CorporateBench: Large-Scale Q&A Benchmarking with Temporal Knowledge Bases

Sil Hamilton[1, 2]*, Albert Yu Sun[1], Oscar J. Romero[1], Carl-Leander Henneking[1], David Mimno[2], Bishan Yang[1], Igor Labutov[1]
[1] Epiq AI Labs [2] Cornell University

Abstract

LLMs are increasingly able to answer complex questions about enterprise-scale document collections. But evaluation is hard: companies don't want to share internal communications, and synthetic datasets have been overly simple. We present CORPORATEBENCH (CB), a human-validated multi-task Q&A benchmark whose scale approaches the conditions LLMs encounter in corporate communication networks, with evaluation corpora surpassing 230,000 documents. CB evaluates LLMs across two dimensions (information extraction and knowledge base querying) through four synthetically generated firms ranging from 12 to 10,000 employees. Each corpus is sampled from a temporally evolving knowledge base describing a consistent world, guaranteeing cross-document logical consistency even across hundreds of thousands of documents. We evaluate five LLMs on CB, revealing increasingly poor performance as input size approaches realistic scales. CB provides LLM developers a metric for corporate communication reasoning, filling a crucial gap in the benchmarking ecosystem.

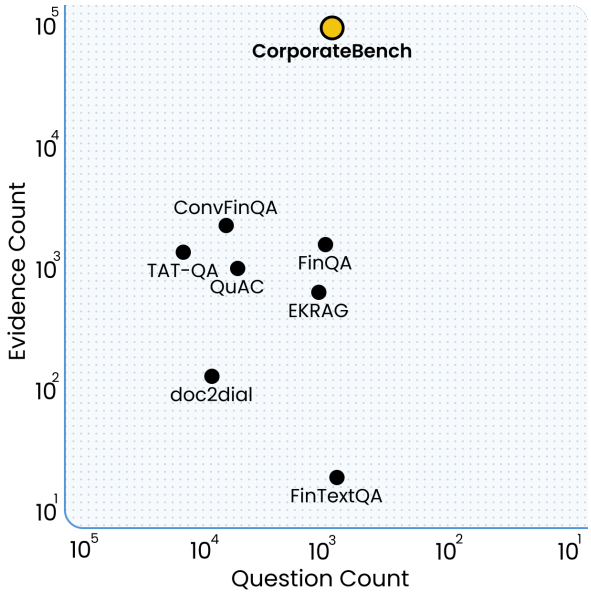

Figure 1: CORPORATEBENCH advances the state of the art in multi-document QA benchmarking when compared against nine previous benchmarks in evidence and question counts. CB achieves a high ratio of **87.6** documents per question, with the closest benchmark (EKRAG) achieving a ratio of 1.5 (Yu et al., 2025). A high ratio indicates that questions require synthesizing information across many documents, which better reflects the complexity of real enterprise knowledge work.

1 Introduction

Complex organizations like corporations produce massive quantities of internal documents through employees communicating and working together. LLMs are increasingly helpful for knowledge work, document analysis, and other tasks requiring general reasoning about organizations and their social networks (Raza et al., 2025; Brachman et al., 2025). But despite significant advances in extending large language model (LLM) context windows (Huang et al., 2024; Beltagy et al., 2020), state-of-the-art LLMs continue to struggle with complex reasoning over many documents as regularly required in enterprise scenarios (Liu et al., 2024; Hsieh et al., 2024). This deficiency hinders wide-scale LLM adoption in industry, but current evaluation metrics fail to capture this failure mode.

Common long context understanding tests do not reproduce the complexity and interconnectedness of real corporate environments. Synthetic benchmarks (Kamradt, 2023) are excessively simplified, while more realistic benchmarks based on real-world enterprise data are scarce due to non-disclosure agreements (Zhang et al., 2024). The few existing enterprise benchmarks moreover fail to adequately reproduce corporate conditions due to limitations in scale and realism. Enterprise corpora can scale to millions of documents, making the cost of generating a whole corpus from scratch a major concern when designing benchmarks.

*Correspondence to: srh255@cornell.edu

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 667–701
October 24-29, 2026 ©2026 Association for Computational Linguistics

To address these limitations, we introduce CORPORATEBENCH, an enterprise benchmark containing large and internally consistent corporate communication networks with verifiable ground truth as shown in Figure 2.[1] CORPORATEBENCH bridges the synthetic-real gap by first sampling structures, and then documents, from procedurally generated knowledge bases (KBs) that capture complex relationships between employees, teams, projects, and tasks over time. These KBs serve as our foundation for generating diverse corporate corpora at realistic scales. Our primary contributions are as follows:

- **Comprehensive company simulation.** We adapt KB-to-corpus generation to produce corporate document collections closely mimicking the real world, preserving temporal sequencing and role-respecting communication directions across teams and departments.
- **Deeply interconnected evidence sets.** We construct a large-scale benchmark in which answering typical queries requires integrating evidence across many documents, stressing multi-document retrieval and reasoning over enterprise-scale communication sets whose total size can exceed context windows.
- **Enterprise-scale tasks.** We provide a task suite with deterministic labels directly computed from the KB, enabling reproducible evaluation. This KB-grounded design targets the gap between models' extended context windows and genuine long context understanding in enterprise settings.

2 Related Work

Benchmarking LLMs in long contexts. Enterprise datasets are naturally long context. Contemporary LLMs can now attend to millions of tokens (OpenAI et al., 2024; Grattafiori et al., 2024; Team Gemini et al., 2024), but these models suffer from poor long context understanding (Kaplan et al., 2020; Liu et al., 2024). One method for identifying these failure modes is the "needle-in-a-haystack" test and its descendants (Kamradt, 2023; Vodrahalli et al., 2024; OpenAI, 2025), but these synthetic benchmarks trade realism for generation speed, leading to newer benchmarks testing models over documents like books and movie scripts (Kočiský et al., 2018; An et al., 2023; Karpinska et al., 2024; Hamilton et al., 2025) whose limited number fail to qualify as enterprise scale. Researchers seeking solutions have introduced enterprise benchmarks based on financial records (Deußer et al., 2022; Xie et al., 2024; Xu et al., 2024), legal (Manor and Li, 2019; Guha et al., 2023; Fei et al., 2023; Ryan et al., 2025), knowledge work (Boisvert et al., 2024; Drouin et al., 2024; Styles et al., 2024; Yao et al., 2024; Xu et al., 2025), and other general corporate tasks (Jiang et al., 2024; Zhang et al., 2024; Wang et al., 2025). These benchmarks suffer from low quality data sources, typically transforming non-corporate data to avoid violating non-disclosure agreements. An alternative approach involves generating diverse synthetic enterprise corpora with LLMs emulating knowledge work (Choubey et al., 2025; Huang et al., 2025; Vishwakarma et al., 2025), but existing implementations suffer from small scale (ca. $\leq$30,000 documents) and problematic ground truth because datasets are created via prompting LLMs and there is no single method for guaranteeing LLM output will match all desired characteristics. Matching real corporate scale therefore demands new generation methods.

Synthetic knowledge bases. Because enterprise environments are structurally self-similar (e.g. employees exist at all levels of an organization), they can be conveniently described in a knowledge base. KBs have long been used to represent human knowledge (Bollacker et al., 2008; Talmor and Berant, 2018), and more recently for producing grounded synthetic documents (Agarwal et al., 2021), but complex systems like enterprise environments were rarely profiled in early studies due to difficulties with parsing relations from unstructured text (Kwiatkowski et al., 2013; Yang et al., 2015; Yang and Mitchell, 2019). LLMs have since proven adept at this task (Labutov et al., 2019b; Sun et al., 2023; Machado et al., 2024; Sahay et al., 2025; Gong et al., 2025) and likewise converting text to structured queries (Labutov et al., 2019a; Trivedi et al., 2022; Wang et al., 2023; Sun et al., 2025; Chen et al., 2025). We look to KBs to provide a single, arbitrarily detailed world state from which we can (i) synthesize document collections at realistic, million-scale sizes and (ii) derive exact ground truth by querying the underlying graph.

3 Dataset Construction

We begin constructing our synthetic enterprise benchmark by establishing ground truth. In this section we describe a corpus construction pipeline

[1] We release data and code at https://huggingface.co/datasets/epiq-ai-labs/corporatebench.

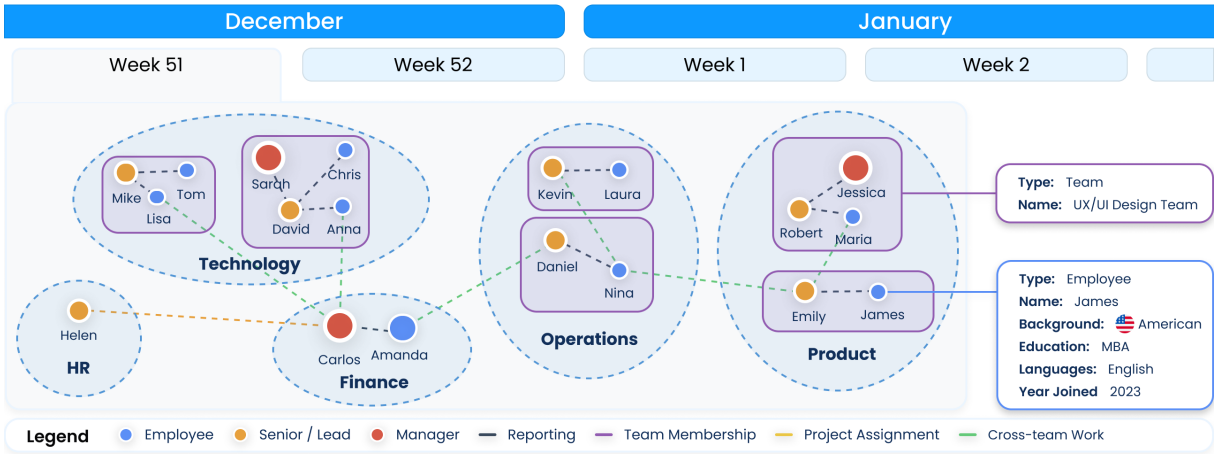

Figure 2: CORPORATEBENCH contains four synthetic companies spanning different industries, headcounts, and business models. The figure shows a vertical slice of a hypothetical company and its social network at a given time.

whose output enjoys two properties:

- **Arbitrary scale.** For any desired corpus size n, the pipeline can generate $\geq n$ documents while maintaining consistent quality across all generated documents, maintaining future relevance as LLM context windows grow.
- **Logical consistency.** $\forall d \in C$, no statement in document d contradicts any fact established in our ground truth knowledge base.

We first generate a knowledge base representing the world of a company C, encoding all entities together with their properties and relationships. We then generate emails that substantiate organizational relationships, tasks, and project progress from the KB, enforcing consistency by inserting "evidence" strings revealing relationships. This pipeline satisfies logical consistency because under ideal circumstances the graph can be reconstructed from all documents sampled from it, and arbitrary scale because documents are templated.

3.1 Defining the Knowledge Base

We begin with a formal ontology capturing the desired semantics of our companies, written in Terse RDF Triple Language (Turtle; Beckett et al. 2014). We formally define a company $C = (D, T, E)$ as the union of all employees E organized by teams T and departments D, where each team strictly belongs to one department and all employees strictly belong to one team, barring department leads and executives. Serializing our companies with an OWL parser guarantees consistency.

Our ontology defines two base classes, `Entity` and `Relationship`, serving as nodes and edges in the knowledge base. We extend these with organizational entities (companies, departments, teams, employees) and work matter (projects, meetings, tasks), linked by seven relationship predicates: `MemberOf`, `ReportsTo`, `WorksOn`, `WorksAt`, `Attends`, `Organize`, and `BelongsTo`. These tie together employees along three axes: organizational (membership and reporting hierarchies), work-wise (task assignments), and communication (meeting attendance and organization).

3.2 Generating Companies

To generate company knowledge bases, we run the following procedure conditioned on the type of company and desired number of employees:

1. Produce company hierarchy. We logarithmically scale department and team counts with the desired employee count, targeting an average team headcount of 5–10 and department headcount of 8–68 when scaling from 10^1 to 10^4 employees.[2] C-suite scales from 1 to 13 executives; departments and teams are equipped with one or more managers. We link all entities with `ReportsTo` and `MemberOf` relationships and verify the hierarchy forms a valid tree with no orphan nodes.

2. Simulate work over time. We simulate the company over the course of one quarter, or 90 days. For each day, we randomly assign each employee a task with $P = 1/45$ such that each employee completes between 1 to 3 tasks by the end of the quarter.[3] We track task assignment and comple-

[2] We select these values from real-world distributions.

[3] A more realistic workload would have employees complete 1 to 3 tasks a day. We purposefully limit this value to ensure uniform relationship counts (cf. Table 1).

tion dates. We verify that all task assignments fall within the simulated quarter and that no employee exceeds the target of 1 to 3 tasks.

3. Simulate meetings over time. We simulate meetings for employees across all six meeting types: direct reports, team meetings, task collaboration sessions, executive meetings, project reviews, and department meetings. The meeting generation process considers employee roles, reporting structures, team memberships, and project assignments to determine appropriate meeting schedules. Appendix D includes more details on meetings. We confirm all generated meetings respect role constraints, e.g. that direct report meetings only pair employees with their managers.

4. Update company hierarchy. At the end of the quarter we update the company hierarchy by hiring an additional ≈10% employees into a random assortment of teams. This process necessarily forces the splitting of those teams to maintain the desired average headcount across all teams, producing additional `MemberOf` relationships. This process ensures that not all `MemberOf` relationships are valid at the end of the quarter, thus requiring at least two documents to fully "reveal" employment status when sampling evidence. We verify that new hires are correctly linked with fresh `MemberOf` and `ReportsTo` relationships and that prior relationships are marked with appropriate end dates.

5. Annotate all entities. The company hierarchy at this point contains many entities, but no semantic properties such as names or titles. We prepare a one-paragraph "story" for each company in Figure 2 containing details such as company origin, motto, focus, industry, etc.[4] We then recursively pass each entity beginning at the top of the hierarchy down to the bottom to Claude Haiku 4.5 prompting it to produce a set of appropriate properties for each entity given its type and size. We finally produce aliases for employees and projects for diversity. We manually inspect a sample of annotated entities to confirm properties are contextually appropriate.

6. Produce triples. We finally render all entities, properties, and relationships in a sequence of triples as described in subsection 3.2. We store the graph in a N-Quads file (Tomaszuk and Hyland-Wood, 2020), compressing it with the ontology. We validate the serialized graph against the OWL ontology to ensure no constraint violations.

