truth for question answering and structured prediction tasks. Across nine LLMs using RAG and CLI agent harnesses, we find that models answering most accurately on answerable questions differ from models that abstain most reliably on unanswerable questions, when the required evidence is missing. We also find that Process and Event Reconstruction remain challenging for all evaluated models under both harnesses.

Limitations. Our evaluation is limited to four synthetic enterprise scenarios, each containing approximately 1,000 documents. Scenario realism and task difficulty depend highly on the authored spec sheets and their included inference rules. Our notion of answerability is determined relative to the visible (unmasked) document corpus. While our human annotation study validates fact surfacing in documents, it does not establish whether humans could answer every single question labeled as unanswerable from the remaining evidence. Finally, our harness comparison does not determine which differences in retrieval or tool access explain the observed performance gaps.

Future work. Future work could extend evaluation to additional domains outside of the enterprise setting, larger document corpora, or more complex inference rules. An additional targeted human study could examine whether questions labeled as unanswerable can nevertheless be answered using evidence from the visible corpus through commonsense reasoning beyond the specified inference rules. Lastly, more fine-grained comparisons of retrieval and tool access could help explain the observed differences between harnesses.

AI USE STATEMENT

We used AI tools to assist us in speeding up the development process and improve code documentation. All AI-assisted code was reviewed by at least one author who did not produce the code in review. Our datasets are synthetic in nature and the document corpus produced from the latent world is generated by an LLM, before human annotators to validate its contents. AI was used to draft portions of the paper, yet, all text in this work was written by hand. All figures were designed by hand and did not use an AI assistance. AI was not used for any other tasks with required disclosure. All authors take full responsibility for the content of this work and its claims.

ETHICS STATEMENT

All data in this work is synthetic, with person and company names drawn from a random name generator. No personal data or proprietary information was used in our research or generated corpus. All annotators involved in our human validation study were colleagues who participated voluntarily.

We see limited potential for misuse of our datasets. Our generation pipeline does not target any real organization, and producing convincing fake documents has few to no ethical concerns.

REPRODUCIBILITY STATEMENT

We release the code, generated document corpora, benchmark tasks, and evaluation outputs used in this work here: `https://anonymous.4open.science/r/when-evidence-is-missing-ICLR-26FA/`. The latent world of each scenario can be regenerated deterministically using the released implementation and the same spec sheet, seed, start year, and lane count. Document generation and model evaluation involve non-deterministic LLM outputs that may vary between runs. Our released artifacts preserve the exact corpora used for evaluation and all outputs underlying our reported results. Dataset generation settings and model configurations are provided in Appendix A.

REFERENCES

Amirhossein Abaskohi, Tianyi Chen, Miguel Muñoz-Mármol, Curtis Fox, Amrutha Varshini Ramesh, Étienne Marcotte, Xing Han Lù, Nicolas Chapados, Spandana Gella, Christopher Pal, Alexandre Drouin, and Issam H. Laradji. DRBench: A Realistic Benchmark for Enterprise Deep Research. October 2025. URL https://openreview.net/forum?id=IGYQ4c92e2.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench: A Bilingual, Multitask Benchmark for Long Context Understanding. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3119–3137, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.172. URL https://aclanthology.org/2024.acl-long.172/.

Yejin Bang, Ziwei Ji, Alan Schelten, Anthony Hartshorn, Tara Fowler, Cheng Zhang, Nicola Cancedda, and Pascale Fung. HalluLens: LLM Hallucination Benchmark. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24128–24156, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1176. URL https://aclanthology.org/2025.acl-long.1176/.

Aileen Cheng, Alon Jacovi, Amir Globerson, Ben Golan, Charles Kwong, Chris Alberti, Connie Tao, Eyal Ben-David, Gaurav Singh Tomar, Lukas Haas, Yonatan Bitton, Adam Bloniarz, Aijun Bai, Andrew Wang, Anfal Siddiqui, Arturo Bajuelos Castillo, Aviel Atias, Chang Liu, Corey Fry, Daniel Balle, Deepanway Ghosal, Doron Kukliansky, Dror Marcus, Elena Gribovskaya, Eran Ofek, Honglei Zhuang, Itay Laish, Jan Ackermann, Lily Wang, Meg Risdal, Megan Barnes,

Michael Fink, Mohamed Amin, Moran Ambar, Natan Potikha, Nikita Gupta, Nitzan Katz, Noam Velan, Ofir Roval, Ori Ram, Polina Zablotskaia, Prathamesh Bang, Priyanka Agrawal, Rakesh Ghiya, Sanjay Ganapathy, Simon Baumgartner, Sofia Erell, Sushant Prakash, Thibault Sellam, Vikram Rao, Xuanhui Wang, Yaroslav Akulov, Yulong Yang, Zhen Yang, Zhixin Lai, Zhongru Wu, Anca Dragan, Avinatan Hassidim, Fernando Pereira, Slav Petrov, Srinivasan Venkatachary, Tulsee Doshi, Yossi Matias, Sasha Goldshtein, and Dipanjan Das. The FACTS Leaderboard: A Comprehensive Benchmark for Large Language Model Factuality, December 2025. URL http://arxiv.org/abs/2512.10791. arXiv:2512.10791 [cs].

Prafulla Kumar Choubey, Xiangyu Peng, Shilpa Bhagavath, Kung-Hsiang Huang, Caiming Xiong, and Chien-Sheng Wu. Benchmarking Deep Search over Heterogeneous Enterprise Data. In Saloni Potdar, Lina Rojas-Barahona, and Sebastien Montella (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 501–517, Suzhou (China), November 2025. Association for Computational Linguistics. ISBN 979-8-89176-333-3. doi: 10.18653/v1/2025.emnlp-industry.34. URL https://aclanthology.org/2025.emnlp-industry.34/.

