International Journal of Research in Engineering, Science and Management
Volume 8, Issue 11, November 2025
www.ijresm.com | E-ISSN: 2581-5792 | RESAIM Publishing (www.resaim.com)

Multimodal Retrieval-Augmented Generation for Financial Report Question-Answering: Architecture and Evaluation

Pratham Shelke[1], Omkar Yellaram[2*], Sunil Ghane[3]
[1,2]Student, Department of Computer Science and Engineering, Sardar Patel Institute of Technology, Mumbai, India
[3]Professor, Department of Computer Engineering, Sardar Patel Institute of Technology, Mumbai, India

***Abstract*: The exponential growth of digital financial reports-rich in text, tables, and graphical data-poses significant challenges for efficient information extraction and analysis. This paper presents a multimodal Large Language Model (LLM)-powered Question Answering (QnA) system for financial documents, focusing on a Retrieval-Augmented Generation (RAG) pipeline. Our system integrates advanced natural language processing, vision-language models, and optimized chunking strategies to retrieve and synthesize information from complex financial filings. Comprehensive evaluation across multiple configurations demonstrates the pipeline's robustness, highlighting the impact of hyperparameter tuning, chunking methods, and embedding strategies on answer faithfulness, relevance, and factual correctness.**

***Keywords*: Multimodal RAG, Financial document question answering, Large language models, Vision-Language models, PDF parsing, Table extraction, Graph data extraction, Retrieval-augmented generation pipeline, Semantic embeddings, Chunking strategies, Vector database, Similarity search, RAG evaluation, Faithfulness, Answer relevance, Context precision, Context recall, Semantic similarity.**

1. Introduction

The financial sector is undergoing a digital shift, with organizations depending on PDF-based financial reports that integrate text, structured tables, and visual elements like charts and diagrams. These multimodal documents are useful for stakeholders-including analysts, investors, and auditors-since they contain crucial data for decision-making and regulatory compliance. Yet, the extraction of actionable insights from such heterogeneous sources remains a formidable challenge. Manual review is not only resource-intensive and slow but also susceptible to errors, especially as report volumes and complexity escalate.

Automated question answering (QnA) over financial documents presents unique obstacles. First, the fusion of narrative explanations, numerical tables, and graphical summaries within a single report demands a system capable of processing multiple modalities. Standard text-based retrieval techniques often fall short, as they are unable to seamlessly integrate insights from text, tables, and images. However, financial documents contain intricate structures, cross-references, and domain-specific language, complicating the process of accurately locating and synthesizing answers.

As Large Language Models (LLMs) have made significant advances in natural language understanding, they are limited when operating without targeted context. Providing an entire lengthy financial document as input is computationally inefficient and may still result in incomplete or inaccurate answers. This is especially problematic in finance, where factual accuracy and traceability to the source are paramount.

To address these issues, our project employs a Retrieval-Augmented Generation (RAG) pipeline tailored for multimodal financial data. The system leverages vision-language models to extract and standardize content from text, tables, and images, converting them into a unified format suitable for downstream processing. By segmenting the data into logical chunks and indexing them using semantic embeddings, the system can efficiently retrieve the most relevant information in response to user queries. The LLM then synthesizes a coherent, context-aware answer grounded in the retrieved evidence.

A critical aspect of this work is the systematic evaluation of the RAG pipeline. Traditional benchmarks do not fully capture the requirements of financial QnA, such as the need for high faithfulness, relevance, and factual correctness. Our evaluation framework incorporates domain-specific metrics and explores the impact of various pipeline configurations, including chunking strategies and embedding methods, on answer quality. Through this approach, we aim to demonstrate both the effectiveness and the adaptability of our solution for complex, real-world financial analysis tasks.

2. Related Work

The intersection of financial document analysis and artificial intelligence has witnessed rapid progress in recent years, particularly with the emergence of Retrieval-Augmented Generation (RAG) systems and multimodal large language models (LLMs). This section reviews foundational and recent advances in the field, highlighting how these developments directly inform and enhance the present research on multimodal

*Corresponding author: omkar.yellaram@spit.ac.in

RAG pipelines for financial question-answering.