[4]We provide all stories in Appendix A.

```
From: Emilia A. <eanglada@hotmail.com>
To: Jared C. <jaredchartier@gmail.com>
Date: 2024-01-10
Subject: Addressing code quality concerns
in our operations

Hi Jared, I've been reflecting on how our
current business operations could benefit
from a more intentional approach to
managing technical
debt. As our codebase continues to grow,
I'm noticing that we're accumulating
layers of complexity that might slow down
our team's ability to iterate
efficiently. I think it would be valuable
for us to establish a clearer strategy
around this.

Building on that concern, Jared, I wanted
to get your direction here on how we
should prioritize which areas to tackle
first. I'm thinking we could start by
identifying the components that have the
highest maintenance burden and then work
through a systematic refactoring plan.
Would you be open to discussing this
further so we can align on the best path
forward for our technical infrastructure?

Best,
Emilia Anglada
```

Figure 3: An email generated for CORPORATEBENCH. Employees, topic, and evidence strings are in bold.

3.3 Generating Documents

We generate emails that surface a set of one or more relationships $r \in R$ in our KB, where an email contains a sender, recipient, subject, body, and timestamp as retrieved from the KB. We guarantee we substantiate relationship triples by adding **evidence string(s)** that explicitly reveal the relationship r from a predefined set of 3,217 strings across all seven `Relationship` types, split into 360 strings per temporal bound and 40 strings per level of difficulty.[5] Each string differs in how explicit the string is in revealing the relationship, and the time frame it reveals (either beginning or end of a relationship). Meetings use a separate set of evidence strings, illustrating task and project progress by the meeting date, with strings binned by progress up to 25%, 50%, 75%, and 100%. We provide examples

[5]For instance, "Just going through the object onboarding materials today" is an evidence string that substantiates that an employee's WorksOn relationship has begun.

in Appendix B. We finish the email by passing the evidence string(s) to Claude Haiku 4.5 to in-fill the remainder of the email. An example is given in Figure 3. We use the following strategies to enhance lexical diversity in the output:

Format. Our email formatting conventions (headers, signatures, reply structure) are modeled on the Enron corpus (Klimt and Yang, 2004), a publicly available corporate communication dataset, to ensure structural realism.

Topic. To ensure diversity of synthetically generated data at scale, we sample a random topic $\mathcal{T}$ and use it to generate the majority of each email. Topics are sampled from a Dirichlet distribution with $\alpha = 0.75$ to mimic real-world distributions where topic frequency is non-uniform. There are two sets: 3,000 business topics (e.g. clinical trials), and 3,000 non-business topics (e.g. food, travel, and transportation). Meeting topics are sampled from a separate set of 60 topics.[6]

Personality and level of formality. We assign each employee one of sixteen personalities sourced from the Myers–Briggs Type Indicator typology (Briggs, 1976). At the document level, we instruct the model to write emails in either a formal or a casual manner. Qualitative observations indicate these contribute positively to lexical diversity.

4 Dataset Validation

We use the pipeline described in section 3 to produce four company knowledge bases with 10^1 to 10^4 employees. We present these companies in Table 1. We validate the dataset in four ways:

Network properties. We observe consistent scaling patterns across each company size. For example, the smallest company (Zenith Labs) contains 12 employees, while the largest company (Pound) contains 10,210 individuals. Zenith Labs contains 455 total relationships at an average ≈38 degrees per employee; while Pound contains ≈37 degrees. This stable scaling behaviour helps downstream tasks control for issues arising from poor scaling.

Examining sampled documents. We generate a total 263,466 documents across all four company KBs to produce four corpora containing, respectively: 354 documents, 3,926 documents, 26,493 documents, and 232,693 documents. Each document explicitly reveals one or more relationships in each company KB following the sampling methodology described in subsection 3.3.[7]

Validating document topics. We provide topic distributions for each company corpus in Table 8. Total topic counts scale linearly from 6 in Zenith (S), 60 for Streamvibe (M), 600 for Biocure (L), and finally 6,000 topics in Pound (XL).

Human validation. We sample 100 random emails from the full CORPORATEBENCH corpus for human validation. Because each document positively asserts a relationship (e.g. that one person manages another), we prompt Claude Sonnet 4 to invert each document by modifying the evidence string to negate the relationship. This yields a balanced test set of positive and negative examples. We pair ten groups of five non-expert reviewers with batches of 20 documents each, asking them to judge whether each document positively asserts the relationship on a binary scale. Across all 1,000 judgements, reviewers correctly identify the intended relationship 76.2% of the time.

5 Tasks

CORPORATEBENCH tests models on five tasks organized in two categories: extraction and QA. The two assess distinct capabilities. Extraction assesses whether a model can *construct* a structured knowledge base from raw documents while QA evaluates *reasoning* over an already-constructed representation. A model may answer KB QA questions correctly yet fail to build the KB in the first place.

5.1 Extraction Tasks

Our two extraction tasks evaluate models on their ability to recover (i) relations and (ii) topics from each company corpus. The tasks are:

KB Evaluation. This task involves three stages: ingestion, mapping, and evaluation. During ingestion, LLMs extract entities and relationships from documents using the prompt template in Appendix P. We then apply deduplication techniques like exact name match and email-based clustering to reduce redundancy. This process yields triples in the form ⟨subject, predicate, object⟩. We evaluate these ingested triples against ground truth triples from CORPORATEBENCH and compute F_1 for all matched entities/relationships.

[6] We provide more details in Appendix C.

[7] We provide representative sample documents for each QA type in Appendices G–I, illustrating document diversity, question complexity, and ground-truth answer precision.

Company	Zenith	Streamvibe	Biocure	Pound
Size	Small (S)	Medium (M)	Large (L)	Extra Large (XL)
Industry	Tech	Media	Pharmacare	Finance
Entity Types				
Task	103	1,134	6,479	52,504
Meeting	76	890	6,445	57,265
Employee	12	122	1,110	10,210
Project	7	77	317	587
Team	1	11	90	207
Department	1	7	11	15
Company	1	1	1	1
Relationship Types				
WorksOn	124	1,356	7,708	63,431
ReportsTo	11	133	1,255	11,455
MemberOf	11	126	1,576	15,954
WorksAt	12	122	1,110	10,210
Attends	221	2,709	22,772	213,590
Organize	76	890	6,445	57,265
Total Entities	201	2,242	14,453	120,789
Total Relationships	455	5,336	40,866	371,905
Total Documents	354	3,926	26,493	232,693

Table 1: Summary statistics for each of the four synthetic companies generated for CORPORATEBENCH. Note that each company represents an additional order of magnitude over the previous in all respects.

KB QA	How many people began working at Pound Financial Group LLC after 7th March 2024?
Topic QA	Did Marion Friend discuss both transportation and value creation in communications after 30 Jan 2024?
Integrated QA	Who among client financial documentation preparation workers sent external emails about due diligence?

Table 2: Example questions for Pound (XL) across KB, Topic, and Integrated QA. More extensive question breakdowns are available in Appendix F.

Topic Classification. This task tests the model's ability to classify email topics via multi-class classification on a subset of topics from the dataset. The model must select one of either 31 classes (for the large company *Biocure*) or 17 classes (for the extra large company *Pound*) on a stratified test set of 1000 documents per corpus as provided in Appendix R. We subset these topics from the original full set of topics by navigating up the topic hierarchy as described in Appendix C to arrive at a lower count tenable for classification with current LLMs.[8] We finally evaluate performance with a macro F_1 averaged across the classes.

5.2 QA Tasks

Our three QA tasks evaluate models on factual questions about the company and its documents. QA questions are generated from 250 manually written question templates instantiated across companies and tasks with placeholder variables (e.g., employee or task) that are filled dynamically with values from the KB via verified SPARQL queries (and thus not by LLM generation). By construction, more complex questions require gathering information across more nodes in the knowledge graph and thus across more documents. QA questions come in five answer types: string, date, integer, boolean, and sets of strings. All types, except for sets, are judged on exact match (1 if match, 0 if not), but we calculate F_1 for sets to allow models partial credit. We compare models in two QA settings. The first is **RAG** (Lewis et al., 2021). We equip models with a RAG-based search tool containing all the documents.[9] This setting establishes performance in realistic corporate settings. The next setting is **KB**, where we equip models with a SQL tool for querying data directly imported with our ETL pipeline from our ground truth KB file.[10] This setting establishes an upper bound on model performance *relative to the RAG setting*, since models receive direct access to structured ground truth rather than retrieving from noisy documents. We note that text-to-SQL remains a challenging task in its own right, and that these scores do not represent an absolute performance ceiling. The tasks are as follows:

KB QA. This task involves answering questions about entities, relationships, and the temporality of these relationships. The example question in Table 2 requires identifying employees from emails, *and* temporal reasoning to identify when employees joined based on onboarding documents.

[8] We originally defined 31 topics for Pound as well, reducing it to 17 topics for evaluation due to topic overlap.

[9] Using `pgvector` and `text-embedding-3-small`.

[10] We describe the pipeline in Appendix E.

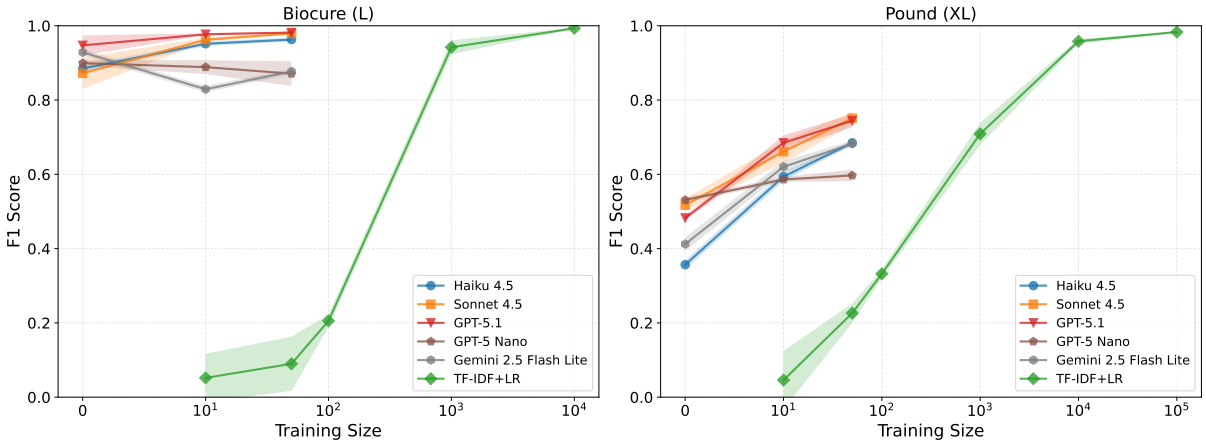

Figure 4: Topic classification F_1 scores across number of few-shot examples provided in the prompt (for LLMs) or labeled training examples (for TF-IDF+LR) for Biocure (L) and Pound (XL) datasets.

		Entities			Relationships			Temporal Relationships†		
Size	**Model**	**Precision**	**Recall**	**F_1**	**Precision**	**Recall**	**F_1**	**Precision**	**Recall**	**F_1**
S	Haiku 4.5	.866	.786	**.824**	.751	.966	**.845**	.453	.488	**.470**
	GPT-5 Nano	.926	.624	.746	.795	.285	.419	.317	.092	.142
	Gemini 2.5 F-L	.846	.792	.818	.677	.969	.797	.441	.444	.442
M	Haiku 4.5	.911	.721	**.805**	.782	.595	**.676**	.427	.209	.281
	GPT-5 Nano	.902	.645	.752	.569	.165	.256	.273	.046	.078
	Gemini 2.5 F-L	.849	.749	.796	.613	.708	.657	.391	.279	**.326**
L	Haiku 4.5	.902	.704	**.790**	.646	.556	**.597**	.390	.207	**.271**
	GPT-5 Nano	.928	.673	.780	.479	.167	.247	.254	.055	.090
	Gemini 2.5 F-L	.825	.707	.762	.417	.539	.470	.010	.012	.011
XL	Haiku 4.5	.937	.648	.766	.616	.412	**.494**	.310	.120	**.173**
	GPT-5 Nano	.962	.652	**.777**	.456	.133	.206	.191	.037	.062
	Gemini 2.5 F-L	.899	.594	.715	.227	.186	.205	.289	.082	.128

Table 3: KB Evaluation Results. **Bold** denotes best score out of the three models.

Topic QA. This task involves answering questions about the topics of the documents that are covered in emails. The Topic QA example in Table 2 requires models to read the employee's emails during a specific interval of time and answer whether the employee discussed that topic.

Integrated QA. The final QA task combines KB QA and Topic QA. The example question in Table 2 requires models to determine which employees have worked on a task and *then* analyze the content of specific emails they sent.

6 Results

We evaluate five contemporary models on CORPORATEBENCH to provide a sense of the current state of the art. These models are: Claude Haiku 4.5, Claude Sonnet 4.5, GPT-5 Nano, GPT-5.1, and Gemini 2.5 Flash Lite. We specifically select "lighter" models to strike a balance between performance and cost given evaluating a model on the full 263,466 document set in CB can incur significant API costs. We provide a full run-down of our evaluation metrics in Appendix K.

6.1 Extraction Tasks

For KB evaluation, Table 3 shows entity extraction substantially outperforms relationship extraction across all models and dataset sizes. Entity extraction remains stable across scales (F_1: 0.715–0.824), while relationship extraction degrades notably from smaller (S: 0.419–0.845, M: 0.256–0.676) to larger datasets (L: 0.247–0.597, XL: 0.205–0.494). Temporal relationships prove most challenging, deteriorating from F_1 0.142–0.470 (S) to 0.062–0.173 (XL). These findings indicate models struggle to maintain relational consistency at scale, even as factual consistency (entity extraction) is preserved. Error analysis revealed two primary causes: (1) entity extraction errors cascade into relationship extraction, and (2) contradictory relationship descriptions accumulate across documents, with each model

Method	Model	KB QA				Topic QA				Integrated QA			
		S	M	L	XL	S	M	L	XL	S	M	L	XL
RAG	Haiku 4.5	.265	.273	.189	.164	.650	**.563**	.418	.464	.332	.277	.327	.424
	Sonnet 4.5	.322	.311	.198	.215	.656	.535	**.423**	**.492**	.439	.250	.272	.401
	GPT-5 Nano	**.527**	**.501**	**.312**	.244	**.672**	.479	.385	.456	**.511**	.475	.468	.510
	GPT-5.1	.470	.440	.276	**.269**	.634	.485	.406	.456	.469	**.564**	**.560**	**.566**
	Gemini 2.5 F-L	.230	.200	.199	.147	.572	.401	.374	.376	.338	.218	.313	.295
KB	Haiku 4.5	.702	.724	.614	.597	.811	.619	.517	.488	.488	.540	.538	.540
	Sonnet 4.5	**.767**	**.742**	**.628**	**.642**	**.848**	**.687**	**.576**	**.571**	**.611**	.624	.644	**.647**
	GPT-5 Nano	.577	.484	.519	.522	.528	.431	.384	.409	.434	.611	.588	.567
	GPT-5.1	.555	.547	.481	.525	.468	.403	.387	.397	.431	**.640**	**.653**	.607
	Gemini 2.5 F-L	.258	.207	.248	.192	.384	.384	.304	.352	.272	.483	.528	.412

Table 4: KB QA, Topic QA, and Integrated QA Results. **Bold** denotes best model performance for the particular QA task using the respective methods (RAG or KB).

struggling with different relationship types.[11]

For topic classification, we compare against a TF-IDF+LR baseline. Figure 4 and Table 13 reveal distinct patterns across datasets. On Biocure (L), LLMs achieve strong few-shot performance (F_1: 0.872–0.950 zero-shot, 0.877–0.982 with 50 examples), matching TF-IDF+LR's 10K-example performance (F_1: 0.993) with 200× less data. On Pound (XL), LLMs show steeper learning curves (F_1: 0.357–0.531 zero-shot to 0.598–0.751 with 50 examples) but underperform the data-rich baseline (F_1: 0.983 at 100K examples). LLMs performed better on Biocure (31 categories) than Pound (17 categories) despite more classes, likely because Pound's hierarchical topic merging created ambiguous boundaries and semantic overlap.