Bahare Fatemi, Mehran Kazemi, Anton Tsitsulin, Karishma Malkan, Jinyeong Yim, John Palowitch, Sungyong Seo, Jonathan Halcrow, and Bryan Perozzi. Test of Time: A Benchmark for Evaluating LLMs on Temporal Reasoning. October 2024. URL https://openreview.net/forum?id=44CoQe6VCq.

James Ferguson, Matt Gardner, Hannaneh Hajishirzi, Tushar Khot, and Pradeep Dasigi. IIRC: A Dataset of Incomplete Information Reading Comprehension Questions. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1137–1147, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.86. URL https://aclanthology.org/2020.emnlp-main.86/.

Albert Gong, Kamilė Stankevičiūtė, Chao Wan, Anmol Kabra, Raphael Thesmar, Johann Lee, Julius Klenke, Carla P. Gomes, and Kilian Q. Weinberger. PhantomWiki: On-Demand Datasets for Reasoning and Retrieval Evaluation. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 19964–19995. PMLR, October 2025. URL https://proceedings.mlr.press/v267/gong25d.html.

Raphael Gruber, Abdelrahman Abdallah, Michael Färber, and Adam Jatowt. ComplexTempQA: A 100m Dataset for Complex Temporal Question Answering. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9100–9112, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.463. URL https://aclanthology.org/2025.emnlp-main.463/.

Yuxiang Guo, Zhuoran Du, Nan Tang, Kezheng Tang, Congcong Ge, and Yunjun Gao. DTBench: A Synthetic Benchmark for Document-to-Table Extraction. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '26, pp. 8942–8953, New York, NY, USA, August 2026. Association for Computing Machinery. ISBN 979-8-4007-2259-2. doi: 10.1145/3770855.3817516. URL https://dl.acm.org/doi/10.1145/3770855.3817516.

Sil Hamilton, Albert Yu Sun, Oscar J. Romero, Carl-Leander Henneking, David Mimno, Bishan Yang, and Igor Labutov. CorporateBench: Large-Scale Q&A Benchmarking with Temporal Knowledge Bases, August 2026. URL http://arxiv.org/abs/2608.27391. arXiv:2608.27391 [cs.AI].

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing A Multi-hop QA Dataset for Comprehensive Evaluation of Reasoning Steps. In Donia Scott, Nuria Bel, and Chengqing Zong (eds.), *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.580. URL https://aclanthology.org/2020.coling-main.580/.

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 D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference. Curran Associates, Inc., 2025. doi: 10.52202/085713-4235. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/b8badadce3f482ba340ff870f4894441-Paper-Datasets_and_Benchmarks_Track.pdf.

Svenja Kenneweg, Jörg Deigmöller, Philipp Cimiano, and Julian Eggert. TRAVELER: A Benchmark for Evaluating Temporal Reasoning Across Vague, Implicit and Explicit References. *SN Computer Science*, 7(5), April 2026. doi: 10.1007/s42979-026-04973-y. URL https://doi.org/10.1007/s42979-026-04973-y.

Polina Kirichenko, Mark Ibrahim, Kamalika Chaudhuri, and Samuel Bell. AbstentionBench: Reasoning LLMs Fail on Unanswerable Questions. October 2025. URL https://openreview.net/forum?id=OkHC30LLpO.

Igor Labutov, Bishan Yang, Anusha Prakash, and Amos Azaria. Multi-Relational Question Answering from Narratives: Machine Reading and Reasoning in Simulated Worlds. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 833–844, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1077. URL https://aclanthology.org/P18-1077/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6449–6464, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.397. URL https://aclanthology.org/2023.emnlp-main.397/.

Zhanli Li, Yixuan Cao, Lvzhou Luo, and Ping Luo. Navigating Large-Scale Document Collections: MuDABench for Multi-Document Analytical QA. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 6877–6898, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.341. URL https://aclanthology.org/2026.findings-acl.341/.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring How Models Mimic Human Falsehoods. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL https://aclanthology.org/2022.acl-long.229/.

Xuannan Liu, Xiao Yang, Zekun Li, Peipei Li, and Ran He. AgentHallu: Benchmarking Automated Hallucination Attribution of LLM-based Agents, January 2026. URL http://arxiv.org/abs/2601.06818. arXiv:2601.06818 [cs].

Yifei Ming, Senthil Purushwalkam, Shrey Pandit, Zixuan Ke, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. FaithEval: Can Your Language Model Stay Faithful to Context, Even If "The Moon is Made of Marshmallows". October 2024. URL https://openreview.net/forum?id=UeVx6L59fg.

Yein Park, Chanwoong Yoon, Jungwoo Park, Donghyeon Lee, Minbyul Jeong, and Jaewoo Kang. ChroKnowledge: Unveiling Chronological Knowledge of Language Models in Multiple Domains. October 2024. URL https://openreview.net/forum?id=whaO3482bs.

Abhilasha Ravichander, Shrusti Ghela, David Wadden, and Yejin Choi. HALoGEN: Fantastic LLM Hallucinations and Where to Find Them. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1402–1425, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.71. URL https://aclanthology.org/2025.acl-long.71/.

Michael J. Ryan, Danmei Xu, Chris Nivera, and Daniel Campos. EnronQA: Towards Personalized RAG over Private Documents, May 2025. URL http://arxiv.org/abs/2505.00263. arXiv:2505.00263 [cs.IR].

Maojia Song, Shang Hong Sim, Rishabh Bhardwaj, Hai Leong Chieu, Navonil Majumder, and Soujanya Poria. Measuring and Enhancing Trustworthiness of LLMs in RAG through Grounded Attributions and Learning to Refuse. *International Conference on Learning Representations*, 2025:30783–30823, May 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/4c88827decab6c046b881a2c3a99c76f-Abstract-Conference.html.