[1] Nie et al. (2024) conducted a survey on the application of LLMs in finance, highlighting their potential for tasks such as sentiment analysis, compliance, and financial reasoning. They noted that while LLMs can enhance p efficiency and contextual understanding, issues like data privacy, interpretability, and factual reliability persist, especially when models operate without access to external sourcesThis drawback emphasizes the need for retrieval-augmented methods, in which models are based on current and contextually relevant data.

[2] Emphasizing the shift from general-purpose to domain-specific models, Lee et al. (2024) investigated the evolution of financial language models (FinLLMs). Their analysis revealed that FinLLMs outperform generic LLMs in tasks such as classification and summarization within financial contexts. However, they also observed that these models require large, high-quality datasets and expert-driven evaluation to achieve robust performance, particularly when handling the nuanced language and regulatory requirements of financial documents.

[3] The challenge of handling multimodal data-text, tables, and images-has been addressed through specialized RAG pipelines. Yepes et al. (2024) introduced a chunking strategy that leverages the structural elements of financial documents to improve retrieval and answer accuracy. Their element-based chunking approach, tested on the FinanceBench dataset, demonstrated near state-of-the-art performance while reducing indexing costs. This work established the importance of aligning chunking methods with document structure to enhance the relevance of retrieved content.

[5] Shah et al. (2024) focused on multi-document financial question answering, identifying a gap in existing benchmarks that primarily target single-document tasks. They proposed new strategies, such as RAG_SEM and KG_RAG, to address the challenges of integrating information from multiple sources and leveraging knowledge graphs for improved semantic understanding. Their work highlighted persistent issues with hallucinations and the importance of grounding answers in reliable context.

In summary, the collective contributions of Nie et al., Lee et al., Yepes et al., and Shah et al. have laid the groundwork for robust, multimodal RAG pipelines in financial document analysis. Their work informs the present research by demonstrating the necessity of modality-aware parsing, structure-based chunking, and domain-specific evaluation. Building on these insights, this project advances the state of the art by implementing and rigorously evaluating a multimodal RAG system tailored to the unique challenges of financial reports, with a focus on optimizing answer quality, faithfulness, and practical utility for stakeholders.

3. Methodology

The methodology for building a robust multimodal Retrieval-Augmented Generation (RAG) pipeline for financial document question-answering is designed to address the challenges posed by complex, heterogeneous data found in real-world financial filings. Financial reports typically contain a blend of narrative text, structured tables, and graphical elements, each requiring specialized processing for accurate information extraction and synthesis. To enable precise, context-aware responses to user queries, our system integrates advanced vision-language models, efficient data chunking, semantic embedding, and a streamlined retrieval and generation workflow. This section details each stage of the architecture, from initial data ingestion to user interaction, highlighting the rationale and technical choices that underpin the system's effectiveness.

A. System Architecture and Design of the QnA RAG Pipeline

1) Input: Financial Report PDF as Data Source

A financial report in PDF format is ingested to start the procedure. These reports frequently contain a range of data formats, such as tables, graphical content, and free-form text. Every PDF page is transformed into an image to guarantee interoperability with multimodal extraction technologies. This transformation allows subsequent vision-based models to access and process all embedded data, regardless of its original format.

2) Multimodal Parsing Using Vision LLMs

Once the document pages are available as images, a vision-language model (VLM) is employed to extract information across modalities. The VLM is tasked with three core functions:

- *Text Extraction*: Identifying and extracting narrative passages, headings, and annotations.
- *Table Extraction*: Detecting tabular structures and converting them into structured JSON representations.
- *Graph/Chart Extraction*: Interpreting visual elements such as bar charts, line graphs, or pie charts, and translating their content into machine-readable text or numerical JSON format.

This approach ensures that all relevant information, regardless of modality, is captured for downstream processing.

3) Data Consolidation and Preprocessing

After extraction, the diverse outputs from text, tables, and charts are unified into a standardized format. This consolidation step is critical for maintaining consistency and enabling efficient querying. Preprocessing routines further clean and structure the data, bridging the gap between raw extraction and intelligent retrieval. This includes normalization, deduplication, and resolving ambiguities across modalities.