6.2 QA Tasks

We report all QA performance in Table 4. Models consistently perform better with direct KB access than with RAG. Our design holds the number of questions constant at 250 per company per task (Appendix K) while increasing corpus size, so larger companies present proportionally more documents per question (≈1.4 to ≈931). Across all three QA types, the best KB scores exceed the corresponding best RAG scores, and this gap widens with scale: on KB QA, the KB–RAG difference increases from 0.24 (S) to 0.37 (XL). This suggests RAG may suffice for smaller corpora, but scale causes problems when surfacing knowledge.

Performance degrades with corpus size for KB QA and Topic QA, reflecting the difficulty of locating precise information in larger collections. Integrated QA exhibits a different pattern: performance slightly improves with scale for both methods, likely because larger corpora provide richer cross-source context for synthesis.

We finally note model-level differences: both GPT-5 models outperform others with RAG, while Claude Sonnet 4.5 excels at KB querying. However, GPT-5.1 exhibits an early stopping problem that disproportionately affects the KB setting (191 occurrences vs. 56 with RAG), partially explaining the performance gap between methods.[12]

7 Conclusion

We release CORPORATEBENCH, a benchmark for testing LLMs on enterprise-specific retrieval and reasoning tasks. CB features 263,466 total synthetic documents sampled from four procedurally generated knowledge bases representing companies of varying scale. This novel generation process allows CB to produce realistic tasks over corpora of arbitrary scale while maintaining inter-document logical consistency, a quality predecessor enterprise benchmarks lacked. Testing five recent LLMs from Anthropic, OpenAI, and Google indicates contemporary LLMs are marginally competent at completing enterprise tasks, with performance decreasing inversely with corpora scale up to and beyond 10^5 documents. Our methods, benchmark, and results are of value to researchers and industry practitioners interested in assessing LLM performance when deployed in corporate environments.

Next steps. We encourage future researchers to continue closing the gap between synthetic and realistic benchmarks. Researchers working in other domains (e.g. deep research) will want to evaluate whether replicating our use of procedurally generated KBs will be similarly fruitful.

[11] See Appendix M for more information.

[12] Specifically, the model returns empty string after receiving a tool response instead of generating an answer.

Limitations

We acknowledge the following limitations:

Scope limited to communication networks. CORPORATEBENCH focuses on email, which remains the dominant document type in corporate communication networks: the Enron corpus, the largest publicly available corporate dataset, is composed almost entirely of email (Klimt and Yang, 2004). Real corporate settings additionally involve communication channels such as Slack and Microsoft Teams, and document types such as technical specifications, financial reports, and presentations, each with distinct structural characteristics. Extending CORPORATEBENCH to additional communication channels is a direction for future work.

Simplified document types. Our benchmark focuses primarily on email communications. Real corporate settings involve newer communication channels such as Slack and Microsoft Teams, and documents themselves may be diverse across document types such as technical specifications, financial reports, spreadsheets, presentations. Each document type has distinct structural characteristics that may present different challenges for LLMs beyond just emails. We note that emails still remain a large majority document type in recent corporate discovery contexts (Gallivan, 2024).

Temporal dynamics and evolution. In simulating one quarter of company activity, we explicitly do not model longer-term organizational evolution such as strategic pivots, mergers and acquisitions, cultural shifts, or the gradual accumulation of institutional knowledge over years. We leave it for future work to expand beyond the 90-day simulation window to surface longer term phenomena.

LLM-generated artifacts. Despite efforts to ensure diversity through topics, personalities, and formality levels, the documents are generated by Claude Haiku 4.5, which may introduce systematic biases or artifacts in writing style, vocabulary, and document structure. Models evaluated on this benchmark may perform differently on human-written corporate documents. We see our tasks as a "lower bound" in terms of difficulty, where LLM agents need to be able to solve our tasks first before solving likely messier, more difficult, and more complex human-written corpora.

QA datasets. There are a few assumptions and/or design decisions that affect the QA datasets. These are offered to the reader in Appendix J.

Ethical Considerations

All synthetically generated materials in CORPORATEBENCH were generated via OpenAI and Anthropic-provided API endpoints with content filtering enabled. We note LLM developers often use benchmarks as a reference point during training, and that this practice can lead to benchmark-specific idiosyncrasies and can disproportionately emphasize certain semantic and/or syntactic qualities in model output. We therefore promoted ethnic and racial diversity in our generated companies by pre-computing names with Faker, a Python library for procedurally generating personal information like names and emails. We enable all Latin alphabet-based Faker lists, ensuring a fair and diverse sample of humans across all four companies.

References

Oshin Agarwal, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2021. Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3554–3565.

Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. 2023. L-Eval: Instituting Standardized Evaluation for Long Context Language Models. *Preprint*, arXiv:2307.11088.

David Beckett, Tim Berners-Lee, Eric Prud'hommeaux, and Gavin Carothers. 2014. Rdf 1.1 turtle. *World Wide Web Consortium*, 25:18–31.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Léo Boisvert, Megh Thakkar, Maxime Gasse, and Massimo Caccia. 2024. WorkArena++: Towards Compositional Planning and Reasoning-based Common Knowledge Work Tasks. In *38th Conference on Neural Information Processing Systems (NeurIPS 2024) Track on Datasets and Benchmarks.*

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 1247–1250, New York, NY, USA. Association for Computing Machinery.

Michelle Brachman, Amina El-Ashry, Casey Dugan, and Werner Geyer. 2025. Current and future use of

large language models for knowledge work. *Preprint*, arXiv:2503.16774.

Katharine C Briggs. 1976. *Myers-Briggs type indicator*. Consulting Psychologists Press Palo Alto, CA.

Yongrui Chen, Zhiqiang Liu, Jing Yu, Lin Ren, Nan Hu, Xinbang Dai, Jiajun Liu, Jiazhen Kang, Shenyu Zhang, Xinda Wang, Keyan Ding, Pengfei Shen, Haolei Zhu, Hongjie Deng, Yisong Wang, Tongtong Wu, Sheng Bi, Wen Zhang, Tianxing Wu, and 5 others. 2025. OneEval: Benchmarking LLM Knowledge-intensive Reasoning over Diverse Knowledge Bases. *Preprint*, arXiv:2506.12577.

Prafulla Kumar Choubey, Xiangyu Peng, Shilpa Bhagavath, Kung-Hsiang Huang, Caiming Xiong, and Chien-Sheng Wu. 2025. Benchmarking Deep Search over Heterogeneous Enterprise Data. *Preprint*, arXiv:2506.23139.

Tobias Deußer, Syed Musharraf Ali, Lars Hillebrand, Desiana Nurchalifah, Basil Jacob, Christian Bauckhage, and Rafet Sifa. 2022. KPI-EDGAR: A Novel Dataset and Accompanying Metric for Relation Extraction from Financial Documents. In *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1654–1659.

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H. Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez, Nicolas Chapados, and Alexandre Lacoste. 2024. WorkArena: How Capable Are Web Agents at Solving Common Knowledge Work Tasks? *Preprint*, arXiv:2403.07718.

Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Songyang Zhang, Kai Chen, Zongwen Shen, and Jidong Ge. 2023. LawBench: Benchmarking Legal Knowledge of Large Language Models. *Preprint*, arXiv:2309.16289.

Bill Gallivan. 2024. ediscovery - 80% of all documents are email. `https://web.archive.org/web/20250622120805/https://www.digitalwarroom.com/blog/ediscovery-80-of-documents-are-email-and-attachments`. Blog post, Digital WarRoom. Accessed via Internet Archive.

Albert Gong, Kamilė Stankevičiūtė, Chao Wan, Anmol Kabra, Raphael Thesmar, Johann Lee, Julius Klenke, Carla P. Gomes, and Kilian Q. Weinberger. 2025. PhantomWiki: On-Demand Datasets for Reasoning and Retrieval Evaluation. *Preprint*, arXiv:2502.20377.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 11 others. 2024. The Llama 3 Herd of Models. *Preprint*, arXiv:2407.21783.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, and 21 others. 2023. Legalbench: A Collaboratively Built Benchmark for Measuring Legal Reasoning in Large Language Models. *SSRN Electronic Journal*.

Sil Hamilton, Rebecca MM Hicke, Matthew Wilkens, and David Mimno. 2025. Too long, didn't model: Decomposing llm long-context understanding with novels. *arXiv preprint arXiv:2505.14925*.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. RULER: What's the Real Context Size of Your Long-Context Language Models? *Preprint*, arXiv:2404.06654.

Kung-Hsiang Huang, Akshara Prabhakar, Sidharth Dhawan, Yixin Mao, Huan Wang, Silvio Savarese, Caiming Xiong, Philippe Laban, and Chien-Sheng Wu. 2025. CRMArena: Understanding the Capacity of LLM Agents to Perform Professional CRM Tasks in Realistic Environments. In *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)*, pages 3830–3850, Albuquerque, New Mexico. Association for Computational Linguistics.

Yunpeng Huang, Jingwei Xu, Junyu Lai, Zixu Jiang, Taolue Chen, Zenan Li, Yuan Yao, Xiaoxing Ma, Lijuan Yang, Hao Chen, Shupeng Li, and Penghao Zhao. 2024. Advancing transformer architecture in long-context large language models: A comprehensive survey. *arXiv preprint arXiv:2311.12351*.

Feihu Jiang, Chuan Qin, Kaichun Yao, Chuyu Fang, Fuzhen Zhuang, Hengshu Zhu, and Hui Xiong. 2024. Enhancing Question Answering for Enterprise Knowledge Bases using Large Language Models. *Preprint*, arXiv:2404.08695.

Greg Kamradt. 2023. Needle In A Haystack - Pressure Testing LLMs.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling Laws for Neural Language Models. *Preprint*, arXiv:2001.08361.

Marzena Karpinska, Katherine Thai, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. One Thousand and One Pairs: A "novel" challenge for long-context language models. *Preprint*, arXiv:2406.16264.

Bryan Klimt and Yiming Yang. 2004. The enron corpus: A new dataset for email classification research. In *European conference on machine learning*, pages 217–226. Springer.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA Reading Comprehension Challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Tom Kwiatkowski, Eunsol Choi, Yoav Artzi, and Luke Zettlemoyer. 2013. Scaling Semantic Parsers with On-the-Fly Ontology Matching. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1545–1556, Seattle, Washington, USA. Association for Computational Linguistics.

Igor Labutov, Bishan Yang, and Tom Mitchell. 2019a. Learning to Learn Semantic Parsers from Natural Language Supervision. *Preprint*, arXiv:1902.08373.

Igor Labutov, Bishan Yang, Anusha Prakash, and Amos Azaria. 2019b. Multi-Relational Question Answering from Narratives: Machine Reading and Reasoning in Simulated Worlds. *Preprint*, arXiv:1902.09093.

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. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Marcelo Machado, João M B Rodrigues, Guilherme Lima, and Sandro Rama Fiorini. 2024. {LLM Store}: Leveraging Large Language Models as Sources of {Wikidata}-Structured Knowledge. In *KBC-LM'24: Knowledge Base Construction from Pre-trained Language Models Workshop at ISWC 2024*.

Laura Manor and Junyi Jessy Li. 2019. Plain English Summarization of Contracts. In *Proceedings of the Natural Legal Language Processing Workshop 2019*, pages 1–11, Minneapolis, Minnesota. Association for Computational Linguistics.

OpenAI. 2025. OpenAI MRCR: Long Context Multiple Needle in a Haystack Benchmark. https://huggingface.co/datasets/openai/mrcr.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, A. J. Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, and 45 others. 2024. GPT-4o System Card. *Preprint*, arXiv:2410.21276.

Mubashar Raza, Zarmina Jahangir, Muhammad Bilal Riaz, Muhammad Jasim Saeed, and Muhammad Awais Sattar. 2025. Industrial applications of large language models. *Scientific Reports*, 15(1):13755.

Michael J. Ryan, Danmei Xu, Chris Nivera, and Daniel Campos. 2025. EnronQA: Towards Personalized RAG over Private Documents. *Preprint*, arXiv:2505.00263.

Rishav Sahay, Arihant Jain, Purav Aggarwal, and Anoop Saladi. 2025. AutoKB: Automated Creation of Structured Knowledge Bases for Domain-Specific Support. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track)*, pages 708–723, Albuquerque, New Mexico. Association for Computational Linguistics.

Olly Styles, Sam Miller, Patricio Cerda-Mardini, Tanaya Guha, Victor Sanchez, and Bertie Vidgen. 2024. WorkBench: A Benchmark Dataset for Agents in a Realistic Workplace Setting. *Preprint*, arXiv:2405.00823.

Albert Sun, Varun Nair, Elliot Schumacher, and Anitha Kannan. 2024. CONSCENDI: A contrastive and scenario-guided distillation approach to guardrail models for virtual assistants. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4009–4030, Mexico City, Mexico. Association for Computational Linguistics.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel Ni, Heung-Yeung Shum, and Jian Guo. 2023. Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph. In *The Twelfth International Conference on Learning Representations*.

Qiang Sun, Yuanyi Luo, Wenxiao Zhang, Sirui Li, Jichunyang Li, Kai Niu, Xiangrui Kong, and Wei Liu. 2025. Docs2KG: A Human-LLM Collaborative Approach to Unified Knowledge Graph Construction from Heterogeneous Documents. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 801–804, Sydney NSW Australia. ACM.

Alon Talmor and Jonathan Berant. 2018. The Web as a Knowledge-Base for Answering Complex Questions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 641–651, New Orleans, Louisiana. Association for Computational Linguistics.

Team Gemini, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, Soroosh Mariooryad, Yifan Ding, Xinyang Geng, Fred Alcober, Roy Frostig, Mark Omernick, Lexi Walker, Cosmin Paduraru, Christina Sorokin, and 4 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *Preprint*, arXiv:2403.05530.

Dominik Tomaszuk and David Hyland-Wood. 2020. Rdf 1.1: Knowledge representation and data integration language for the web. *Symmetry*, 12(1):84.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. MuSiQue: Multi-hop Questions via Single-hop Question Composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Harsh Vishwakarma, Ankush Agarwal, Ojas Patil, Chaitanya Devaguptapu, and Mahesh Chandran. 2025. Can LLMs help you at work? a sandbox for evaluating LLM agents in enterprise environments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9178–9212, Suzhou, China. Association for Computational Linguistics.