Maojia Song, Liu Renhang, Xinyu Wang, Yong Jiang, Pengjun Xie, Fei Huang, Soujanya Poria, and Jingren Zhou. Demystifying Deep Search: A Holistic Evaluation with Hint-free Multi-Hop Questions and Factorised Metrics. *International Conference on Learning Representations*, 2026:95487–95512, April 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/9a27f07ee41dd59d527062e045c16353-Abstract-Conference.html.

Yuhong Sun, Joachim Rahmfeld, Chris Weaver, Weijia Chen, Roshan Desai, Wenxi Huang, and Mark H. Butler. EnterpriseRAG-Bench: A RAG Benchmark for Company Internal Knowledge, May 2026. URL http://arxiv.org/abs/2605.05253. arXiv:2605.05253 [cs.IR].

Manan Suri, Puneet Mathur, Franck Dernoncourt, Kanika Goswami, Ryan A. Rossi, and Dinesh Manocha. VisDoM: Multi-Document QA with Visually Rich Elements Using Multimodal Retrieval-Augmented Generation. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6088–6109, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.310. URL https://aclanthology.org/2025.naacl-long.310/.

Yixuan Tang and Yi Yang. MultiHop-RAG: Benchmarking Retrieval-Augmented Generation for Multi-Hop Queries. August 2024. URL https://openreview.net/forum?id=t4eB3zYWBK.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Is Multihop QA in DiRe Condition? Measuring and Reducing Disconnected Reasoning. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8846–8863, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.712. URL https://aclanthology.org/2020.emnlp-main.712/.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop Questions via Single-hop Question Composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL https://aclanthology.org/2022.tacl-1.31/.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. FreshLLMs: Refreshing Large Language Models with Search Engine Augmentation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13697–13720,

Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.813. URL https://aclanthology.org/2024.findings-acl.813/.

Hang Wang, Hang Dong, Lu Liu, and Chuanru Ren. MissDiag: Diagnostic Evaluation of Incomplete-Knowledge Robustness in KGQA and KG-RAG, August 2026a. URL http://arxiv.org/abs/2608.18489. arXiv:2608.18489 [cs.CL].

Minzheng Wang, Longze Chen, Fu Cheng, Shengyi Liao, Xinghua Zhang, Bingli Wu, Haiyang Yu, Nan Xu, Lei Zhang, Run Luo, Yunshui Li, Min Yang, Fei Huang, and Yongbin Li. Leave No Document Behind: Benchmarking Long-Context LLMs with Extended Multi-Doc QA. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5627–5646, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.322. URL https://aclanthology.org/2024.emnlp-main.322/.

Yan Wang, Keyi Wang, Shanshan Yang, Jaisal Patel, Jeff Zhao, Fengran Mo, Xueqing Peng, Lingfei Qian, Yankai Chen, Víctor Gutiérrez-Basulto, Jimin Huang, Guojun Xiong, Xiao-Yang Liu, and Jian-Yun Nie. FinAuditing: A Financial Taxonomy-Structured Multi-Document Benchmark for Evaluating LLMs. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '26, pp. 3456–3463, New York, NY, USA, July 2026b. Association for Computing Machinery. ISBN 979-8-4007-2599-9. doi: 10.1145/3805712.3808578. URL https://dl.acm.org/doi/10.1145/3805712.3808578.

Yuqing Wang and Yun Zhao. TRAM: Benchmarking Temporal Reasoning for Large Language Models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 6389–6415, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.382. URL https://aclanthology.org/2024.findings-acl.382/.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models, November 2024. URL http://arxiv.org/abs/2411.04368. arXiv:2411.04368 [cs] version: 1.

Shaohang Wei, Wei Li, Feifan Song, Wen Luo, Tianyi Zhuang, Haochen Tan, Zhijiang Guo, and Houfeng Wang. TimE: A Multi-level Benchmark for Temporal Reasoning of LLMs in Real-World Scenarios. October 2025. URL https://openreview.net/forum?id=su3Ch8Ia4a.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. Constructing Datasets for Multi-hop Reading Comprehension Across Documents. *Transactions of the Association for Computational Linguistics*, 6:287–302, May 2018. ISSN 2307-387X. URL https://transacl.org/ojs/index.php/tacl/article/view/1325.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. Know Your Limits: A Survey of Abstention in Large Language Models. *Transactions of the Association for Computational Linguistics*, 13:529–556, 2025. doi: 10.1162/tacl_a_00754. URL https://aclanthology.org/2025.tacl-1.26/.

Jianan Xie, Xin Sun, Zhongqi Chen, Xing Zheng, Qiang Liu, and Bowen Song. HopRefusalBench: Diagnosing Refusal Failures in Search-Augmented Agents for Multi-Hop Reasoning, August 2026. URL http://arxiv.org/abs/2608.01358. arXiv:2608.01358 [cs.CL].

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 Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, Brussels, Belgium, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

A Experimental Setup

A.1 Datasets

Using the provided code, the following seed & lane combinations were used to generate the datasets we release as part of this work: M&A (seed: 42, lanes: 20), Employee Lifecycle (seed: 1, lanes: 8), Sales Lifecycle (seed: 42, lanes: 19), and, Venture Fundraising (seed: 42, lanes: 27). With these combinations, all datasets can be re-generated.

A.2 Model Configuration

Table 4 outlines all the configuration parameters used for our experiments.

Table 4: Model and evaluation configuration. Unspecified sampling and reasoning settings use provider defaults. QA and SP caps represent per turn model response limits.

