Contextualized Financial Intelligence: A Retrieval-Augmented Generation System for Company Financial Question Answering

Dipto Paul, Harish Gururaj Kotre, Poorna Chandra Singh
Team Name: Oneironauts
M.Sc. International Software System Science, University of Bamberg, Germany
Supervised by: Sabine Weber, Lynn Greschner

Abstract

We introduce a Retrieval-Augmented Generation (RAG) system designed to answer complex financial questions grounded in SEC filings such as Form 10-K and 10-Q reports. Our system combines content-aware chunking, hybrid semantic and lexical retrieval, reranking with CrossEncoders, and task-specific prompting using LLaMA3-70B. We evaluate on 134 questions across fact, comparative, reasoning, summarization, and hallucination types. Our system shows significant improvement over a Flan-T5 baseline, achieving an average semantic F1 score of 0.3415. We explain architectural decisions, challenges, and analysis, and propose future work in numerical reasoning and table-aware models.

1 Introduction

Understanding financial documents is critical for investors, analysts, and auditors. Documents like 10-K and 10-Q contain dense tabular and textual data. Large Language Models (LLMs) like GPT or LLaMA often hallucinate or return outdated responses. Retrieval-Augmented Generation (RAG) solves this by retrieving up-to-date, relevant document chunks and conditioning generation on them.

Our work builds a full RAG pipeline customized for financial filings. We show how careful chunking, hybrid retrieval, and tailored generation improves factual accuracy. We also design a test set with 134 annotated questions to evaluate performance across difficulty and types.

2 System Architecture

Our architecture, as shown in fig 1, follows the RAG paradigm, with improvements in chunking, retrieval, and generation to handle financial domain complexity.

2.1 Document Chunking

SEC documents often mix paragraphs, tables, and headers. Naive fixed-size chunking breaks logical blocks, especially tables. We use **Docling** for document-aware chunking, preserving table structure and section boundaries.

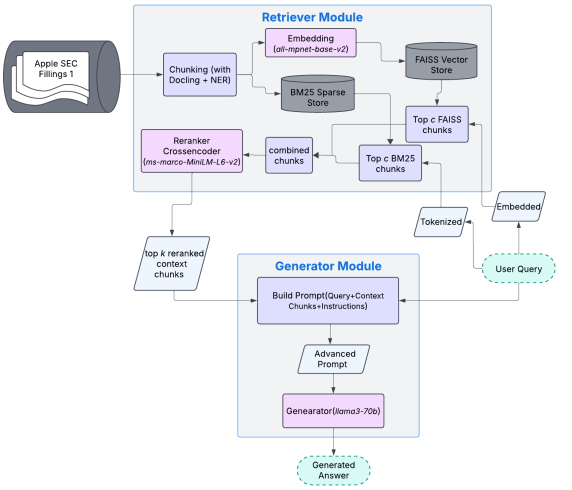

Figure 1: System architecture showing the flow from SEC document chunking to hybrid retrieval and final generation. The retriever uses hybrid approach to select top *k* context, which is passed into task-specific prompts for grounded answer generation.

Chunks are also tagged using Named Entity Recognition (NER) to mark organizations (like "Apple Inc."). This helps filter and prioritize relevant content during retrieval.

2.2 Retriever

Retrieval is a critical component in any Retrieval-Augmented Generation (RAG) pipeline. Our system implements a **hybrid retrieval mechanism** that combines both semantic and lexical approaches to balance robustness and precision. This is particularly important in the financial domain, where both numerical precision and linguistic variation are common.

Pure dense retrieval (semantic similarity) may fail when exact terms are critical (e.g., specific numbers or acronyms like “EPS”), while sparse retrieval (e.g., BM25) may miss semantically equivalent phrasings. Our hybrid system leverages the strengths of both:

- **Dense retrieval (FAISS)** captures the intent and meaning of a question.
- **Sparse retrieval (BM25)** captures exact keyword matches that often exist in financial tables or notes.

2.2.1 Dense Retrieval with FAISS

We use FAISS for dense retrieval, embedding chunks and queries into 768-dimensional vectors using `all-mpnet-base-v2` from SentenceTransformers. This model supports slightly longer context (512 tokens) compared to MiniLM (256) and produces consistent semantic embeddings. Chunks are indexed using `IndexFlatIP`, a fast inner-product similarity index.

During retrieval, a query is embedded and matched against the index to retrieve the top-c semantically similar chunks.

2.2.2 Sparse Retrieval with BM25

BM25 ranks chunks using lexical overlap. We tokenize and index all document chunks using a classic BM25 implementation. This method excels at matching exact phrases and numerics such as “Q2 2024” or “$36.3 billion”.