Kiran Vodrahalli, Santiago Ontanon, Nilesh Tripuraneni, Kelvin Xu, Sanil Jain, Rakesh Shivanna, Jeffrey Hui, Nishanth Dikkala, Mehran Kazemi, Bahare Fatemi, Rohan Anil, Ethan Dyer, Siamak Shakeri, Roopali Vij, Harsh Mehta, Vinay Ramasesh, Quoc Le, Ed Chi, Yifeng Lu, and 5 others. 2024. Michelangelo: Long Context Evaluations Beyond Haystacks via Latent Structure Queries. *Preprint*, arXiv:2409.12640.

Liya Wang, David Yi, Damien Jose, John Passarelli, James Gao, Jordan Leventis, and Kang Li. 2025. Enterprise Large Language Model Evaluation Benchmark. *Preprint*, arXiv:2506.20274.

Xintao Wang, Qianwen Yang, Yongting Qiu, Jiaqing Liang, Qianyu He, Zhouhong Gu, Yanghua Xiao, and Wei Wang. 2023. KnowledGPT: Enhancing Large Language Models with Retrieval and Storage Access on Knowledge Bases. *Preprint*, arXiv:2308.11761.

Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, Yijing Xu, Haoqiang Kang, Ziyan Kuang, Chenhan Yuan, Kailai Yang, Zheheng Luo, Tianlin Zhang, Zhiwei Liu, Guojun Xiong, and 15 others. 2024. FinBen: A Holistic Financial Benchmark for Large Language Models. In *38th Conference on Neural Information Processing Systems (NeurIPS 2024) Track on Datasets and Benchmarks.*

Frank F. Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Z. Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, Mingyang Yang, Hao Yang Lu, Amaad Martin, Zhe Su, Leander Maben, Raj Mehta, Wayne Chi, Lawrence Jang, Yiqing Xie, and 2 others. 2025. TheAgentCompany: Benchmarking LLM Agents on Consequential Real World Tasks. *Preprint*, arXiv:2412.14161.

Liang Xu, Lei Zhu, Yaotong Wu, and Hang Xue. 2024. SuperCLUE-Fin: Graded Fine-Grained Analysis of Chinese LLMs on Diverse Financial Tasks and Applications. *Preprint*, arXiv:2404.19063.

Bishan Yang and Tom Mitchell. 2019. Leveraging Knowledge Bases in LSTMs for Improving Machine Reading. *Preprint*, arXiv:1902.09091.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. *Preprint*, arXiv:1412.6575.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. 2024. τ-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains. *Preprint*, arXiv:2406.12045.

Tan Yu, Wenfei Zhou, Leiyang Leiyang, Aaditya Shukla, Mmadugula Mmadugula, Pritam Gundecha, Nicholas Burnett, Anbang Xu, Viseth Viseth, Tbar Tbar, Rama Akkiraju, and Vivienne Zhang. 2025. EKRAG: Benchmark RAG for Enterprise Knowledge Question Answering. In *Proceedings of the 4th International Workshop on Knowledge-Augmented Methods for Natural Language Processing*, pages 152–159, Albuquerque, New Mexico, USA. Association for Computational Linguistics.

Bing Zhang, Mikio Takeuchi, Ryo Kawahara, Shubhi Asthana, Md Maruf Hossain, Guang-Jie Ren, Kate Soule, and Yada Zhu. 2024. Enterprise Benchmarks for Large Language Model Evaluation. *Preprint*, arXiv:2410.12857.

A Company Information Overview

Table 5 contains information about the four companies that we have generated. We have generated four companies with different attributes: name, industry, size, model, and story.

B Evidence Strings

Non-meeting evidence strings. These strings substantiate relationships between email senders and entities (teams, managers, tasks, or companies) through linguistic patterns that vary in explicitness and temporal stage.

Each relationship type contains evidence strings organized by temporal stage (start, ongoing, end) and evidence strength (explicit, implicit). Explicit strings directly state the relationship using clear relationship verbs, while implicit strings demonstrate active engagement through actions, coordination, or possessive language.

To ensure lexical diversity within implicit strings, we organize them into thematic subcategories that represent different aspects of workplace behavior.

For example, the `memberOf` relationship includes implicit subcategories such as "onboarding_to_team" for new members learning processes, "syncing_with_team_members" for ongoing coordination, and "knowledge_transfer_to_team" for members departing the team. The `memberOf` relationship contains 20 explicit strings per temporal stage and 140 implicit strings (7 subcategories with 20 strings each) per temporal stage, totaling 480 strings. Similar distributions apply to other relationship types, resulting in 3,217 total evidence string variations across all six relationship types.

Table 6 shows example evidence strings for the `memberOf` relationship.

Meeting evidence strings. These evidence strings differ from non-meeting strings by focusing on task and project progress rather than relationship formation or dissolution.

These strings are organized by progress level rather than temporal stage, with four progress bins: 0-25%, 25-50%, 50-75%, and 75-100%. Each progress bin contains both explicit strings that directly state completion percentage and implicit strings organized into thematic subcategories representing typical activities at that stage of work.

For example, the 0-25% progress bin includes implicit subcategories such as "initial_research" for gathering requirements, "project_kickoff" for launching initiatives, and "resource_gathering" for assembling team and materials. As progress advances, the subcategories shift to reflect later-stage activities: the 50-75% bin includes "beta_testing" and "integration_work", while the 75-100% bin includes "final_review", "deployment_prep", and "closing_activities".

Each progress bin contains 20 explicit strings and 120 implicit strings (6 subcategories with 20 strings each), totaling 560 strings for the `taskProgress` evidence type.

Table 7 shows example evidence strings for different progress levels.

C Topics

Non-meeting topics. For non-meeting documents, we assign topics related to either business or non-business operations. To maintain lexical diversity as the corpus grows, we scale the number of topics proportionally by creating hierarchical topic structures of varying depth, similar to the scenario-seeded approach defined in (Sun et al., 2024).

We begin with broad top-level categories. Non-business topics consistently use three main categories: "Food", "Travel", and "Transportation". Business topics vary by company domain: for example, Streamvibe (media) uses "Content Creation", "Production and Distribution", and "Audience Engagement", while Biocure (pharmaceuticals) uses "Drug Development", "Drug Approval", and "Drug Sales", and Pound (finance) uses "Deal Sourcing", "Due Diligence", and "Value Creation".

The hierarchy depth depends on corpus size. Zenith, our smallest company, requires no subcategories—its business topics are specific from the start (e.g., "API Security Standards", "Microservice Architecture Patterns"), as are its non-business topics (e.g., "Home Cooking", "Vacation Plans"). For larger companies like Streamvibe, Biocure, and Pound, we recursively expand top-level categories into deeper hierarchies using Claude Sonnet 4.5, creating subcategories at two, three, or four levels as needed. We manually review each generated topic to ensure it is appropriate for corporate email content before inclusion.

Table 9 shows the hierarchical example of topics for Biocure, our large company.

Finally, to select non-meeting topics, we use a Dirichlet topic distribution with a sparse alpha concentration of 0.75 to determine the probability at which we'd use a topic for a particular document.

Name	Industry	Size	Model	Story
Zenith	Tech	Small	B2C	Nestled in the heart of California's tech scene, Zenith Labs is a nimble, people-first B2C company crafting sleek laptops, phones, and desktops for everyday users who crave simplicity without sacrificing style. With few employees and a laid-back, startup vibe, Zenith Labs punches above its weight in the consumer electronics space, offering intuitive, affordable devices that blend minimalist design with reliable performance. Their audience? Young professionals, students, and creatives who want tech that feels personal—not corporate. Zenith Labs' motto "Keep it crisp" reflects their mission to strip away the noise and deliver clean, user-friendly experiences. Their goal is to become the go-to brand for people who want smart tech that doesn't try too hard.
Streamvibe	Media	Medium	B2C	A creative powerhouse in the entertainment industry, StreamVibe Studios produces original streaming content, podcasts, and digital campaigns for the cord-cutting generation. With talented creators, producers, and marketers spread across Los Angeles and Austin, StreamVibe has carved out a niche creating viral web series, documentary films, and branded content for Gen Z and millennial audiences. Known for their "content first, suits second" mentality, StreamVibe attracts top creative talent with flexible work arrangements, profit-sharing, and a culture that celebrates bold storytelling and authentic voices.
Biocure	Pharma	Large	B2B	Operating from a sprawling research campus with state-of-the-art laboratories, BioCure Solutions is a science-driven B2B pharmaceutical company specializing in contract research and development services for drug discovery and clinical trials. This established industry leader employs scientists, researchers, and regulatory specialists who maintain rigorous FDA compliance standards and operate under strict Good Manufacturing Practice protocols that have secured partnerships with major pharmaceutical corporations worldwide. Their mission is to become the premier CRO partner for pharmaceutical companies developing tomorrow's therapies, providing the scientific rigor and regulatory expertise that transforms promising compounds into approved medicines that improve human health globally.
Pound	Finance	X-Large	B2B	Pound Financial is a mid-market investment firm specializing in private equity and corporate restructuring for manufacturing and technology companies. The firm maintains a conservative, relationship-driven approach, focusing on long-term value creation rather than quick exits. Their clientele includes family-owned businesses seeking growth capital and Fortune 500 companies navigating complex financial transitions. Pound Financial's reputation for discretion, thorough due diligence, and strategic guidance has made them a trusted partner for executives facing critical financial decisions.

Table 5: Company/KB Information Overview.

Temporal Stage	Strength	Examples
Start	Explicit	I just joined {{object}} and wanted to introduce myself I'm excited to announce that I'm now part of {{object}} I recently became a member of {{object}} *... 17 more*
	Implicit	**onboarding_to_team:** Just going through the {{object}} onboarding materials today **learning_team_processes:** Still getting familiar with how {{object}} runs things **getting_added_to_distribution_lists:** Just got added to the {{object}} email lists *... 4 more subcategories, 20 strings each*
Ongoing	Explicit	As a member of {{object}}, I wanted to share our latest findings I'm on {{object}}, and we've been tracking this issue My group, {{object}}, has identified a solution *... 17 more*
	Implicit	**discussing_team_priorities:** This aligns with {{object}}'s Q1 priorities **syncing_with_team_members:** Let me check with my {{object}} teammates first **contributing_to_team_decisions:** {{object}} voted on this approach last week *... 4 more subcategories, 20 strings each*
End	Explicit	I'm leaving {{object}}, effective today My time with {{object}} is coming to an end I'm departing from {{object}} today *... 17 more*
	Implicit	**knowledge_transfer_to_team:** Documenting everything for {{object}} before I transition off **handing_off_team_responsibilities:** Transitioning my {{object}} responsibilities to Sarah **wrapping_up_team_projects:** Finishing up my last {{object}} project this week *... 4 more subcategories, 20 strings each*

Table 6: Example evidence strings for the `memberOf` relationship showing temporal stages and evidence strength levels. Explicit strings directly state team membership, while implicit strings demonstrate engagement through workplace actions. The placeholder `{{object}}` represents the team name.

These distributions are illustrated in Figure 5. The actual topic distributions after generation are illustrated in Figure 6.

Meeting topics. Meeting documents require a different approach to topic assignment, as their content is shaped by meeting dynamics rather than general business operations. We select meeting topics from a separate pool, conditioned on meeting type to reflect typical discussion patterns. For example, direct report meetings commonly cover "Employee Performance Review", while executive meetings focus on "Company Strategy and Vision Alignment".

Not all meeting emails receive topic assignments. Blank calendar invites and similar minimal-content emails lack substantive discussion and are therefore excluded from topic assignment and from our Topic Classification, Topic QA, and Integrated QA evaluation tasks.

Table 10 lists the meeting topics that we use.

D Meetings

We simulate meetings for employees based on their organizational roles and relationships. Table 11 defines the six meeting types supported in our simulation framework. Each meeting type is characterized by several parameters: a description of the participants, the probability of generating meeting minutes, the probability of generating an agenda, the probability of leaving the calendar event empty (without minutes or agenda), the meeting frequency, the designated organizer, and the knowledge graph relationships that the meeting substantiates.

For each meeting type, we determine participants from relevant organizational relationships. For example, direct report meetings involve an employee and their manager, while team meetings include all employees who belong to or lead a particular team. Meetings are scheduled according to their specified frequencies (weekly, biweekly, or monthly), and associated artifacts (meeting minutes and agendas) are generated based on the probabilities shown in Table 11. The "Empty Cal." column indicates the probability that a meeting appears on calendars without any accompanying documentation.

The "Substantiates" column indicates which knowledge graph relationships are evidenced or reinforced by each meeting type. For instance, direct report meetings substantiate task progress, re-

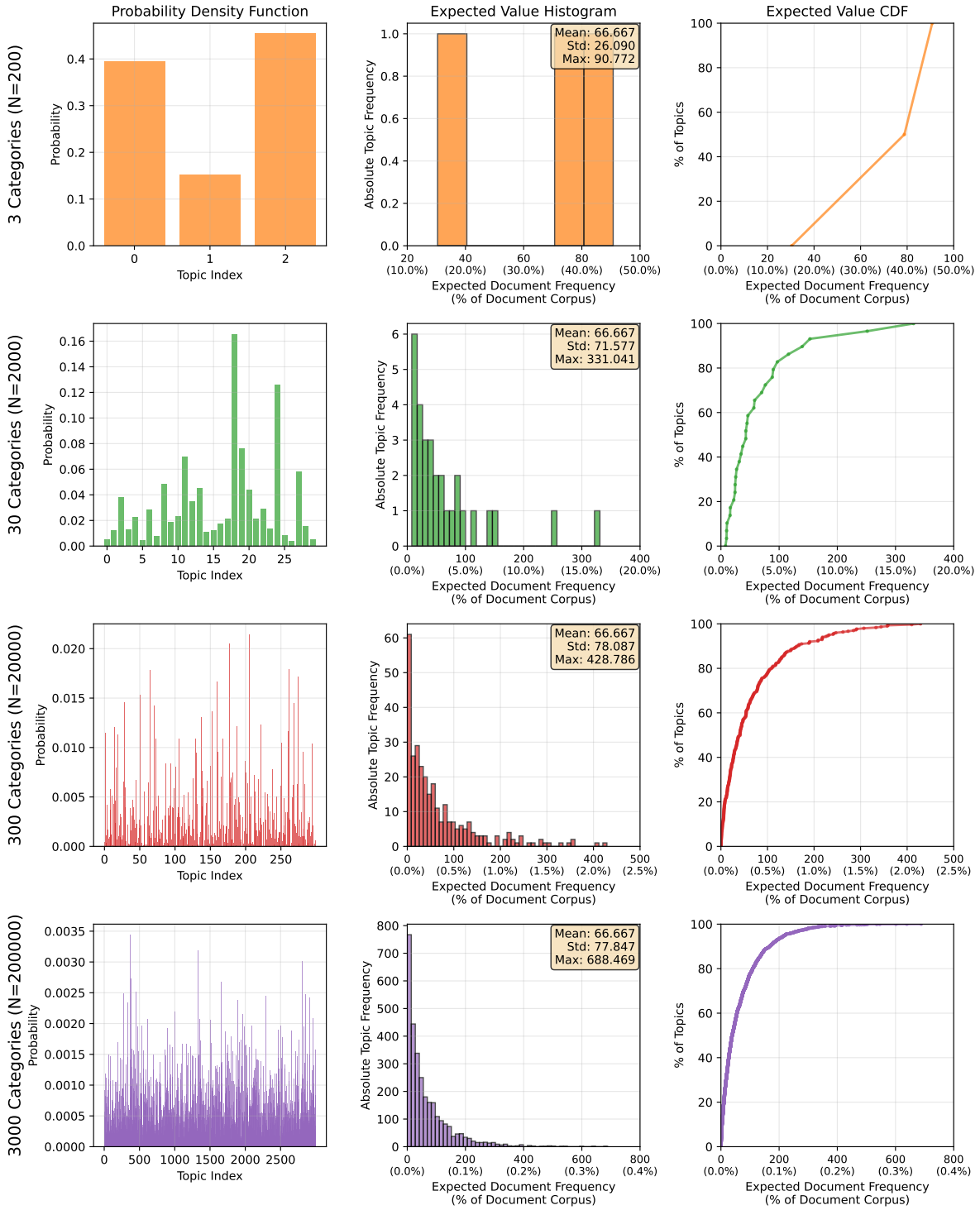

Figure 5: Summary of theoretical Dirichlet distributions used for topic sampling when generating non-meeting documents.