Model	Context	Reasoning	Sampling	Max Turns	QA Cap	SP Cap
Opus 5	1,000,000	High	Default	80	16,384	65,536
Sonnet 5	1,000,000	High	Default			
Haiku 4.5	200,000	Default	Default			
GPT-5.6 Sol	1,050,000	High	Default			
GPT-5.6 Terra	1,050,000	High	Default			
GPT-5.6 Luna	1,050,000	High	Default			
DeepSeek-V4-F	262,144	High	temperature=1.0 top-p=1.0 seed=42			
GPT-OSS-120B	131,072	High	seed=42			
Qwen3.6-27B	262,144	High	temperature=1.0 top-p=0.95 top-k=20 seed=42			

A.3 Judging

QA and structured prediction tasks are scored by a deterministic program, without any LLM judge involved. Predictions and gold answers are normalized before comparison. Dates are rewritten as `YYYY-MM-DD`, texts are lowercased, ampersands become *and*, dashes become spaces, and excess whitespace is collapsed.

In QA tasks, a prediction is correct if it matches any accepted gold answer, post-normalization. We also accept lenient matches if any of the following conditions are met: token containment in either direction (if an agent inserts an answer in a sentence), numeric equality after stripping currency and percent symbols, and duration/range equivalence. Agents answer through a submission tool with an abstain flag. An unanswerable question is correct only if the agent abstains. We report accuracy over answerable and unanswerable questions separately.

In structured prediction tasks, predicted rows are deduplicated and aligned to gold rows by key, with each gold row claimed at most once. We then compute precision and recall over the non-key cells. Precision and recall are averaged within each of the three categories (Synthesis, Process, Events).

B Scoring Details

B.1 Normalization

A single normalizer is applied to predictions and gold answers before any comparison. It applies the following steps in order:

1. Strip leading and trailing whitespace.
2. If the string contains a four-digit year (`19xx` or `20xx`), try to parse it as a date: first as ISO 8601, then as an RFC 2822 date, then against a fixed list of common formats such as `April 7, 2024`, `7 Apr 2024`, and `04/07/2024`. If any parse succeeds, the result is emitted as `YYYY-MM-DD` and the remaining steps are skipped.
3. Lowercase.
4. Replace `&` with *and*.
5. Replace dashes and underscores with spaces.
6. Collapse runs of whitespace into a single space.

B.2 SHORT-FORM QA

Each QA task carries a list of accepted gold answers (empty for unanswerable questions). Agents submit through a tool that takes an explicit `abstain` flag and an answer string. If the agent abstains, the task is correct if and only if the gold list is empty. If the agent answers, the task is correct if and only if the answer matches one of the (non-empty) gold answers under the checks below.

Exact match. The normalized prediction equals a normalized gold answer.

Token containment. A prediction is accepted if every gold token (split by whitespace) occurs in the normalized prediction string, so *"the LOI was signed on 2024-04-07"* matches a gold of *"2024-04-07"*. It is also accepted if every prediction token occurs in the normalized gold string. Neither direction applies when the gold answer is purely numeric, and the second direction also does not apply when the prediction is purely numeric, (*5* does not match *550*).

Numeric equality. Both sides are stripped of `$`, `,`, and `%` and parsed as floating-point numbers. The prediction is accepted if it parses and equals any gold answer that also parses (*$1,200* matches *1200*).

Range and duration equivalence. Strings are split on range separator (arrow, em/en dash, *to*, or *through*) into two endpoints. A predicted range matches a gold range if both endpoints agree after normalization. A predicted duration of the form *N days*, *N weeks*, or *N months* (a week counts as 7 days, a month as 30) matches a gold range whose endpoints are ISO dates if the predicted number of days differs from the span of the gold range by less than one day. This ensures *70 days* matches *2024-04-07 to 2024-06-15*.

B.3 STRUCTURED PREDICTION

Each structured-prediction task declares a header list and a subset of key columns. Gold and predicted rows are dictionaries over the headers. A prediction that is not a list of rows scores zero precision and zero recall.

Cell normalization. Cells are normalized with a type-aware extension of the QA normalizer. `None`, the empty string, and the strings *none*, *null*, *n/a*, *na*, *not available*, *not applicable*, and *unknown* all map to a single null sentinel. Booleans become *true* or *false*. Numeric values, and strings that parse as numbers after stripping `$`, `,`, and `%`, are emitted as numbers with trailing decimals removed. Everything else goes through the QA normalizer.

Cell comparison. Two cells match if their normalized forms are equal. A gold of *true* also accepts *yes* or *1*, and a gold of *false* accepts *no* or *0*. If the gold cell is null or purely numeric, only exact equality counts. Otherwise the same token containment rule as in QA applies.

Row alignment. Predicted rows are first deduplicated by their normalized key tuple, keeping the first occurrence. Each remaining predicted row is aligned to the first gold row, in gold order. Predicted rows that find no gold row remain unmatched.

Per-task precision and recall. When the table has non-key columns, scoring is over non-key cells. Recall is the number of correct non-key cells divided by the number of non-key cells in the gold table. Precision is the same numerator divided by the number of non-key cells in the deduplicated prediction. When every column is a key, each matched row is worth one credit instead.

C PROMPTS

The prompts provided to the respective agent depend on the combination of harness and task that is being evaluated.

C.1 CLI HARNESS

C.1.1 SYSTEM PROMPT

You answer questions by searching a corpus of .eml/.txt documents.

Use ‘bash‘ inside an isolated container. The current directory is writable scratch space at ‘/workspace‘; the evidence corpus is read-only at ‘/workspace/documents‘. You may use normal shell commands and scripts to inspect it. A failed shell command is recoverable: correct the command and continue.

Strategy:
1. Use ‘bash‘ to locate and inspect relevant documents.
2. Try multiple keywords and variations — entity names, predicate words, dates, and related terms.
3. Do not give up after a single search. If the first keyword returns nothing useful, try synonyms, partial names, and grep across all files.
4. Finish only by calling the provided final-submission tool exactly once; never provide the final answer only as prose.
5. For QA, submit ‘abstain=false‘ with a non-empty answer when supported. If reasonable searches do not support an answer, submit ‘abstain=true‘ with ‘answer=null‘. Do not use the literal text ”I don’t know” as a substitute for explicit abstention.
6. For structured prediction, submit the requested rows, or an empty array when no rows match.