4) Ingestion Pipeline: Orchestrating RAG Techniques

The ingestion pipeline prepares the consolidated data for retrieval-augmented generation. Its key components are:

- *Text Splitting*: Dividing the unified content into smaller, contextually meaningful chunks. Splitting can be based on sentences, tokens, or document structure, optimizing both retrieval accuracy and model efficiency.
- *Text Embedding*: A pre-trained embedding model is used to embed each piece into a high-dimensional vector space. Similarity-based search is made possible by these embeddings, which capture the content's semantic meaning.
- *Vector Store Database*: Embeddings and their

associated metadata are stored in a vector database, allowing for rapid and scalable retrieval during user queries.

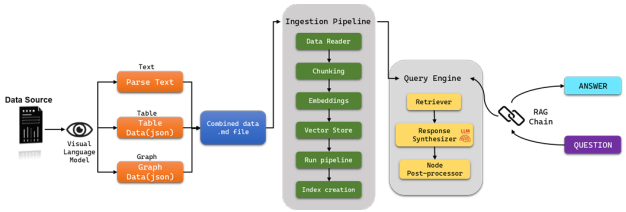

Fig. 1. System and RAG pipeline architecture

5) Retriever and Similarity Search

When a user submits a query, the system generates an embedding for the query and searches the vector database for the most semantically similar chunks. The retriever selects the top-n relevant results, using metrics such as cosine similarity. A similarity threshold is applied to filter out less relevant passages, ensuring that only the most pertinent information is considered for answer generation.

6) Response Synthesizer: Generating Human-Readable Answers

The retrieved context is then passed to a response synthesizer powered by a large language model. This component fuses the selected information, generating a coherent, natural-language answer that is both contextually grounded and easy for users to understand. The LLM is responsible for filling in minor contextual gaps and ensuring the response is comprehensive and fluent.

7) Query Engine for Interactive Question-Answering

The query engine integrates the retrieval and synthesis components, orchestrating the end-to-end flow from user query to answer delivery. It manages conversational state, maintains chat history, and ensures real-time responsiveness, facilitating an interactive and user-friendly experience.

8) Streamlit Chat Interface

To make the system accessible, a Streamlit - based chatbot interface is deployed. This interface allows users to upload PDF files, pose questions in natural language, and receive answers in a conversational format. The UI maintains a history of interactions and is designed for ease of use, requiring no technical expertise from end users.

B. RAG Hyperparameters

The effectiveness and flexibility of a Retrieval-Augmented Generation (RAG) pipeline depend heavily on the careful tuning of its core hyperparameters. These parameters directly influence the relevance, completeness, and factual accuracy of the answers generated in response to user queries, especially when dealing with complex and information-dense financial documents. Below are the primary RAG hyperparameters optimized in our system, along with their roles and trade-offs:

1) Top-K Retrieved Passages

In the retrieval stage of the RAG pipeline, the system identifies and fetches the most relevant context passages from a pre-indexed vector database. The “Top-K” parameter governs how many such chunks are selected for any given query.

A higher Top-K value (e.g., 8–10) broadens the scope of information retrieved, thereby improving contextual recall and the completeness of the generated answer. However, this can increase noise, potentially affecting precision. Conversely, a lower Top-K (e.g., 2–4) restricts the input to only the most relevant chunks, enhancing focus and relevance, but may risk omitting useful details.

This parameter represents a trade-off between answer breadth and specificity and must be tuned based on the complexity and information density of the source content.

2) Similarity Cutoff Threshold

The similarity cutoff is a threshold that helps decide whether a piece of information is close enough in meaning to the question being asked. It’s usually a value between [0 and 1]. The higher this number is set, the more strictly the system filters out loosely related or off-topic content. This is especially useful in tasks like financial question answering, where sticking to precise, relevant data is critical. Lowering the threshold allows more content in, but with that comes a greater chance of including unrelated or distracting information.

Using a higher value (e.g., 0.7 or more) means only strongly related information is brought in, which is helpful when accuracy and clarity are a priority — such as in financial use cases. However, being too strict may also leave out useful context that doesn't exactly match the query but is still valuable.

3) Embedding Mode (text_search vs similarity)

The embedding mode determines how queries and passages are semantically matched during retrieval. The system supports two modes:

- *Text_search*: Optimized for relevance based on keyword overlap and lexical similarity. It is particularly effective for structured or fact-based queries.
- *Similarity*: Prioritizes semantic closeness, making it more suitable for open-ended, conceptual questions common in financial analytics.