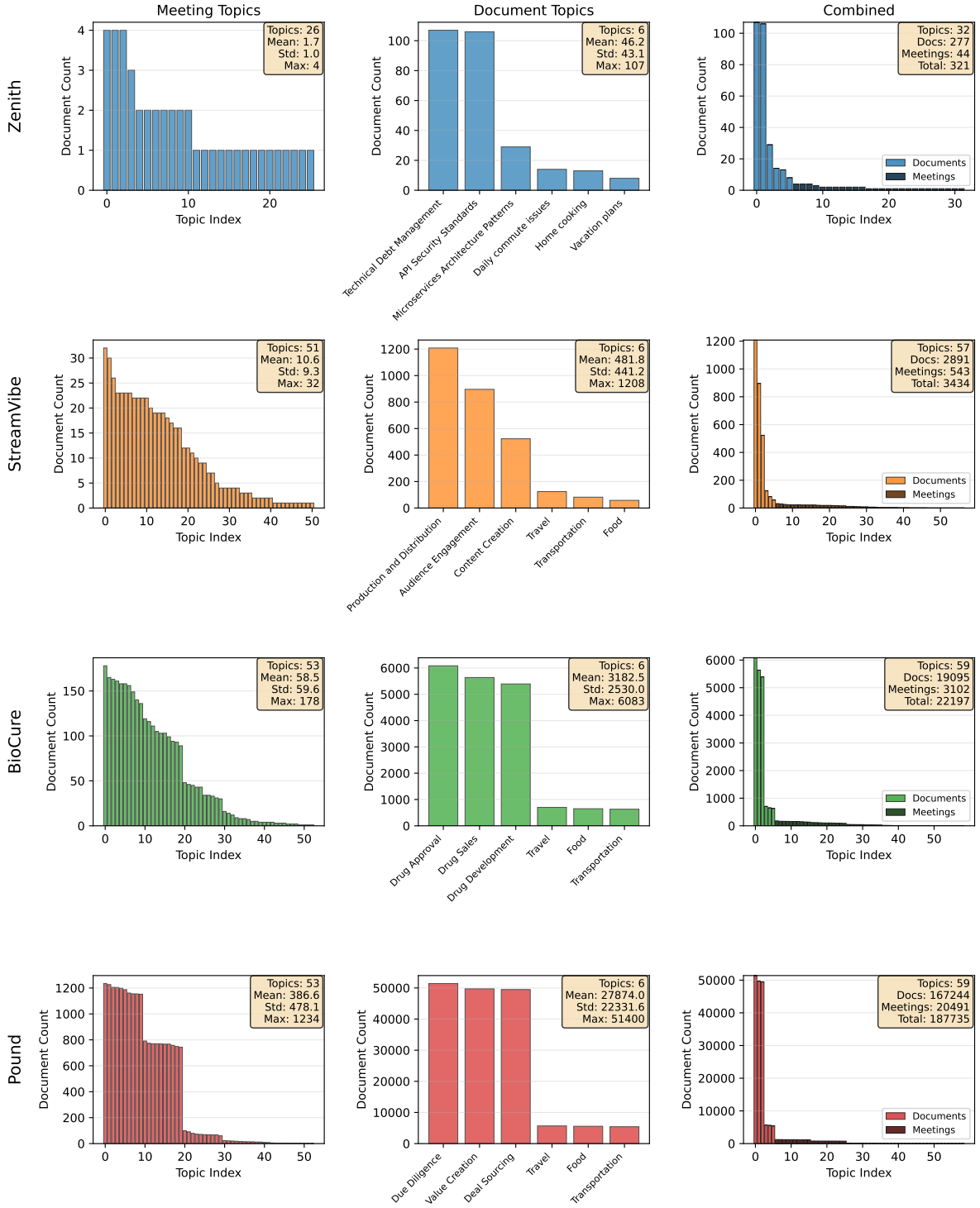

Figure 6: Summary of empirical topic distributions of generated documents across all four KBs. Includes meeting and non-meeting topics. See Appendix C for more details on topics.

Progress Level	Strength	Examples
0-25%	Explicit	{{employee}} {{have_verb}} {{task}} about 20% done {{employee}} {{be_verb}} roughly a quarter of the way through {{task}} {{task}} is still in early stages with {{employee}}, around 15-25% complete *... 17 more*
	Implicit	**initial_research:** {{employee}} {{be_verb}} still gathering requirements for {{task}} **project_kickoff:** {{employee}} sent out the project charter for {{task}} today **early_planning:** {{employee}} {{be_verb}} sketching out the roadmap for {{task}} *... 3 more subcategories, 20 strings each*
25-50%	Explicit	{{employee}} {{have_verb}} {{task}} about 40% complete {{employee}} {{be_verb}} roughly a third done with {{task}} {{task}} is coming along with {{employee}}, around 35% finished *... 17 more*
	Implicit	**prototype_development:** {{employee}} just got the first prototype working for {{task}} **foundation_complete:** {{employee}} {{have_verb}} the core framework built for {{task}} **initial_deliverables:** {{employee}} submitted the first milestone deliverable for {{task}} *... 3 more subcategories, 20 strings each*
50-75%	Explicit	{{employee}} {{have_verb}} {{task}} about 60% done now {{employee}} {{be_verb}} past halfway on {{task}}, around 65% complete {{task}} is roughly two-thirds finished by {{employee}} *... 17 more*
	Implicit	**beta_testing:** {{employee}} {{be_verb}} running trial programs for {{task}} **debugging_phase:** {{employee}} {{be_verb}} fixing the remaining problems in {{task}} **integration_work:** {{employee}} {{be_verb}} bringing together all the pieces of {{task}} *... 3 more subcategories, 20 strings each*
75-100%	Explicit	{{employee}} {{have_verb}} {{task}} nearly done, about 85% complete {{employee}} {{be_verb}} almost finished with {{task}}, around 80-90% there {{task}} is in the final stretch with {{employee}}, roughly 80% done *... 17 more*
	Implicit	**final_review:** {{employee}} {{be_verb}} doing the final review of {{task}} **polishing_details:** {{employee}} {{be_verb}} polishing the documentation for {{task}} **deployment_prep:** {{employee}} {{be_verb}} preparing {{task}} for implementation *... 3 more subcategories, 20 strings each*

Table 7: Example evidence strings for the `taskProgress` relationship showing progress levels and evidence strength. Explicit strings directly state completion percentages, while implicit strings demonstrate typical activities at each stage. The placeholders `{{employee}}` and `{{task}}` represent the employee name and task name respectively.

Size	Documents	Non-Business	Business
S	$\approx 4 \cdot 10^2$	$3 \cdot 10^0$	$3 \cdot 10^0$
M	$\approx 4 \cdot 10^3$	$3 \cdot 10^1$	$3 \cdot 10^1$
L	$\approx 3 \cdot 10^4$	$3 \cdot 10^2$	$3 \cdot 10^2$
XL	$\approx 2 \cdot 10^5$	$3 \cdot 10^3$	$3 \cdot 10^3$

Table 8: Breakdown of the number of topics used to condition the generation of documents in CORPORATEBENCH based on different sizes of the company. We provide more detail in Appendix C.

porting relationships, and work assignments, while executive meetings provide evidence of project progress and employee titles. This ensures that meetings generate realistic traces of organizational activity that align with the underlying knowledge graph structure.

This simulation process results in each employee having an average of 6 meetings per quarter, with individual counts varying based on their position, team membership, project involvement, and reporting relationships.

E ETL Pipeline

ETL for KB-Gold. To extract ground truth from knowledge base files into postgres databases accessible to our agent, we use the StoryBeat KB API to access entities, relationships, and documents.

Extract: We use the StoryBeat KB API's entity, relationship, and document interfaces to extract data from .kb files. The extraction uses generator functions to process data in configurable batches (default: 1000 items) rather than loading entire datasets into memory.

Transform: Entities are mapped from KB properties to database columns based on entity type (Company, Employee, Department, etc.). Each entity type has specific property mappings (e.g., Employee uses `hasName` for name, `hasAlias` for aliases). Relationships are extracted with their tem-

Level 0	Level 1	Level 2	Level 3 (Examples)
Non-Business	Food	Cooking	Grilling techniques shared (... *9 more*)
		Baking	Bread making adventures (... *9 more*)
		Dining Out	Restaurant service reviews (... *9 more*)
		Food Culture	Cultural food traditions (... *9 more*)
		Nutrition	Vitamin supplement discussions (... *9 more*)
		... *5 more subcategories*	
	Travel	Destinations	Mountain hiking trails (... *9 more*)
		Accommodations	Hotel loyalty programs (... *9 more*)
		Activities	Adventure sports planning (... *9 more*)
		Transportation Methods	Train journey experiences (... *9 more*)
		Travel Planning	Itinerary creation process (... *9 more*)
		... *5 more subcategories*	
	Transportation	Public Transit	Bus schedule reliability (... *9 more*)
		Personal Vehicles	Car model comparisons (... *9 more*)
		Alternative Transport	Electric scooter trials (... *9 more*)
		Commuting	Route optimization strategies (... *9 more*)
		Ride Sharing	App comparison reviews (... *9 more*)
		... *5 more subcategories*	
Business	Drug Development	Preclinical Research	Animal Model Selection (... *9 more*)
		Clinical Trial Planning	Protocol Design Elements (... *9 more*)
		Regulatory Strategy	Submission Pathway Selection (... *9 more*)
		Partnership Collaborations	Due Process (... *9 more*)
		Biomarker Identification	Biomarker Discovery Methods (... *9 more*)
		... *5 more subcategories*	
	Drug Approval	Regulatory Submission Preparation	Module Compilation Process (... *9 more*)
		FDA Communication	Meeting Request Preparation (... *9 more*)
		Clinical Data Analysis	Statistical Trial Evaluation (... *9 more*)
		Safety Reporting	Adverse Event Classification (... *9 more*)
		Manufacturing Compliance	GMP Inspection Preparation (... *9 more*)
		... *5 more subcategories*	
	Drug Sales	Market Access Strategy	Payer Landscape Analysis (... *9 more*)
		Pricing Negotiations	Price Justification Strategy (... *9 more*)
		Sales Force Training	Product Knowledge Development (... *9 more*)
		Key Opinion Leader Engagement	KOL Identification Strategy (... *9 more*)
		Competitive Intelligence	Market Landscape Analysis (... *9 more*)
		... *5 more subcategories*	

Table 9: Example non-meeting topic hierarchy for Biocure showing the four-level structure from top-level categories down to specific discussion topics.

poral information (`beginning`, `end`) and classified as "ongoing" or "bounded". Documents are processed separately using the KB documents API, with properties including sender/recipient information, dates, topics, and meeting references. Array fields (aliases, recipient lists) are converted from string representations to PostgreSQL array types using `ast.literal_eval`.

Load: Data is loaded into PostgreSQL using batch INSERT operations within transactions for consistency. Entity tables use the KB entity ID as primary key (e.g., `employee_EMP-D44FE8C5`). The pipeline supports multiple schemas (one per company/knowledge base), with schema names inferred from KB filenames. Duplicate detection uses `ON CONFLICT DO NOTHING` to handle idempotent loading.

ETL for RAG. The RAG pipeline builds vector embeddings for document retrieval.

Extract: Documents are exported from KB files to a temporary directory using the KB documents API.

Transform: Document content is truncated to 22,000 characters (≈8,000 tokens) for embedding generation to stay within model limits. Embeddings are generated using OpenAI's text-embedding-3-small model (1536 dimensions) with parallel processing (4 workers by default).

Load: Embeddings are stored in PostgreSQL using the pgvector extension with HNSW indexing for efficient similarity search. Batch inserts (500 records per batch) optimize database write performance. Duplicate documents are detected and skipped to avoid redundant API calls. Full document content (not truncated) is stored alongside

Meeting Type	Topics
Direct Report	Employee Performance Review, Goal Setting and Progress Check-In, Career Development Discussion, Feedback and Coaching Session, Workload and Prioritization Review, Upcoming Deadlines and Deliverables, Team Dynamics and Collaboration Feedback, Skill Growth and Training Opportunities, Manager-Report Relationship Check-In, Quarterly Performance Summary
Team	Weekly Team Standup, Team Goals and KPIs Review, Process Improvement Discussion, Cross-Functional Collaboration Update, Retrospective and Lessons Learned, Upcoming Projects and Assignments, Team Communication and Workflow Alignment, Resource Planning and Allocation, Team Morale and Culture Check, Problem-Solving and Roadblock Discussion
Task	Task Progress and Status Update, Dependency and Blocker Resolution, Timeline and Deliverable Planning, Quality Review and Standards Alignment, Task Ownership and Accountability, Cross-Task Integration Discussion, Testing and Validation Checkpoint, Task Documentation and Handover, Risk and Issue Review, Next Steps and Action Items
Executive	Company Strategy and Vision Alignment, Financial Performance and Budget Review, Key Metrics and KPIs Discussion, Organizational Priorities and Focus Areas, Market Trends and Competitive Analysis, Resource Allocation and Headcount Planning, Operational Efficiency Review, Major Risks and Mitigation Planning, Executive Decisions and Approvals, Quarterly Business Review
Project	Project Kickoff and Scope Definition, Milestone Progress Review, Timeline and Deliverable Alignment, Resource and Budget Status, Stakeholder Communication Plan, Risk Management and Mitigation Discussion, Project Dependencies and Coordination, Quality Assurance and Testing Review, Project Retrospective and Lessons Learned, Next Phase Planning and Execution
Department	Department Strategy and Goals Review, Cross-Team Coordination Update, Department Resource Planning, Organizational Changes and Updates, Department Performance Metrics Review, Inter-departmental Collaboration Discussion, Budget and Resource Allocation Review, Department Culture and Team Building, Major Initiative Planning, Department Communication and Updates

Table 10: Meeting topics.