C.1.2 QA USER PROMPT

You have shell access to {docs_dir}.

Question: {question}

Read the relevant documents to find the answer. Finish only by calling ‘submit_answer‘:
- If supported, call it with abstain=false and the answer value.
- If unsupported by the documents, call it with abstain=true and answer=null.

C.1.3 STRUCTURED PREDICTION USER PROMPT

You have read-only access to {docs_dir}.

Question: {question}
Expected columns: {headers}

Finish only by calling ‘submit_table‘ with a JSON array of objects. Each object must have exactly these keys: {headers}.
If the documents contain no matching rows, submit an empty array.

C.2 RAG HARNESS

C.2.1 SYSTEM PROMPT

You answer questions about a corpus of documents you cannot see directly.

Your only tool is ‘retrieve(query, k)‘, which returns the top-k most relevant text chunks from an embedding index. You have no filesystem access, no shell, and no way to read documents otherwise.

Strategy:
1. Issue several retrieve calls with different keywords — entity names, predicate words, dates, synonyms, related terms.

2. Do not stop after one retrieval. If the first result is not enough, refine or broaden the query.
3. Finish only by calling the provided final-submission tool exactly once; never provide the final answer only as prose.
4. For QA, submit ‘abstain=false‘ with a non-empty answer when supported. If the retrieved corpus does not support an answer, submit ‘abstain=true‘ with ‘answer=null‘. Do not use the literal text ”I don’t know” as a substitute for explicit abstention.
5. For structured prediction, submit the requested rows, or an empty array when no rows match.

C.2.2 QA User Prompt

Question: {question}
Use the ‘retrieve‘ tool to find the answer. Finish only by calling ‘submit_answer‘:
- If supported, call it with abstain=false and the answer value.
- If unsupported by the retrieved corpus, call it with abstain=true and answer=null.

C.3 Structured Prediction User Prompt

Question: {question}
Expected columns: {headers}
Use the ‘retrieve‘ tool to find the rows. Finish only by calling ‘submit_table‘ with the rows. Each object must have exactly these keys: headers. If no rows match, submit an empty array.

D Experimental Results

D.1 Oracle QA Results

We provide the full question answering results for the oracle setting in Table 5. Table 6 includes a side-by-side comparison between the standard and oracle settings.

Table 5: Oracle question answering results. Results only include answerable questions. **Bold** denotes best score among all models for each harness.

		Answerable		
Harness	**Model**	**Correct**	**False Refusal**	**Incorrect**
CLI	Opus 5	.421	.136	.443
	Sonnet 5	.555	.065	.380
	Haiku 4.5	.773	.030	.197
	GPT-5.6 Sol	.585	.006	.408
	GPT-5.6 Terra	.469	.006	.525
	GPT-5.6 Luna	.518	.004	.477
	DeepSeek-V4-F	.799	.019	.181
	GPT-OSS-120B	.816	**.000**	.184
	Qwen3.6-27B	**.849**	.011	**.140**
RAG	Opus 5	.421	.095	.484
	Sonnet 5	.641	.050	.309
	Haiku 4.5	.737	.030	.233
	GPT-5.6 Sol	.575	.006	.419
	GPT-5.6 Terra	.579	.002	.419
	GPT-5.6 Luna	.523	.006	.471
	DeepSeek-V4-F	.760	.024	.216
	GPT-OSS-120B	.773	.011	.216
	Qwen3.6-27B	**.793**	**.000**	**.207**

D.2 Relaxed Visibility QA Results

The relaxed visibility rule treats inferred facts as visible when at least 50% of their prerequisite facts are visible. Applying this rule reclassified 46 QA items from unanswerable to answerable across

Table 6: Standard versus oracle question answering results. Δ is oracle minus standard answerable correctness.

Harness	Model	Standard	Oracle	Δ
CLI	Opus 5	.432	.421	-.011
	Sonnet 5	.499	.555	+.056
	Haiku 4.5	.743	.773	+.030
	GPT-5.6 Sol	.540	.585	+.045
	GPT-5.6 Terra	.501	.469	-.032
	GPT-5.6 Luna	.495	.518	+.024
	DeepSeek-V4-F	.775	.799	+.024
	GPT-OSS-120B	.758	.816	+.058
	Qwen3.6-27B	.816	.849	+.032
RAG	Opus 5	.436	.421	-.015
	Sonnet 5	.568	.641	+.073
	Haiku 4.5	.698	.737	+.039
	GPT-5.6 Sol	.557	.575	+.017
	GPT-5.6 Terra	.570	.579	+.009
	GPT-5.6 Luna	.499	.523	+.024
	DeepSeek-V4-F	.691	.760	+.069
	GPT-OSS-120B	.734	.773	+.039
	Qwen3.6-27B	.672	.793	+.121

the four scenarios: Employee Lifecycle (8), Mergers & Acquisitions (21), Sales Lifecycle (14), and, VC Fundraising (3).

Table 7: QA sensitivity to relaxed visibility. A derived fact is classified as visible when at least 50% of its required facts are visible. Balanced accuracy is reported, macro-avering across answerable and unanswerable items.