Selecting the appropriate mode is critical for aligning retrieval behavior with query intent. In financial use cases, similarity mode often provides superior results by capturing implicit relationships and domain-specific phrasing.

4) Text Splitters: Sentence-Based vs Token-Based

Before indexing, source documents are divided into smaller units known as "chunks" using text splitting strategies. The system supports:

- *SentenceSplitter*: Breaks documents at sentence boundaries, resulting in contextually coherent and human-readable segments. This improves generation quality and helps the model maintain narrative flow.
- *TokenTextSplitter*: Divides content strictly based on token count, regardless of sentence structure. This method is optimal for input length control, especially when working with models that have limited token windows.

C. RAG Evaluation Metrics

A set of focused assessment measures is employed to evaluate the Retrieval-Augmented Generation (RAG) pipeline's

performance in financial question-answering. These metrics are intended to measure the generated responses' relevance, contextual grounding, and semantic alignment with user inquiries in addition to their factual accuracy. We made sure that the responses are reliable and helpful for financial decision-making by assessing the system using a variety of measures.

1) Faithfulness

This metric measures the degree to which the generated answer is directly supported by the content retrieved from the source documents. An answer is considered faithful if every claim it makes can be traced back to the retrieved context. This metric is essential for minimizing hallucinations, as it penalizes responses that introduce information not present in the supporting passages. For example, if a query asks for a company's quarterly revenue, a faithful answer will state the exact figure found in the retrieved context, while an unfaithful answer will present unsupported or fabricated data.

$$\text{Faithfulness} = \frac{\text{Number of Truthful Claims}}{\text{Total Number of Claims}}$$

2) Answer Relevance

This metric evaluates how well the generated response addresses the user's question, independent of its source. A highly relevant answer directly and comprehensively responds to the query, while an irrelevant answer may include factual information that does not pertain to the user's intent. For instance, when asked about operational costs, the answer should focus on cost-related details, not on unrelated company activities. High answer relevance ensures that users receive information that is meaningful and actionable for their specific needs.

$$\text{Answer Relevancy} = \frac{\text{Number of Relevant Statements}}{\text{Total Number of Statements}}$$

3) Context Precision

Context precision assesses the proportion of retrieved context that is genuinely useful for answering the query. High precision indicates that most of the retrieved passages are directly relevant to the question, while low precision suggests the presence of extraneous or off-topic information. This metric is particularly important in financial QnA, where concise and focused retrieval helps prevent information overload and streamlines the answer synthesis process.

$$\text{Contextual Precision} = \frac{1}{\text{No. of Relevant Nodes}} \sum_{k=1}^{n} \left(\frac{\text{No. of Relevant Nodes Up to Position } k}{k} \times r_k \right)$$

4) Context Recall

This metric checks whether all key information needed to answer the query has been retrieved. High recall implies that no crucial details were missed, while low recall suggests that some important content was left out. In financial document analysis, high recall is essential to provide complete and accurate answers, especially when the question draws on information from multiple sections or formats.

$$\text{Contextual Recall} = \frac{\text{Number of Attributable Statements}}{\text{Total Number of Statements}}$$

5) Semantic Similarity

Semantic similarity quantifies how closely the meaning of the generated answer aligns with a reference or ground truth answer, regardless of the exact wording. This metric focuses on conceptual equivalence, ensuring that the response captures the intended substance of the correct answer. It is particularly useful for evaluating the system's ability to paraphrase and generalize, which is often necessary in financial analysis where terminology and phrasing can vary.

6) Answer Correctness

Answer correctness verifies the factual accuracy of the generated response by comparing it directly to the ground truth. This metric is vital for questions with objective, data-driven answers, such as financial ratios or reported figures. Correctness ensures that the system's outputs can be relied upon for critical financial decisions.