Meeting Type	Description	Minutes	Agenda	Empty Cal.	Frequency	Organizer	Substantiates
direct_report	employee and their manager	30%	30%	40%	weekly	manager	task progress, reports to, works on
team	employees that belong and lead a team	25%	25%	50%	monthly	team lead	task progress, works on, member of
task	employees collaborating on a task	30%	30%	40%	biweekly	any	task progress, works on
executive	employees that are executives	50%	50%	0%	monthly	any	project progress, title
project	employees that work on tasks part of a project	50%	50%	0%	monthly	team lead of the project	project progress, works on, has task
department	employees that lead the department and the teams part of it	50%	50%	0%	monthly	department lead	has task, team belongs to department, department belongs to company

Table 11: Supported Meeting Types.

embeddings for retrieval.

F QA Benchmarks

In the following section, we outline sample questions from three QA dataset types across the different companies they are generated from. In each example, we include the minimal set of documents required to review/analyze to be able to fully answer the presented question. There may be multiple minimal sets of documents to answer said questions, of which we select one. The selected

questions are a small subset of all questions in the dataset and aim to show the diversity of questions, answers, and documents. Many questions require analyzing many documents (e.g. "Who are the employees of pound financial group llc?"), but these questions are omitted for the sake of conciseness.

For each question, we measure the number of constraints, which represents the number of filters on the documents or KB that an agent must apply to answer a question. However, this doesn't necessarily correlate with question difficulty, because low constraint count questions may be more difficult to answer due to their expansive breadth (see example in paragraph above).

G KB QA

Question	Who participated in the meeting titled cross-team protocol alignment - work session?
Answer	James Moreira, James Vera
Company/KB	Streamvibe
Answer Type	List[str]
Constraints	1

calendar_invite_james_vera_f685.txt

```
Subject: Cross-team protocol alignment -
Work Session

From: James Vera
<vera.Jamie@streamvibestudios.com>
To: James Vera
<vera.Jamie@streamvibestudios.com>, James
Moreira
<Jim.moreira@streamvibestudios.com>
Date: 2024-02-22

CALENDAR INVITE

You are invited to: Cross-team protocol
alignment - Work Session
Scheduled for: 2024-02-22

Organizer: James Vera
(vera.Jamie@streamvibestudios.com)

Attendees:
  - James
Vera (vera.Jamie@streamvibestudios.com)
  - James Moreira
(Jim.moreira@streamvibestudios.com)

This meeting has been scheduled by James
Vera.

Please accept or decline this invitation
at your earliest convenience.
```

The calendar invite clearly lists out all attendees: `James Vera`, `James Moriera`. We judge these responses as sets, so ordering doesn't matter.

Question	Who organized the target company due diligence discussion meeting?
Answer	Birgitta Lagarde
Company/KB	Pound
Answer Type	str
Constraints	1

meeting_minute_Quality _Review_and_Standards_Alignment_07a7.txt

```
Message-ID:
61f61ba7-8cd2-4a17-a311-6cdfd059f7bc
From: Birgitta Lagarde
<blagarde@poundfinancialgroupllc.com>
To: Piermaria Ferguson
<pferguson@poundfinancialgroupllc.com>
Date: 2024-01-03
Subject: Target company due diligence
Discussion
MIME-Version: 1.0
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 7bit

Piermaria,

Thanks for making time to sync up on the
target company due
diligence work we've got going. I wanted
to recap what we covered in our last
meeting and make sure we're both tracking
the same priorities as we move forward
with this.

We spent some good time going through our
quality review process and making sure
our standards are aligned across the
board. We talked through the key
documentation we need to pull together,
identified some gaps in our initial
assessment, and went over what the next
phase of verification looks like. It was
helpful to lock in on our approach and
get clear on who's handling what pieces.

Here's where things stand with our
progress:

Target company due
diligence is advancing steadily with you
and I, around 30-40% finished.

I think we're in a solid spot to keep
pushing forward on this. Let's keep the
momentum going and touch base again soon
on any blockers or adjustments we need to
make.

Thank you,
Birgitta

Birgitta Lagarde
Analyst
Pound Financial Group LLC

blagarde@poundfinancialgroupllc.com
Desk: +1 (650) 925-3434
Mobile: +1 (650) 205-2167
[INSERT_BOX_LOGO]
```

The meeting minutes document implicitly indicates that `Birgitta Lagarde`, as the organizer, is sending meeting minutes to the sole other attendee Piermaria Ferguson.

H Topic QA

Question	When was drug approval first mentioned?
Answer	2024-01-01
Company/KB	Biocure
Answer Type	date
Constraints	1

email_Label_Negotiation_Process_546e.txt

```
Message-ID:
3ce33e9f-3d3c-4100-bf30-b2fc26cbbd05
From: Jeronimo Zobel
<jeronimo.zobel@yahoo.com>
To: Manuella Barrios <manuella@gmail.com>
Date: 2024-01-01
Subject: Update on drug
approval labeling coordination
MIME-Version: 1.0
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 7bit

Hi Manuella, I wanted to touch base on a
couple of items related to our business
operations and the drug
approval process. As we continue managing
our labeling requirements across various
products, I've been thinking through how
we can streamline our approach to the
label negotiation process with regulatory
bodies. The negotiations have been
progressing, but I believe we could
benefit from a more structured timeline
and clearer documentation of our
discussions with the FDA.

Recently, manuella, checking in with you
as my direct supervisor, and I wanted to
ensure we're aligned on priorities for
the labeling requirements phase. The
label negotiation process is becoming
more complex as we handle multiple
products simultaneously, and having
direct oversight will help us maintain
consistency across submissions. I think
it would be valuable to establish
checkpoints where we review our
negotiation strategy and any feedback
we've received from regulatory reviewers.

I'd like to schedule a brief meeting this
week to discuss our current bottlenecks
in the drug
approval process, particularly around the
labeling timeline. Once we optimize the
label negotiation process, I believe we
can accelerate our overall business
operations in this area. Let me know your
availability and if there are specific
concerns you'd like to address.

Thank you,
Jeronimo Zobel

Jeronimo Zobel
Systems Administrator
+1 (650) 682-8707 |
jeronimo.z@biocuresolutions.com
```

The email is the first mention of `Drug Approval`. The topic hierarchy for this email is as follows: (0) `Label Negotiation Process`, (1) `Labeling Requirements`, and (2) `Drug Approval`. See Appendix C for a more detailed explanation on topic levels. The higher-level topics are interwoven into the main content of the email, which is centered around the more granular topic of `Label Negotiation Process`.

I Integrated QA

Question	Did anyone who works at Zenith Labs send emails about Daily commute issues during month 1 of 2024?
Answer	true
Company/KB	Zenith
Answer Type	bool
Constraints	4

email_Daily_commute_issues_5201.txt

```
Message-ID:
f57b61b0-4543-423d-9172-e8f1753abb7f
From: Albert Garcia
<Bert.garcia@zenithlabs.com>
To: Emilia Anglada <eanglada@hotmail.com>
Date: 2024-01-10
Subject: Morning commute struggles
MIME-Version: 1.0
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 7bit

Hi Emilia,

I wanted to check in since we've both
been navigating some rough commute
situations lately. The traffic on my
usual route has been absolutely brutal
the past couple of weeks, and I've been
leaving earlier just to make it to the
office on time. I know you mentioned
something similar last week when we were
chatting by the desks.

I've been thinking about how these daily
commute issues really affect our whole
day and energy levels at work. When
you're stuck in traffic or dealing with
transit delays, it's hard to come in
feeling fresh and ready to tackle complex
projects. Building on that, accepted the
firmware performance benchmarking role
this morning, so I've been reflecting on
how important it is to stay focused even
when the morning doesn't go as planned.

I've actually started trying out a
different route a couple times this week,
and it's made a noticeable difference on
days when the main highway is congested.
Leaving just fifteen minutes earlier has
been a game-changer for me, and it gives
me time to grab coffee and settle in
before diving into emails. Maybe we could
compare notes on what's been working for
each of us?

Anyway, just wanted to reach out and see
if you've found any good solutions to the
commute chaos. It seems like a lot of us
in the office have been dealing with
similar frustrations, so I figured it
might be worth discussing what strategies
have helped. Hope things smooth out for
you soon!

Thanks,
Albert Garcia
Marketing Specialist, Sales & Marketing
Zenith Labs

+1 (415) 400-8511 |
Bert.garcia@zenithlabs.com
[INSERT_HORIZONTAL_LOGO]
```

In this Email, the sender `Albert Garcia` can be identified as a Zenith Labs employee by his signature or his email domain. The topic of the email is `Daily commute issues` and it was sent on `2024-01-10`, which falls in the date range of the question.

Question	How many of jared chartier's direct reports sent emails about technical debt management after 29 Mar 2024?
Answer	0
Company/KB	Zenith
Answer Type	int
Constraints	3

email_Technical_Debt_Management_72dc.txt

```
Message-ID:
54b67748-00c1-4d5c-b996-62c9e0241545
From: Emilia
Anglada <eanglada@hotmail.com>
To: Jared
Chartier <jaredchartier@gmail.com>
Date: 2024-01-10
Subject: Addressing code quality concerns
in our operations
MIME-Version: 1.0
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 7bit

Hi Jared, I've been reflecting on how our
current business operations could benefit
from a more intentional approach to
managing technical
debt. As our codebase continues to grow,
I'm noticing that we're accumulating
layers of complexity that might slow down
our team's ability to iterate
efficiently. I think it would be valuable
for us to establish a clearer strategy
around this.

Building on that concern, Jared, I wanted
to get your direction here on how we
should prioritize which areas to tackle
first. I'm thinking we could start by
identifying the components that have the
highest maintenance burden and then work
through a systematic refactoring plan.
Would you be open to discussing this
further so we can align on the best path
forward for our technical infrastructure?

Best,
Emilia Anglada
Marketing Specialist
+1 (650) 524-2805
```

This is the only email about `Technical Debt Management` that a direct report of `Jared Chartier` has sent. Due to the size of the Zenith Labs company, `Emilia Anglada` is also the only direct report. The email was sent on January 10, 2024, so it doesn't fall within the date range of the question. Also, the mention of

"I want to get your direction here" serves as an implicit mention of Emilia reporting to Jared.

J QA Dataset Assumptions

We explicitly state the following assumptions and details we were using to create our QA ensure clarity and proper use:

- Ground truth answers are obtained through manually-verified SPARQL queries executed on the underlying knowledge base. This verification process ensures that the correct answers are returned regardless of which template variables are instantiated.
- All questions requesting lists (e.g., "Who are all the employees of X?") return results without duplicate entries. This deduplication is applied consistently across the dataset.
- Questions exclusively target regular email topics and intentionally exclude meeting-related topics, as these constitute distinct semantic categories with different structural properties.
- Due to the scale of the Pound knowledge base, minimally-constrained questions often yield extremely large answer sets. For example, the query "Who is an employee of Pound?" returns several thousand names, potentially exceeding typical LLM context window limits. In contrast, questions with additional constraints produce more manageable, specific answer sets that are better suited for LLM agent processing.
- The answer key generation process applies SELECT DISTINCT operations to remove duplicate names from result sets. Consequently, a minor discrepancy may exist between count-based answers and the actual length of returned name lists, as overlapping names are consolidated into single entries.

K Experimental Setup

KB Evaluation

- **Datasets:** All documents from the four company datasets from Table 1.
- **Models:** Claude Haiku 4.5, GPT-5 Nano, Gemini 2.5 Flash Lite.
- **Prompting:** Few-shot with 8 examples (prompt template in Appendix P). We selected samples to properly cover diverse entity/relationship types when providing examples.
- **Entity Evaluation:** Per-type precision, recall, and F_1, macro-averaged across types. TP: matched entities; FP: spurious extractions; FN: missed ground truth entities.
- **Relationship Evaluation:** Dual evaluation measuring (1) structural correctness (triple exists) and (2) temporal accuracy (triple with correct dates). Per-type metrics macro-averaged across relationship types. Temporal evaluation requires both correct triple structure and matching start/end dates.
- **Implementation:** Temperature=1.0, max tokens=8,192, exponential backoff retry (max 3 attempts, 60s timeout).

Topic Classification

- **Datasets:** Biocure (L, 31 topics) and Pound (XL, 17 topics) from Table 1. Topic definitions in Table 15.
- **Models:** Claude Haiku 4.5, Claude Sonnet 4.5, GPT-5 Nano, GPT-5.1, and Gemini 2.5 Flash Lite.
- **Baseline:** TF-IDF features with Logistic Regression classifier.
- **Train/Test Split:** Stratified sampling with $k \in \{0, 10, 50\}$ training examples for LLMs and $k \in \{10, 50, 100, 1000, 10000, 100000\}$ for baseline. Test set: 1,000 examples per configuration (or maximum available). 5 repetitions with different random splits.
- **Prompting:** Zero-shot ($k = 0$) or few-shot ($k > 0$) examples per topic (template in Appendix Q).
- **Evaluation:** Per-topic precision, recall, and F_1, macro-averaged across topics.
- **Implementation:** Temperature=1.0, max tokens=500, exponential backoff retry (max 3 attempts, 60s timeout).

QA

- **Datasets**: KB QA, Topic QA, and Integrated QA. 250 Questions * 4 KB sizes * 3 QA Datasets = 3k questions. Single full dataset run.
- **Models**: Claude Haiku 4.5, Claude Sonnet 4.5, GPT-5 Nano, GPT-5.1, and Gemini 2.5 Flash Lite. Used through Anthropic API, OpenAI API, and OpenRouter respectively.

- **Methods**: RAG ($k = 10$): Agent with access to a tool to query a vector db. KB: Agent with access to a tool to query a Postgres DB with SQL. DBs contain ground truth information extracted from the KB and follow a standard relational schema (not relevant for RAG Agent). Database access is restricted for each dataset. Restriction applies based on KB, method, and dataset to ensure no data leakage happens: e.g. when answering questions in Topic QA for Biocure, the KB agent will exclusively have access to the table that stores information on the documents from Biocure.
- **Evaluation**: Case-insensitive exact match (string, date, integer, boolean) scoring 0/1 and F_1 score for set[string] questions to allow models to obtain partial credits (as these questions are typically the hardest to answer). For boolean we accepted alternative wording such as "Yes" and "No".
- **Implementation**: Pydantic AI agent with custom tools to retrieve from Vector DB (RAG) or Postgres DB (KB). Tool call limit of 5. Few shot examples (specific to each QA dataset) were provided for each prompt.