Harness	Model	Strict	Relaxed	Δ	Rank
CLI	Opus 5	.602	.597	-.005	**1** $\rightarrow$ **2**
	Sonnet 5	.558	.567	+.009	4 $\rightarrow$ 4
	Haiku 4.5	.574	.579	+.006	3 $\rightarrow$ 3
	GPT-5.6 Sol	.384	.362	-.022	8 $\rightarrow$ 8
	GPT-5.6 Terra	.376	.348	-.027	9 $\rightarrow$ 9
	GPT-5.6 Luna	.405	.402	-.003	7 $\rightarrow$ 7
	DeepSeek-V4-F	.579	.599	+.020	**2** $\rightarrow$ **1**
	GPT-OSS-120B	.500	.485	-.015	6 $\rightarrow$ 6
	Qwen3.6-27B	.548	.563	+.016	5 $\rightarrow$ 5
RAG	Opus 5	.505	.512	+.007	4 $\rightarrow$ 4
	Sonnet 5	.597	.570	-.026	1 $\rightarrow$ 1
	Haiku 4.5	.555	.547	-.007	3 $\rightarrow$ 3
	GPT-5.6 Sol	.385	.369	-.016	**6** $\rightarrow$ **7**
	GPT-5.6 Terra	.377	.352	-.025	8 $\rightarrow$ 8
	GPT-5.6 Luna	.378	.375	-.003	**7** $\rightarrow$ **6**
	DeepSeek-V4-F	.574	.563	-.010	2 $\rightarrow$ 2
	GPT-OSS-120B	.448	.437	-.011	5 $\rightarrow$ 5
	Qwen3.6-27B	.354	.337	-.017	9 $\rightarrow$ 9

D.3 Inference-Depth-Aware QA Results

D.4 Structured Prediction

We separate Synthesis tasks according to whether their ground truth table is empty or not. For empty tables, correct [] denotes an empty submission, unsupported denotes a submission containing rows, and terminal denotes a terminal model failure. For nonempty tables, false [] denotes an empty submission. All rates are calculated within the corresponding subset.

Table 8: Answerable QA results by fact inference depth. D_0 denotes a directly surfaced fact and higher depths denote inference steps. D_0 contains 345, D_1 99, and, D_2 19 items. Accuracy is reported (Correct/Total).

		Standard			Oracle		
Harness	**Model**	D_0	D_1	D_2	D_0	D_1	D_2
CLI	Opus 5	.554	.091	.000	.542	.081	.000
	Sonnet 5	.626	.121	.158	.707	.121	.053
	Haiku 4.5	.817	.525	.526	.852	.545	.526
	GPT-5.6 Sol	.638	.242	.316	.696	.253	.316
	GPT-5.6 Terra	.594	.222	.263	.533	.242	.474
	GPT-5.6 Luna	.591	.182	.368	.612	.202	.474
	DeepSeek-V4-F	.800	.727	.579	.823	.768	.526
	GPT-OSS-120B	.809	.626	.526	.849	.737	.632
	Qwen3.6-27B	.846	.758	.579	.878	.788	.632
RAG	Opus 5	.557	.101	.000	.536	.101	.000
	Sonnet 5	.713	.152	.105	.774	.253	.263
	Haiku 4.5	.788	.434	.421	.829	.455	.526
	GPT-5.6 Sol	.646	.263	.474	.675	.242	.474
	GPT-5.6 Terra	.678	.222	.421	.690	.212	.474
	GPT-5.6 Luna	.586	.212	.421	.614	.222	.421
	DeepSeek-V4-F	.751	.556	.316	.806	.636	.579
	GPT-OSS-120B	.786	.596	.526	.814	.667	.579
	Qwen3.6-27B	.783	.384	.158	.849	.657	.474

Table 9: Synthesis results separated by ground-truth emptiness across all scenarios. The empty and nonempty subsets contain, 162 tasks and 135 tasks, respectively. Terminal denotes failure without an accepted table submission.

		Gold empty			Gold nonempty		
Harness	**Model**	**Correct []**	**Unsupported**	**Terminal**	**F_1**	**False []**	**Terminal**
CLI	Opus 5	.864	.136	.000	.576	.007	.000
	Sonnet 5	.877	.123	.000	.570	.044	.000
	Haiku 4.5	.895	.105	.000	.616	.052	.022
	GPT-5.6 Sol	.802	.198	.000	.693	.007	.000
	GPT-5.6 Terra	.852	.148	.000	.676	.007	.000
	GPT-5.6 Luna	.821	.160	.019	.659	.007	.015
	DeepSeek-V4-F	.833	.167	.000	.642	.000	.007
	GPT-OSS-120B	.716	.222	.062	.534	.007	.052
	Qwen3.6-27B	.889	.099	.012	.563	.015	.193
RAG	Opus 5	.877	.123	.000	.587	.007	.000
	Sonnet 5	.889	.111	.000	.608	.030	.000
	Haiku 4.5	.889	.111	.000	.570	.030	.000
	GPT-5.6 Sol	.833	.167	.000	.698	.000	.000
	GPT-5.6 Terra	.840	.160	.000	.687	.007	.000
	GPT-5.6 Luna	.802	.191	.006	.669	.007	.000
	DeepSeek-V4-F	.852	.148	.000	.632	.015	.000
	GPT-OSS-120B	.827	.154	.019	.539	.089	.007
	Qwen3.6-27B	.673	.049	.278	.394	.081	.370

D.5 Tool Calling Statistics

Table 10: Tool-calling statistics across all question answering and structured prediction items. Multiple calls denotes the percentage of tasks where more than one tool call was made. Final-submission calls are excluded.

Harness	Model	Mean Calls	Median Calls	Multiple Calls
CLI	Opus 5	9.6	9.0	100.0%
	Sonnet 5	6.5	4.0	100.0%
	Haiku 4.5	23.0	16.0	100.0%
	GPT-5.6 Sol	8.7	7.0	99.8%
	GPT-5.6 Terra	5.8	5.0	99.1%
	GPT-5.6 Luna	8.7	8.0	99.4%
	DeepSeek-V4-F	34.8	25.5	100.0%
	GPT-OSS-120B	17.2	14.0	99.9%
	Qwen3.6-27B	24.2	17.0	100.0%
RAG	Opus 5	8.1	6.0	100.0%
	Sonnet 5	5.0	3.0	63.1%
	Haiku 4.5	7.2	4.0	88.7%
	GPT-5.6 Sol	16.6	11.0	100.0%
	GPT-5.6 Terra	13.0	9.0	99.9%
	GPT-5.6 Luna	12.7	8.0	99.9%
	DeepSeek-V4-F	8.9	6.0	75.9%
	GPT-OSS-120B	8.2	7.0	90.0%
	Qwen3.6-27B	18.1	4.0	66.5%

Table 11: Terminal model-error breakdown across the standard and oracle-evidence evaluations. No submission denotes normal completion without an accepted submission and max turns indicates termination upon exhausting the turn budget.