2.2.3 Combining FAISS and BM25 with Reranking

Both FAISS and BM25 return c candidates. We merge and deduplicate their outputs to produce up to $2c$ unique candidates per query. These are passed to a reranker (`cross-encoder/ms-marco-MiniLM-L-6-v2`), which scores each query-chunk pair using cross-encoding. This step enables:

- **High recall**: BM25 can recover missed FAISS matches.
- **Semantic precision**: Reranker chooses contextually appropriate passages.

2.2.4 Retrieval Challenges, Design Motivations, and Performance Impact

During system development, we encountered several retrieval-specific challenges, particularly due to the structure and content of financial documents. These documents often exhibit structural redundancy, dense numeric content, and inconsistent terminology, which posed issues for both dense and sparse retrieval methods.

Dense retrieval using FAISS, while effective for semantically phrased questions, often struggled with numeric sensitivity. Embeddings failed to distinguish between values such as "$36.3B" vs. "$36.2B", leading to imprecise matches. Additionally, FAISS frequently returned structurally similar or even duplicate chunks due to the repetitive nature of financial reports (e.g., repeated mentions of revenue breakdowns across quarters).

In contrast, sparse retrieval using BM25 was better at pinpointing exact matches, especially for numeric data and keywords like “Q1 2025” or “EPS.” However, it lacked semantic flexibility. BM25 frequently retrieved off-topic passages that shared lexical overlap with the query, what we call “keyword drift”, where common terms appeared across unrelated sections.

These complementary failure modes motivated our move toward a **hybrid retrieval strategy**. We retrieve the top-c chunks independently from FAISS and BM25, merge and deduplicate them, and rerank the resulting $2c$ candidates using a cross-encoder. This allows us to:

- Capture the intent behind semantically complex questions (via FAISS).
- Preserve exact numeric and keyword matching (via BM25).
- Increase chunk diversity and reduce reliance on any single retrieval method.

Ablation studies confirmed the need for this approach: FAISS alone missed important factual content, while BM25 alone failed on reasoning-oriented or paraphrased queries. The hybrid setup mitigated both issues, enhanced performance, and ultimately better generation.

In terms of impact, we observed:

- Higher chunk relevance, leading to improved grounding and reduced hallucination in generated answers.
- Gains in semantic F1, as more relevant information was surfaced during retrieval.
- Improved reranker precision due to a more informative and balanced candidate pool.

Ultimately, our design choices were guided by empirical challenges and systematically addressed through hybrid retrieval and reranking, forming a more reliable retrieval backbone for financial QA tasks.

2.3 Answer Generation and Task-Specific Prompting

The final step in our pipeline is generating an answer conditioned on the top-ranked chunks. We use the **LLaMA3-70B** language model, which supports long context windows (up to 8192 tokens), allowing us to input multiple retrieved passages along with a structured prompt.

To guide the model more effectively, we designed a set of **task-specific prompts** tailored to the nature of each question. This approach helps prevent misinterpretation of the task and reduces hallucinations, especially when the provided context is ambiguous or limited.

For each question, we classify its type by detecting specific keywords or patterns within the question text using regular expressions. Based on this classification, we dynamically select the most appropriate prompt style. Common prompt types include:

- **Fact-based questions:** Require direct factual answers, with prompts presenting the question followed by a space for the answer.

- **Multiple choice questions:** Instruct the model to select the best option (e.g., ‘a’, ‘b’, etc.) based on the retrieved context.

- **Summarization questions:** Prompts ask for a concise summary of the context relevant to the question’s intent.

- **Yes/No questions:** Require a short answer of “Yes,” “No,” or “I don’t know,” accompanied by brief justification using the context.

This prompt strategy proved to be an effective control mechanism. For instance, comparative prompts helped avoid incomplete answers such as “X is higher” without mentioning the compared value, while hallucination-safe prompts explicitly instructed the model not to guess if evidence was absent.

In our evaluations, we observed that task-aware prompting improved both factual precision and clarity of answers. It was especially beneficial for borderline cases where retrieved context was partial or ambiguous.

3 Test Set and Evaluation

3.1 Dataset Overview

We evaluated on 134 curated questions from actual SEC filings (Apple and Amazon). Each question was annotated with key attributes relevant for evaluation and analysis:

- **Question Type**: fact (29), summarization (21), comparative (20), yes/no (38), MCQ (19), hallucination (3), reasoning (2), calculation (2)

- **Difficulty**: easy (37), medium (67), hard (30)
  - **easy**: Direct answers present in one chunk.
  - **medium**: Requires paraphrase understanding or multiple chunk context.
  - **hard**: Needs multiple retrievals, reasoning, or math (e.g., percentage change).