L QA Results

In this section, we visualize the results we have for QA in graphs. Figure 7 shows the QA results by model and Figure 8 shows the QA results by method.

M KB Evaluation Results

Table 12 reveals distinct error patterns across relationship types and models. For non-temporal relationships, Gemini 2.5 Flash Lite exhibits the highest over-prediction rate for worksOn (69%), while all models show substantial under-prediction of attend relationships (63–71% FN rates). Temporal relationship extraction proves more challenging: Gemini 2.5 Flash Lite over-predicts worksOn at 56%, while Haiku 4.5 shows notable over-prediction of attend (45%) and organize (25%). Under-prediction of attend remains problematic across temporal contexts (50–54% FN rates for all models). These patterns suggest that models struggle with event-based relationships (attend, organize) regardless of temporal context, while structural relationships (worksOn, memberOf) are prone to over-prediction, particularly for Gemini 2.5 Flash Lite. The relatively balanced error rates for core relationships like reportsTo and worksAt (mostly < 10%) indicate better model performance for hierarchical organizational structures.

N Topic Classification Results

Table 13 presents topic classification performance across models and training sizes. On Biocure (L), all models achieve strong zero-shot performance (F1: 0.872–0.950), with minimal improvement in few-shot settings (50-shot F1: 0.877–0.982). The TF-IDF+LR baseline demonstrates clear data scaling, reaching F1 of 0.993 with 10K examples – comparable to LLM zero-shot performance but requiring substantially more training data. On Pound (XL), LLMs show moderate zero-shot performance (F1: 0.357–0.531) with notable few-shot improvements (50-shot F1: 0.598–0.751). The data-rich baseline achieves superior performance at 100K examples (F1: 0.983), outperforming all LLMs.

O QA Error Analysis

In total, we ran 250 questions across 5 LLM models for 4 companies in 3 QA settings with 2 methods (RAG and KB). Thus, we had 30,000 total model runs. Table 14 shows the programmatic errors that we ran into for each model in each setting (Topic QA, KB QA, and Integrated QA).

Our analysis reveals substantial variation in error rates across models, with GPT-5.1 experiencing the highest failure rate (247 errors, 4.1% of its runs) and Sonnet 4.5 showing the most robust performance (91 errors, 3.0% of its runs). The total error count across all models was 697 out of 30,000 runs, representing a 2.3% overall failure rate.

KB-based queries produced 4.7× more errors than RAG-based queries (547 vs. 150 errors). This disparity suggests that knowledge base integration introduces additional failure modes, particularly related to tool calling and output validation. The complexity of KB queries—which require models to navigate structured data and execute multiple tool calls— contributes to this elevated error rate.

Error rates exhibit a clear positive correlation with query complexity (S→M→L→XL). For KB queries, XL-scale questions generated 175 errors compared to just 107 errors for S-scale questions. This pattern is particularly pronounced for context length issues: "Prompt too long" errors occur almost exclusively at L and XL scales, accounting for 72 such errors. This suggests that as queries grow in complexity, models struggle with context

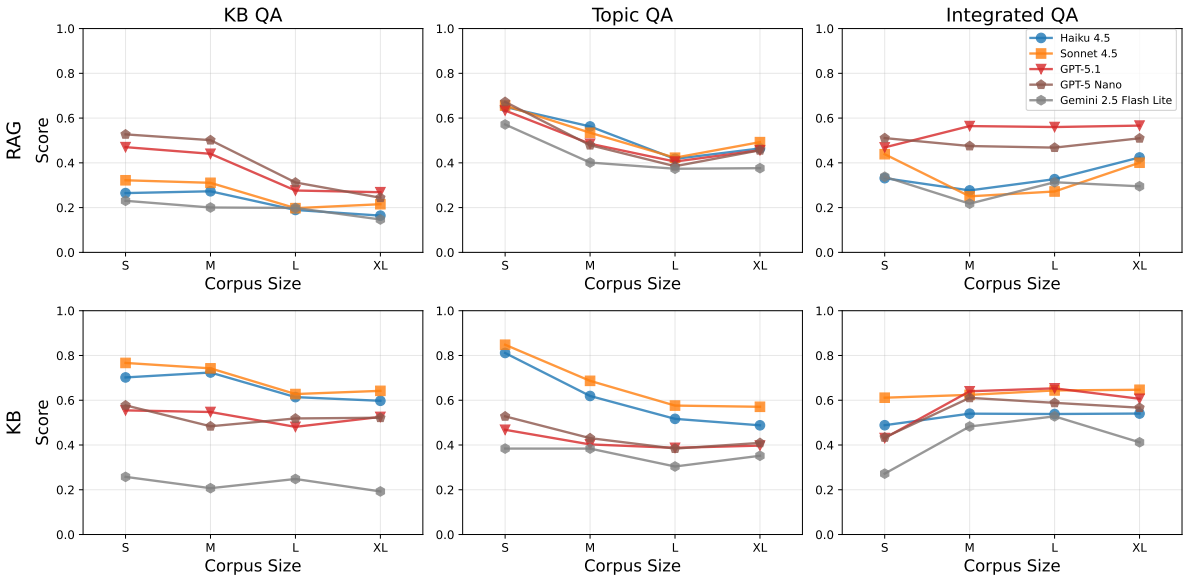

Figure 7: Visualization of results presented in Table 4.

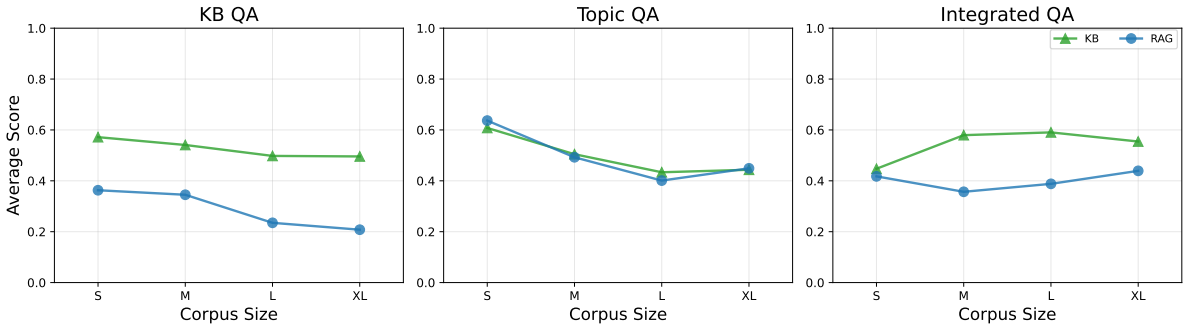

Figure 8: Visualization of results presented in Table 4 aggregated by method.

Error Type	Relationship	Gemini 2.5 FL	Haiku 4.5	GPT-5 Nano
Non-Temporal Relationships				
Over-predicts (FP)	worksOn	69%	64%	43%
	worksAt	7%	–	3%
	reportsTo	8%	–	4%
	memberOf	16%	12%	50%
	attend	–	16%	–
	organize	–	8%	–
Under-predicts (FN)	worksOn	8%	15%	16%
	worksAt	5%	–	–
	reportsTo	1%	6%	4%
	memberOf	3%	3%	3%
	attend	68%	71%	63%
	organize	15%	5%	14%
Temporal Relationships				
Over-predicts (FP)	worksOn	56%	23%	43%
	worksAt	4%	1%	16%
	reportsTo	21%	–	4%
	memberOf	19%	6%	37%
	attend	–	45%	–
	organize	–	25%	–
Under-predicts (FN)	worksOn	31%	34%	36%
	memberOf	3%	3%	3%
	attend	54%	52%	50%
	organize	12%	11%	11%

Table 12: Error Distribution by Relationship Type and Model

Model	Training Size	Biocure (L)			Pound (XL)		
		Precision	Recall	F_1	Precision	Recall	F_1
Haiku 4.5	0	.917±.00	.887±.01	.885±.01	.550±.01	.410±.02	.357±.01
	10	.950±.01	.959±.01	.951±.01	.649±.01	.650±.01	.594±.01
	50	.963±.01	.967±.00	**.963±.01**	.697±.01	.733±.01	**.685±.01**
Sonnet 4.5	0	.875±.04	.873±.04	.872±.04	.583±.01	.544±.02	.517±.02
	10	.963±.00	.966±.01	.962±.01	.681±.03	.692±.03	.661±.03
	50	.979±.00	.982±.00	**.980±.00**	.748±.02	.777±.02	**.751±.02**
GPT-5 Nano	0	.905±.00	.901±.01	**.898±.01**	.626±.00	.563±.01	.531±.01
	10	.897±.01	.898±.02	.893±.02	.661±.01	.583±.01	.585±.00
	50	.878±.03	.884±.03	.877±.03	.642±.01	.615±.01	**.598±.01**
GPT-5.1	0	.954±.02	.952±.03	.950±.03	.620±.01	.524±.01	.483±.00
	10	.975±.00	.979±.00	.977±.00	.709±.02	.721±.02	.684±.02
	50	.981±.00	.985±.00	**.982±.00**	.762±.01	.778±.01	**.748±.02**
Gemini 2.5 F-L	0	.938±.01	.931±.01	**.929±.01**	.563±.02	.456±.02	.412±.02
	10	.826±.01	.841±.00	.831±.01	.609±.01	.656±.02	.620±.02
	50	.873±.01	.889±.01	.879±.01	.669±.01	.713±.01	**.682±.01**
TF-IDF	10	.036±.05	.122±.11	.052±.06	.050±.07	.101±.09	.046±.08
	50	.169±.09	.114±.08	.090±.07	.264±.02	.279±.03	.226±.03
	100	.379±.02	.194±.02	.206±.02	.295±.01	.388±.01	.332±.01
	1000	.969±.02	.934±.02	.942±.02	.900±.01	.677±.02	.709±.03
	10000	.994±.00	.993±.00	**.993±.00**	.970±.01	.949±.01	.958±.01
	100000	-	-	-	.986±.00	.981±.00	**.983±.00**

‡ Gemini 2.5 **Flash Lite**.

Table 13: Topic Classification Results.

management and multi-step reasoning.

Model	Error Type	RAG				KB				Total
		S	M	L	XL	S	M	L	XL	
Haiku 4.5	Prompt too long	0	0	0	2	0	0	8	22	32
	Tool calls limit exceeded	0	0	0	0	19	15	26	24	84
	Tool max retries exceeded	0	0	0	0	0	0	3	0	3
	Total Errors	**0**	**0**	**0**	**2**	**19**	**15**	**37**	**46**	**119**
Sonnet 4.5	Output validation failed	0	6	9	1	0	11	10	1	38
	Prompt too long	0	0	0	2	0	0	2	36	40
	Request timeout	0	0	0	0	0	0	3	2	5
	Tool calls limit exceeded	0	0	0	0	0	1	3	3	7
	Tool max retries exceeded	0	0	0	0	0	0	1	0	1
	Total Errors	**0**	**6**	**9**	**3**	**0**	**12**	**19**	**42**	**91**
GPT-5 Nano	Empty model response	2	2	1	4	3	13	7	0	32
	Other API errors	0	0	0	6	0	1	0	2	9
	Request timeout	0	0	0	0	0	0	0	1	1
	Server error (5xx)	0	0	0	0	0	0	0	1	1
	Tool calls limit exceeded	0	0	2	4	18	15	26	20	85
	Total Errors	**2**	**2**	**3**	**14**	**21**	**29**	**33**	**24**	**128**
GPT-5.1	Empty model response	0	0	19	26	39	36	69	23	212
	Other API errors	0	0	0	0	0	0	0	3	3
	Rate limit exceeded	0	6	4	0	4	11	0	0	25
	Server error (5xx)	0	0	0	0	0	0	0	1	1
	Tool calls limit exceeded	0	0	0	0	2	1	0	2	5
	Unknown error	0	0	1	0	0	0	0	0	1
	Total Errors	**0**	**6**	**24**	**26**	**45**	**48**	**69**	**29**	**247**
Gemini 2.5 Flash Lite	Invalid API response	0	1	1	0	1	2	3	4	12
	Output validation failed	2	1	2	0	19	23	10	25	82
	Tool calls limit exceeded	0	0	0	0	2	5	5	5	17
	Unknown error	0	0	0	0	0	1	0	0	1
	Total Errors	**2**	**2**	**3**	**0**	**22**	**31**	**18**	**34**	**112**

Table 14: Error analysis across all models.

P KB Evaluation Prompt

The following is the extraction prompt used for the KB Evaluation task.

```
# Business Document Knowledge Extraction System

You are an expert knowledge extraction system.
↪ Your task is to analyze business documents
↪ and extract structured information about
↪ entities (people, organizations, projects,
↪ tasks, meetings) and the relationships
↪ between them.

## Output Requirements

**CRITICAL**: Return ONLY valid JSON with no
↪ explanations, preambles, or markdown
↪ formatting. The response must be parseable
↪ JSON that exactly matches the schema provided.
↪

---

## Step 1: Document Classification

Before extracting any information, carefully
↪ read the entire document and classify it as
↪ ONE of the following types:

- **`agenda`**: An email outlining topics and
↪ agenda for an upcoming meeting
- **`meeting_minute`**: An email summarizing the
↪  discussions and outcomes of a past meeting
- **`regular_email`**: An email message between
↪ employees NOT discussing topics related to
↪ upcoming or past meetings

Store this classification as `document_type` in
↪ your response.

---

## Step 2: Entity Extraction

### Entity Extraction Rules

1. **Extract ONLY entities that participate in
↪ relationships** - no orphaned entities. For
↪ instance, if the relationships are "Employee
↪ worksOn Task" and "Employee worksAt Company"
↪ then ONLY extract entities Employee, Task,
↪ and Company, nothing else. Do the same for
↪ each relationship.
2. Extract entities exactly as they appear in
↪ the document (preserve exact wording for
↪ aliases)
3. For each entity, capture ALL variations/
↪ aliases found in the document (e.g., ["John
↪ Smith", "J. Smith", "John", "John S.", "Jonny
↪ "] etc.)
4. Entity aliases MUST be extracted exactly as
↪ they appear in the document (do not make up
↪ aliases)
5. Email addresses are CRITICAL for entity
↪ linking - extract when available
6. Entity types must be EXACTLY as specified
↪ below (case-sensitive)
7. If entity type is Employee, extract the job `
↪ title`, `department`, and `team` if mentioned
↪  in the email body or in the email signature

### Allowed Entity Types by Document Type

**If `document_type` is `regular_email` then
↪ allowed entity types are:**
- Employee
- Company
- Department
- Team
- Task
- Project

**If `document_type` is `agenda` or `
↪ meeting_minute` then allowed entity types are
↪  all of the above plus:**
- Meeting

### Entity Type Definitions & Schemas

#### Employee
Individual people working at the company.