Evaluation	Harness	Model	Task	Error	Count
Standard	CLI	Haiku 4.5	QA	Model Max Turns	7
			SP	Model Max Turns	3
		GPT-5.6 Sol	SP	Model Context Exhaustion	1
		GPT-5.6 Terra	QA	Model No Submission	38
		GPT-5.6 Luna	QA	Model No Submission	6
			SP	Model No Submission	5
		DeepSeek-V4-F	QA	Model Max Turns	8
			SP	Model Max Turns	7
		GPT-OSS-120B	QA	Model No Submission	11
			SP	Model Max Turns	1
			SP	Model No Submission	18
		Qwen3.6-27B	QA	Model Max Turns	47
			SP	Model Max Turns	28
	RAG	GPT-5.6 Luna	SP	Model No Submission	2
		GPT-OSS-120B	QA	Model Context Exhaustion	2
			QA	Model No Submission	9
			SP	Model Context Exhaustion	1
			SP	Model No Submission	5
		Qwen3.6-27B	QA	Model Context Exhaustion	33
			QA	Model Max Turns	103
			QA	Model No Submission	9
			SP	Model Context Exhaustion	80
			SP	Model Max Turns	35
			SP	Model No Submission	2
Oracle	CLI	GPT-5.6 Terra	QA	Model No Submission	92
		GPT-5.6 Luna	QA	Model No Submission	2
	RAG	Qwen3.6-27B	QA	Model Context Exhaustion	3
			QA	Model Max Turns	22
			QA	Model No Submission	1

E SAMPLE TASKS

Below, we outline a sample task for each task type across QA and structured prediction for each scenario.

E.1 QA TASKS

Mergers & Acquisitions

Question: Who has Forge Aerospace Group signed a non-disclosure agreement with?

Expected answer: Abstention (supporting document masked)

Employee Lifecycle

Question: Who ran orientation for Kimberly Allen?

Expected answer: Evelyn Blake

Sales Lifecycle

Question: Who is the champion at Crescent Entertainment for the Forge Northstar Broadcast deal?

Expected answer: Abstention (supporting document masked)

Venture Fundraising

Question: Who presented at the pitch meeting for Bluecrest Industries?

Expected answer: Brent Delacruz

E.2 Structured-Prediction Tasks

E.2.1 Synthesis

Mergers & Acquisitions

Question: List all bidders for Nova Energy Holdings. Columns: bidder, ioi_value.

Expected answer:

bidder	ioi_value
Granite Energy	5,324,000,000
Forge Atlas Works	4,532,000,000
Cobalt Keystone Manufacturing	3,936,000,000

Employee Lifecycle

Question: How many feedback entries of each sentiment did Kimberly Allen receive? Columns: sentiment, count.

Expected answer:

sentiment	count
outstanding	3
neutral	3
critical	3
negative	1
positive	1

Sales Lifecycle

Question: For the Beacon Precision Healthtech deal, summarize evaluation issues by track. Columns: track, issue_count, highest_severity.

Expected answer: empty table (no evaluation issues recorded for this deal)

Venture Fundraising

Question: List every business due diligence question raised across all deals, with startup, track, and topic. Columns: startup, track, topic.

Expected answer:

startup	track	topic
Nova Holdings Enterprises	financial_due_diligence	gross_margin_walk
Nova Holdings Enterprises	financial_due_diligence	arr_bridge_quality_of_revenue
Nova Holdings Enterprises	team_due_diligence	background_and_litigation_check
	… (52 rows total) …	

E.2.2 Process Reconstruction

Mergers & Acquisitions

Question: List all processes in the Summit Flight Systems Technologies deal along with the date of the first and last document in each process. Columns: process, start, end.

Expected answer:

process	start	end
outreach	2023-08-09T02:24	2023-08-09T17:30
non_disclosure_agreement_data_room	2023-08-13T20:01	2023-08-14T18:40
indication_of_interest	2023-08-25T03:48	2023-08-25T03:48
termination	2023-08-27T17:51	2023-08-27T17:51

Employee Lifecycle

Question: List all processes for Dr. Russell Thomas along with the date of the first and last document in each process. Columns: process, start, end.

Expected answer:

process	start	end
onboarding	2022-11-08T21:08	2022-11-15T12:52
ramp_up	2022-12-20T05:18	2023-02-01T06:00
performing	2023-02-14T16:58	2023-04-27T14:33
retention	2023-05-15T11:06	2023-05-20T08:38
performing	2023-07-03T02:21	2023-09-07T07:14
offboarding	2024-02-09T02:04	2024-02-15T18:00

Sales Lifecycle

Question: List all processes for the Silverline Labs account along with the date of the first and last document in each process. Columns: process, start, end.

Expected answer:

process	start	end
prospecting	2023-05-20T08:35	2023-06-03T20:51
discovery	2023-06-01T14:34	2023-06-12T21:31
evaluation	2023-07-04T22:06	2023-07-24T08:00
negotiation	2023-07-29T08:21	2023-08-21T01:00
termination	2023-08-21T01:00	2023-08-22T12:25

Venture Fundraising

Question: List all processes in the Nova Holdings Enterprises deal along with the date of the first and last document in each process. Columns: process, start, end.