- **Source**: text or table

Additional metadata (e.g., gold answers, page numbers, document IDs) support evaluation, with emphasis on the above attributes.

3.2 Evaluation Metrics

To evaluate the effectiveness of our RAG system for question answering over financial SEC filings, we employ both surface-level and semantic metrics to assess factual accuracy, fluency, and meaning preservation.

Semantic F1 Score. Captures overlap in meaning between gold and predicted answers, using sentence embeddings. This is crucial in financial QA, where paraphrasing and rewording are common.

Cosine Similarity. Measures the similarity between sentence embeddings. This helps assess semantic consistency when the wording differs but the financial content aligns.

ROUGE-L. Evaluates lexical overlap via the Longest Common Subsequence. It is useful for capturing sequence-sensitive factual alignment, especially in numeric or regulatory content.

BERTScore (F1). Uses contextual embeddings to compare token-level similarity. Effective in cases where answers are semantically equivalent but lexically diverse.

Precision, Recall, and F1 Score. Token-level metrics to evaluate how much of the gold answer is recovered and how accurate the prediction is:

- **Precision:** Tokens in the prediction that are correct.
- **Recall:** Tokens in the gold answer correctly predicted.
- **F1 Score:** Harmonic mean of precision and recall:

$$\mathrm{F1} = 2 \cdot \frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

These metrics collectively balance semantic and factual evaluation, both critical in the financial domain, where slight deviations can alter interpretations.

3.3 Results

Table 1 compares the overall Semantic F1 Scores of our hybrid model (Hybrid + LLaMA3) against the baseline (FAISS + Flan-T5), showing a clear improvement (0.34 vs. 0.12).

Table 2 (token-level metrics for the hybrid model) reveals that recall (0.60) is substantially higher than precision (0.25), indicating the model retrieves much relevant information but also produces some irrelevant content. The average F1 score is moderate (0.27), and BERTScore (0.60) confirms strong semantic similarity between predictions and references.

Table 3 breaks down token-level metrics by question type. Fact-based questions achieve the highest scores across all metrics (F1: 0.474, BERTScore F1: 0.736), reflecting easier retrieval and generation. In contrast, challenging types like reasoning, calculation, and hallucination show lower precision and F1, suggesting these require further model improvements. For source types, questions grounded in free text achieved higher performance (F1: 0.265, BERTScore F1: 0.593) compared to those from tables (F1: 0.241, BERTScore F1: 0.577), likely due to the structural complexity and sparsity of tabular data.

Overall, the results demonstrate the hybrid system's strengths in handling diverse financial questions while highlighting areas needing refinement.

Model	Semantic F1 Score	Questions
FAISS + Flan-T5 (baseline)	0.1206	40
Hybrid + LLaMA3 (ours)	0.3415	134

Table 1: baseline vs ours improved version

Metric	**Score**
Average Precision	0.25
Average Recall	0.60
Average F1 Score	0.27
Average ROUGE-L	0.29
Average BERTScore F1	0.60

Table 2: Overall token-level evaluation metrics for Improved version

3.4 Strengths and Errors

Our improved system demonstrates strong performance on fact-based questions, such as those asking for net income or the number of employees, where it reliably generates accurate answers. Additionally, many Yes/No and multiple-choice questions are answered correctly with detailed explanations. However, this strength is not always reflected in evaluation metrics, which tend to score these answers lower due to length and phrasing differences compared to the often brief gold standard answers.

For example, consider the question:

> *Does Apple list its common stock on the NASDAQ?*

The gold answer is simply **"Yes"**, whereas the generated answer is longer and provides supporting context:

> *Yes. According to the context, "The Company's Common Stock is listed on The Nasdaq Stock Market LLC under the trading symbol 'AAPL'."*

Despite being correct, this leads to a low F1 score (0.105), illustrating a mismatch between metric scoring and actual answer quality.