```json
{
  "type": "Employee",
  "name": "Full Name",
  "email": "email@company.com",
  "aliases": ["Full Name", "First Last", "
↪ Nickname", "F. Last"],
  "title": "Job Title",
  "department": "Department Name",
  "team": "Team Name"
}
```

Company
The organization itself or partner organizations.
↪

```json
{
  "type": "Company",
  "name": "Company Name",
  "aliases": ["Company Name", "CompanyName Inc",
↪  "CN"]
}
```

Department
Organizational divisions (e.g., Flight
↪ Operations, Ground Services, Maintenance,
↪ Customer Service).

```json
{
  "type": "Department",
  "name": "Department Name",
  "aliases": ["Department Name", "Dept Name", "
↪ DEPT"]
}
```

Team
Working groups within departments.

```json
{
```
```

```
  "type": "Team",
  "name": "Team Name",
  "aliases": ["Team Name", "Team", "Group Name"],
↪
  "department": "Parent Department Name"
}
```

Task
Specific work items and assignments.

```json
{
  "type": "Task",
  "name": "Task Name",
  "aliases": ["Task Name", "Task shorthand"],
  "description": "Task description",
  "status": "active OR completed",
  "assigned_to": ["Employee Name 1", "Employee
↪ Name 2"],
  "project": "Associated Project Name"
}
```

Project
Larger initiatives that encompass multiple tasks.
↪

```json
{
  "type": "Project",
  "name": "Project Name",
  "aliases": ["Project Name", "Project Nickname",
↪  "Initiative Name"],
  "description": "Project description"
}
```

Meeting
Scheduled gatherings of employees (only for `
↪ agenda` or `meeting_minute` documents).

```json
{
  "type": "Meeting",
  "title": "Title of Meeting",
  "aliases": ["Meeting Title", "Alternative
↪ Title"],
  "date": "YYYY-MM-DD",
  "document_type": "agenda OR meeting_minute",
  "meeting_type": "direct_report OR team OR task
↪  OR executive OR project OR department"
}
```

Meeting Type Classification:
If the entity is of type `Meeting`, classify its
↪ `meeting_type` as ONE of the following based
↪ on the document content:
- `direct_report`: Meeting between an employee
↪ and their direct manager
- `team`: Meeting among members of the same team
- `task`: Meeting focused on a specific task
- `executive`: Meeting involving executives or
↪ high-level management
- `project`: Meeting focused on a specific
↪ project
- `department`: Meeting involving members of the
↪ same department

Step 3: Relationship Extraction

Relationship Extraction Rules

1. Extract ONE principal relationship (the most
↪ important/central relationship in the
↪ document)
{% block instructions_relationship %}
2. Extract multiple secondary relationships
↪ according to the criteria below
{% endblock %}
3. Relationship types must be EXACTLY as
↪ specified (case-sensitive)
4. **Temporality is REQUIRED** - must be one of:
↪ `start`, `end`, or `ongoing` (never null)
5. **Date Rules:**
   - If temporality is `start`: `beginning` must
↪ contain a valid date (YYYY-MM-DD), `end` is
↪ empty string
   - If temporality is `end`: `end` must contain
↪ a valid date (YYYY-MM-DD), `beginning` is
↪ empty string
   - If temporality is `ongoing`: both `
↪ beginning` and `end` are empty strings
6. Dates must be derived from the document - DO
↪ NOT fabricate dates
7. Extract `mentions`: direct quotes or
↪ paraphrases from the document that evidence
↪ this relationship

Allowed Relationship Types by Document Type

**If the `document_type` is `regular_email` then
↪ the allowed relationship types are:**
- worksOn
- worksAt
- reportsTo
- memberOf
- departmentBelongsToCompany (secondary only)
- teamBelongsToDepartment (secondary only)

**If the `document_type` is `agenda` or `
↪ meeting_minute` then the allowed relationship
↪ types are all of the above plus:**
- organize
- attend
- mentionsTask
- mentionsProject
- mentionsEmployee
- mentionsTeam
- mentionsDepartment

Relationship Type Definitions & Extraction
↪ Criteria

worksOn
Pattern: `Employee worksOn Task`

Extract when:
- The document explicitly states an employee is
↪ working on, assigned to, or responsible for a
↪ specific task
- Extract ONE relationship per task mentioned

Example mentions:
- "Sarah is working on the fleet maintenance
↪ audit"

```
- "John has been assigned the gate scheduling
↪ optimization task"

---

#### worksAt
**Pattern**: `Employee worksAt Company`

**Extract when:**
- The document mentions an employee works at a
↪ company
- The email sender uses a work email -> extract
↪ `sender worksAt company`
- The email recipient uses a work email ->
↪ extract `recipient worksAt company`

**Example mentions:**
- "As a flight coordinator at SkyWings Airlines
↪ ..."
- Email from: john.smith@skywings.com -> John
↪ Smith worksAt SkyWings Airlines

---

#### reportsTo
**Pattern**: `Employee reportsTo Employee (
↪ manager)`

**Extract when:**
- The document explicitly mentions reporting
↪ structure
- Manager-employee language is used (e.g., "your
↪  progress on...", "I'd like you to...", "
↪ report back to me")

**Example mentions:**
- "As your manager, I wanted to check in on the
↪ crew scheduling..."
- "Please report your findings to Sarah"

---

#### memberOf
**Pattern**: `Employee memberOf Team` OR `
↪ Employee memberOf Department`

**Extract when:**
- The document signature contains the employee's
↪  team or department
- Collective language indicates team membership
↪ (e.g., "our team's progress", "we in the
↪ Ground Operations team")
- Explicit membership statements

**Example mentions:**
- Email signature: "John Smith | Ground
↪ Operations Team"
- "Our team has been coordinating baggage
↪ handling improvements..."

---

#### departmentBelongsToCompany
**Pattern**: `Department
↪ departmentBelongsToCompany Company`

**SECONDARY RELATIONSHIP ONLY** - never the
↪ principal relationship

**Extract when:**
- You have a principal relationship `Employee
↪ memberOf Department` AND the sender uses a
↪ work email
- Document signature contains department AND
↪ company information

**Example mentions:**
- Email signature: "Flight Operations Department
↪  | SkyWings Airlines"

---

#### teamBelongsToDepartment
**Pattern**: `Team teamBelongsToDepartment
↪ Department`

**SECONDARY RELATIONSHIP ONLY** - never the
↪ principal relationship

**Extract when:**
- The document explicitly states a team is
↪ within/part of a department

**Example mentions:**
- "The Crew Scheduling team within our Flight
↪ Operations Department..."

---

#### organize
**Pattern**: `Employee organize Meeting`

**Extract when:**
- Document type is `agenda` or `meeting_minute`
- The document identifies a meeting organizer/
↪ host

**Example mentions:**
- "Organized by: Sarah Johnson"
- "Meeting host: John Smith"
- "From: Sarah Johnson"

---

#### attend
**Pattern**: `Employee attend Meeting`

**Extract when:**
- Document type is `agenda` or `meeting_minute`
- Extract ONE relationship for EACH attendee
↪ listed

**Example mentions:**
- "Attendees: Sarah Johnson, Mike Chen, Emily
↪ Rodriguez"
- "To: Mike Chen, Emily Rodriguez"

---

#### mentionsTask
**Pattern**: `Meeting mentionsTask Task`

**Extract when:**
- Document type is `agenda` or `meeting_minute`
- A task is discussed or referenced in the
↪ meeting

**Example mentions:**
- "Agenda item 2: fleet maintenance audit task"
```

---

mentionsProject
Pattern: `Meeting mentionsProject Project`

Extract when:
- Document type is `agenda` or `meeting_minute`
- A project is discussed or referenced in the meeting

Example mentions:
- "Discussion of Fleet Modernization project timeline"

---

mentionsEmployee
Pattern: `Meeting mentionsEmployee Employee`

Extract when:
- Document type is `agenda` or `meeting_minute`
- An employee is mentioned in the meeting (who may not be an attendee)

Example mentions:
- "Sarah's progress on the safety audit will be reviewed"

---

mentionsTeam
Pattern: `Meeting mentionsTeam Team`

Extract when:
- Document type is `agenda` or `meeting_minute`
- A team is discussed or referenced in the meeting

Example mentions:
- "Ground Operations Team's efficiency metrics"

---

mentionsDepartment
Pattern: `Meeting mentionsDepartment Department`

Extract when:
- Document type is `agenda` or `meeting_minute`
- A department is discussed or referenced in the meeting

Example mentions:
- "Flight Operations Department's Q4 performance review"

Relationship Schema

```json
{
  "type": "EXACT_TYPE_FROM_ABOVE",
  "source": "Source Entity Name",
  "target": "Target Entity Name",
  "context": "1-2 sentence context explaining this relationship from the document",
  "mentions": ["Direct quote 1 from document", "Direct quote 2 from document"],
  "temporality": "start OR end OR ongoing",
  "beginning": "YYYY-MM-DD or empty string",
  "end": "YYYY-MM-DD or empty string"
}
```

Complete Output Schema

```json
{
  "document_type": "regular_email OR agenda OR meeting_minute",
  "entities": [
    {
      "type": "Entity Type",
      "name": "Entity Name",
      "aliases": ["alias1", "alias2"],
      ...additional fields based on entity type ...
    }
  ],
  "relationships": [
    {
      "type": "Relationship Type",
      "source": "Source Entity Name",
      "target": "Target Entity Name",
      "context": "Context from document",
      "mentions": ["mention1", "mention2"],
      "temporality": "start OR end OR ongoing",
      "beginning": "YYYY-MM-DD or empty string",
      "end": "YYYY-MM-DD or empty string"
    }
  ]
}
```

Extraction Checklist

Before finalizing your response, verify:

- [ ] Document type is correctly identified
- [ ] Exactly ONE principal relationship is extracted
- [ ] One or many secondary relationships are extracted as applicable
- [ ] Extract a Meeting entity for `agenda` or `meeting_minute` documents
- [ ] All extracted entities participate in at least one relationship
- [ ] All extracted relationships have a source and target entity present in the entities list
- [ ] All relationship types are spelled EXACTLY as specified
- [ ] All entity types are spelled EXACTLY as specified
- [ ] Every relationship has a valid temporality value (never null)
- [ ] Date fields follow rules (populated for start/end, empty for ongoing)
- [ ] All dates are in YYYY-MM-DD format
- [ ] All dates are extracted from the document (not fabricated)
- [ ] Mentions contain actual quotes/paraphrases from the document
- [ ] Aliases capture all variations found in the document

```
- [ ] Email addresses are extracted when
↪ available
- [ ] Title must be always extracted for
↪ Employee entities
- [ ] Output is valid JSON with no explanations
↪ or markdown

---

## EXAMPLES:

{{ few_shot_examples }}

---

## Your Task

Analyze the document below and return ONLY the
↪ JSON output following the schema. No
↪ explanations, no preambles, no markdown code
↪ blocks - just the raw JSON.

## DOCUMENT TO ANALYZE:
Filename: {{ document_filename }}
Content:
---
{{ document_content }}
---

RESPONSE:
```

Q Topic Classification Prompt

The following is the extraction prompt used for the Topic Classification task.

```
You are an expert topic classification system.

INSTRUCTIONS:
1. Return ONLY one of the provided labels, do
↪ not generate extra unnecessary text.
2. Do NOT generate an empty text as an answer,
↪ you ALWAYS MUST return one of the provided
↪ labels

# Few-shot examples:

## DOCUMENT TO ANALYZE:

Message-ID: d46bb20a-ef29-4936-bdc1-52cf20c9316a
From: Robert Lederer <
↪ Rlederer@poundfinancialgroupllc.com>
To: Nath Miller <miller.nath@gmail.com>
Date: 2024-03-01
Subject: Strengthening our checks and balances
MIME-Version: 1.0
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 7bit

Hey Nath, I wanted to touch base on something
↪ that's been on my radar lately. As we
↪ continue to focus on our broader business
↪ operations and how we're creating value
↪ across our portfolio, I think there's a real
↪ opportunity for us to tighten up our
↪ governance oversight. On that note, I'm newly
↪  working on subordination lien release
↪ verification, and I'm seeing firsthand how
↪ critical it is that we have solid
↪ verification processes in place.

You know, when you zoom out and look at
↪ portfolio management, a lot of it comes down
↪ to having strong accountability mechanisms
↪ that actually work. Right now, I'm thinking
↪ through some ways we could improve how we
↪ handle these verification workflows to make
↪ sure nothing slips through the cracks. It's
↪ not just about crossing boxes---it's about
↪ building confidence in our processes.

I think the key is making sure we've got clear
↪ ownership and tracking at every step. The
↪ current approach is solid, but there's
↪ definitely room to strengthen our controls
↪ and make the handoffs smoother between teams.
↪  Have you noticed any pain points on your end
↪  that we could address?

Would love to grab coffee or hop on a quick call
↪  this week to brainstorm some ideas. I'm
↪ pretty fired up about leveling up our
↪ accountability system, and I think your
↪ perspective would be really valuable here.
↪ Let me know what works for your schedule.

Thank you,
Robert

Robert Lederer
Analyst, Pound Financial Group LLC

P: +1 (415) 660-5966
E: Rlederer@poundfinancialgroupllc.com

## LABELS:

- Competitive Analysis
- Exit Planning
- Financial Analysis
- Legal Compliance
- Market Intelligence
- Market Research
- Network Development
- Non-business
- Operational Assessment
- Operational Improvement
- Opportunity Screening
- Portfolio Management
- Relationship Building
- Stakeholder Management
- Strategic Planning
- Technology Assessment
- Transaction Structuring

## RESPONSE:

Portfolio Management

....

# Your task:
Classify the following document using one of the
↪  labels provided (DO NOT REASON):
```

Biocure (L)	Pound (XL)
Non-business	Non-business
Approval Timeline Management	Exit Planning
Biomarker Identification	Financial Analysis
Clinical Data Analysis	Legal Compliance
Clinical Trial Planning	Market Intelligence
Competitive Intelligence	Market Research
Distribution Channel Management	Network Development
Efficacy Evaluation	Competitive Analysis
FDA Communication	Operational Assessment
Formulation Optimization	Operational Improvement
Healthcare Provider Education	Opportunity Screening
Intellectual Property Strategy	Portfolio Management
International Regulatory Coordination	Relationship Building
Key Opinion Leader Engagement	Stakeholder Management
Labeling Requirements	Strategic Planning
Manufacturing Compliance	Technology Assessment
Manufacturing Process Development	Transaction Structuring
Market Access Strategy	
Advisory Committee Preparation	
Partnership Collaborations	
Patient Assistance Programs	
Post Marketing Commitments	
Preclinical Research	
Pricing Negotiations	
Promotional Campaign Development	
Regulatory Strategy	
Regulatory Submission Preparation	
Safety Assessment	
Safety Reporting	
Sales Force Training	
Sales Performance Analytics	

Table 15: Topic Categories for Biocure (L) and Pound (XL).

```
## DOCUMENT TO ANALYZE:

{{ document_content }}

## LABELS:

{{ topics }}

## RESPONSE:
```

R Topic Categories

Table 15 shows the topic categories used for the topic classification task for Biocure (L) and Pound (XL) datasets.