Expected answer:

process	start	end
sourcing	2023-10-05T16:36	2023-10-06T15:17
screening	2023-10-11T00:32	2023-10-12T07:04
partner_meeting	2023-11-01T19:38	2023-11-06T16:00
business_due_diligence	2023-11-09T14:21	2024-01-21T16:00
termination	2024-01-21T16:00	2024-01-22T21:27

E.2.3 Event Reconstruction

Mergers & Acquisitions

Question: List each event template used with its count, grouped by target. Columns: target, template, count.

Expected answer:

target	template	count
Nova Energy Holdings	emails	3
Nova Energy Holdings	calls	4
Nova Energy Holdings	non_disclosure_agreement_emails	1
Nova Energy Holdings	bidder_ndas	2
Nova Energy Holdings	non_disclosure_agreement_execution	1
Nova Energy Holdings	bidder_indication_of_interest_emails	2
… (376 rows across 20 targets)		

Employee Lifecycle

Question: List each event template used with its count, grouped by employee. Columns: employee, template, count.

Expected answer:

employee	template	count
Kimberly Allen	welcome_email	1
Kimberly Allen	hire_completed	1
Kimberly Allen	orientation	1
Kimberly Allen	it_setup	2
Kimberly Allen	buddy_assignment	1
Kimberly Allen	training_sessions	2
… (367 rows across 8 employees)		

Sales Lifecycle

Question: List each event template used with its count, grouped by customer. Columns: customer, template, count.

Expected answer:

customer	template	count
Beacon Precision Healthtech	initial_outreach_email	1
Beacon Precision Healthtech	lead_qualified	1
Beacon Precision Healthtech	record	1
… (326 rows across 19 customers)		

Venture Fundraising

Question: List each event template used with its count, grouped by startup. Columns: startup, template, count.

Expected answer:

startup	template	count
Bluecrest Industries	intro_outreach_email	1
Bluecrest Industries	founder_response_email	1
Bluecrest Industries	intro_call	1
Bluecrest Industries	internal_pipeline_log_email	1
Bluecrest Industries	deck_request_email	1
Bluecrest Industries	deck_share	1
… (505 rows across 27 startups)		

F HUMAN VALIDATION

Five members of our organization participated voluntarily in the annotation study and were compensated through their regular salaries. Four held graduate degrees and one held an undergraduate degree. All annotators had experience in corporate settings and were familiar with the document types they had to read and annotate as part of the study. For this study, we use facts and statements interchangeably.

Each annotator reviewed the same set of 200 documents consisting of 45 documents randomly sampled from each scenario and five negative-control document per scenario. For regular documents the annotation interface presented the contents of the document alongside the assigned statements and three distractor statements. Control documents were presented with five distractor statements only. Annotators were asked to select every statement that was supported by the presented document or attested that none applied. The annotation interface is shown in Figure 3.

We evaluate each annotator's selection for each document-statement pair, collapsing duplicate candidate statements before computing results. In Table 12 we report the proportion of assigned fact statements selected (recall), the proportion of distractor statements rejected (specificity), and inter-annotator agreement using Fleiss' κ.

One author manually reviewed all 29 document-fact pairs selected by at most one annotator. Upon review, these cases predominantly reflected annotator misses or partial surfacing, where a document expressed the core fact but omitted one of its arguments. The omitted arguments were not targets of our benchmark tasks. The remaining cases, identifying information appeared in the filenames accessible to the evaluated agents, but omitted from the annotation interface. Among all audited cases, no benchmark task depended on an unsupported core fact.

Table 12: Human validation results by scenario.

Scenario	Responses	Units	Recall	Specificity	κ
Mergers & Acquisitions	250	203	.833	.987	.834
Employee Lifecycle	250	243	.810	.991	.762
Sales Lifecycle	250	252	.685	.990	.684
Venture Capital Fundraising	250	233	.814	.986	.797

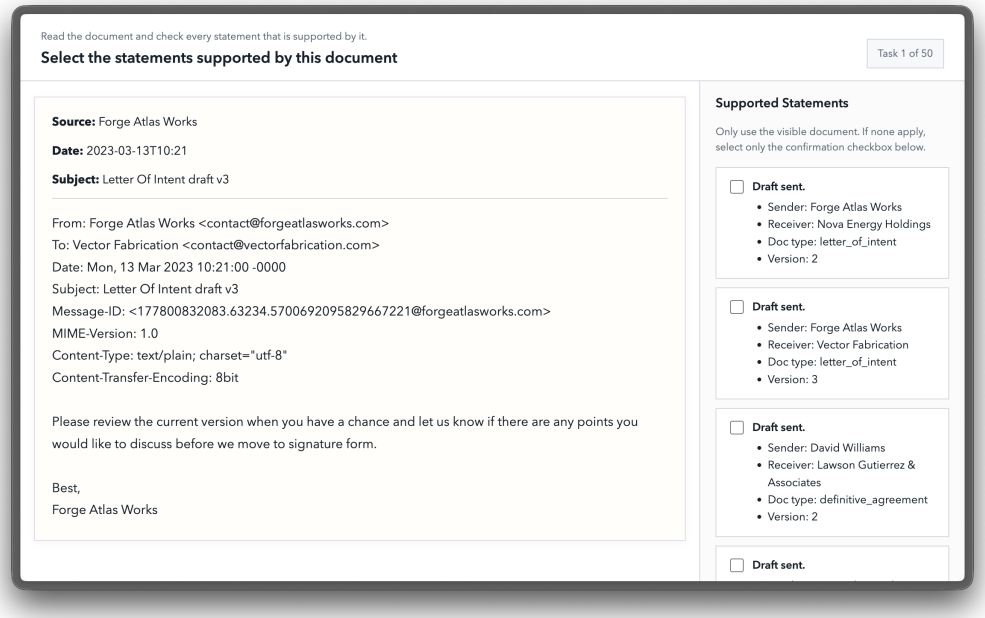

Figure 3: Human annotation interface for the M&A scenario.