Additionally, common errors include confusion caused by retrieval of duplicate or near-identical text chunks, which can mislead the generation process. Sometimes, the retrieval step fails to capture

Category	**Precision**	**Recall**	**F1**	**ROUGE-L**	**BERTScore F1**
Question Type					
Fact-based	0.488	0.764	**0.474**	0.523	**0.736**
Summarization	0.221	0.524	0.289	0.239	0.652
Comparative	0.227	0.360	0.242	0.249	0.622
Yes/No	0.044	0.754	0.078	0.073	0.409
MCQ	0.303	0.430	0.280	0.390	0.615
Calculation	0.040	0.333	0.071	0.072	0.434
Reasoning	0.205	0.667	0.314	0.334	0.724
Hallucination	0.184	0.778	0.295	0.211	0.617
Average (Q-Type)	0.214	0.576	**0.255**	0.262	0.601
Source Type					
Text	0.227	0.645	**0.265**	0.281	**0.593**
Table	0.264	0.515	0.241	0.265	0.577

Table 3: Token-level evaluation metrics grouped by Question Type and Source Type.

the exact ground truth chunk due to subtle wording mismatches, impacting answer accuracy. Furthermore, the language model struggles with precise numeric comparisons, such as determining which value is higher.

While the system clearly has the capability to generate correct answers across various question types, these limitations highlight areas where retrieval quality and evaluation alignment could be improved.

4 System Comparison

We implemented two versions of our RAG pipeline, a baseline and an improved system, differing across chunking, retrieval, reranking, and generation components. The base version used fixed-length chunking, MiniLM-based dense retrieval with FAISS, no reranking, and Flan-T5 with generic prompts. In contrast, the improved system introduced layout-aware chunking via Docling, hybrid retrieval using both FAISS (mpnet embeddings) and BM25, CrossEncoder-based reranking, and task-specific prompting with LLaMA3-70B. These upgrades collectively enhanced retrieval precision, contextual grounding, and answer accuracy, particularly for numerical and comparative questions.

Component	**Base**	**Improved**
Chunking	Fixed-length	Layout-aware (Docling)
Retriever	MiniLM + FAISS	mpnet + FAISS + BM25
Reranker	None	CrossEncoder
Generator	Flan-T5	LLaMA3-70B
Prompting	Generic	Task-specific

Table 4: Comparison of key modules between the base and improved system.

5 Conclusion

We presented a specialized Retrieval-Augmented Generation (RAG) pipeline for question answering over financial documents, focusing on SEC filings such as 10-K and 10-Q reports. Our system integrates several domain-aware components: structurally coherent chunking via Docling, a hybrid retriever combining FAISS and BM25 for semantic and lexical matching, a CrossEncoder reranker for final relevance filtering, and task-specific prompts for more precise and grounded generation using `LLaMA3-70B`.

Several key findings emerged. Hybrid retrieval outperformed single-method alternatives by mitigating issues like semantic drift and numeric insensitivity. CrossEncoder reranking further improved precision, especially when top candidates from FAISS or BM25 were noisy or redundant. Task-specific prompts reduced hallucinations and enhanced factuality, particularly for structured queries. Chunking with Docling preserved the structural integrity of financial documents, allowing the model to better align table headers with corresponding data rows, though subheaders like units were occasionally missed.

Through extensive evaluation on a diverse, handcrafted dataset of 134 questions, we show our system consistently outperformed a strong Flan-T5 baseline, particularly on fact-based questions, while remaining competitive on summarization and comparison tasks.

Despite these gains, challenges remain. The system still struggles with complex tabular reasoning such as interpreting percentage changes or inferring relationships across rows. Docling improves table chunking by attaching headers to their related content, but some subheaders (e.g., units like "in millions" or "per share" or "$") are occasionally omitted, leading to misinterpretations. Retrieval occasionally fails when relevant context is split across chunks or diverges linguistically from the query. Chunking errors also persist when tables span multiple pages or when structural cues are weak.

Future improvements include integrating table-focused models like TAPAS or TABBIE to handle structured data more effectively, leveraging reranker feedback to weight chunks dynamically, and incorporating confidence scoring to handle uncertainty. Expanding retrieval beyond regulatory filings to include press releases or news may also enrich context, particularly for time-sensitive questions. Enhancements in prompt design, chunking fidelity, and evaluation metrics tailored to financial reasoning could further improve both answer quality and interpretability.

Overall, this project demonstrates the value of domain-specialized RAG systems in complex, high-stakes applications such as finance. By designing for the constraints and formats of the financial domain, we can significantly improve the utility of LLM-based QA systems. We believe this work serves as a strong foundation for future efforts in grounded question answering over financial, legal, and technical documents.

Contributions

Dipto Paul: System design, retrieval code design, generator code design, chunking(Docling), reranking(Cross Encoder), prompt design, test set preparation, sec fillings collection, evaluation, presentation slide preparation, report writing.
Harish Gururaj: Hybrid retrieval(FAISS+BM25), test set preparation, retrieval analysis, report writing.
Poorna Chandra Singh: report writing, presentation slide preparation, and providing collaborative support.

GitHub Repository

https://github.com/dpauldac/NLProc-Proj-M-SS25