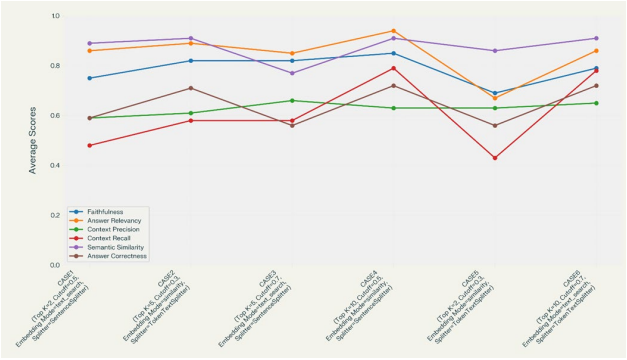

Fig. 2.

4. Results and Discussion

To systematically evaluate the performance and adaptability of the RAG pipeline, we conducted a series of experiments using six distinct configurations (CASE1–CASE6), each defined by different settings for key hyperparameters: the number of top passages retrieved (Top-K), the similarity cutoff threshold, the embedding mode (text_search versus similarity), and the choice of text splitting strategy (SentenceSplitter versus TokenTextSplitter).

For these experiments, we selected three publicly available corporate financial reports - ADIDAS, OLA ELECTRIC, and RELIANCE-as representative data sources. Each report features a blend of textual narratives, structured tables, and graphical information, ensuring a comprehensive test of the system's multimodal capabilities.

A set of 30 diverse questions was carefully curated for this evaluation. These questions were designed to span the full spectrum of modalities present in the reports, including queries that require understanding and reasoning over text, extracting data from tables, and interpreting information from graphs and charts. For each question, a corresponding ground truth answer was established to facilitate objective assessment.

A. Optimal Configuration: Case 4

Among all tested configurations, Case 4—characterized by a Top-K value of 10, a similarity cutoff threshold of 0.5, similarity-based embedding mode, and the use of Sentence Splitter for chunking—consistently yielded the strongest overall performance across multiple financial document types.

This configuration achieved superior outcomes across several critical evaluation metrics, including faithfulness, answer relevance, context recall, and factual correctness. The relatively high Top-K setting allowed the retrieval of sufficient context to support comprehensive answers, while the moderate similarity threshold (0.5) balanced relevance and coverage. This balance proved particularly effective in financial analysis tasks, where completeness and contextual grounding are essential for answer quality.

Importantly, this configuration maintained a strong equilibrium between context precision and recall, ensuring that the model was not only accurate but also thorough in capturing relevant insights. These results indicate that Case 4 is well-suited for use cases where financial reasoning and factual accuracy are prioritized.

B. Impact of Sentence-Based Splitting

The use of the Sentence Splitter emerged as a key contributor to improved answer quality. By segmenting documents at natural language boundaries, this strategy produced context chunks that were semantically coherent and easier for the language model to interpret.

Unlike token-based splitting, which can arbitrarily cut across sentence boundaries and introduce fragmented or contextually incomplete inputs, sentence-level splitting preserved the integrity of information units. This had a measurable positive impact on the generation quality, particularly in financial reports where contextual continuity and logical flow are crucial.

Thus, sentence-based splitting is especially beneficial when the primary objective is to ensure answer fluency, factual grounding, and readability, rather than maximizing recall of specific named entities.

C. Effectiveness of Similarity-Based Embeddings

Embedding mode also played a significant role in retrieval performance. The similarity mode—designed to capture semantic closeness rather than keyword overlap—outperformed the text_search alternative in financial contexts.

Financial queries often involve conceptually dense language and varied terminology. By leveraging semantic similarity, the system was better able to retrieve passages that were meaningfully aligned with the user's intent, even when exact phrasing differed. This was particularly evident in tasks involving qualitative financial insights or multi-faceted analytical queries, where lexical matching alone would have been insufficient.

Overall, similarity-based embedding retrieval contributed to higher context relevance and faithfulness, supporting more accurate and informative responses.

5. Conclusion

This paper shows the transformative potential of large language models (LLM) and multimodal AI to deal with the complexity of financial documents analysis. By integrating advanced natural language processing (NLP), visual language and technology models (RAG), the system is efficiently extracted, processes and synthesizes information from different data formats, including text, tables and graphics images commonly found in financial reports. The main methodology using multimodal analysis and sophisticated strategies of pieces ensures that critical financial knowledge is accessible and interpreted for end users, regardless of their technical background. Rag pipeline continuously delivered a high degree of loyalty, relevance of responses, and factual correctness after rigorously experimenting with hyperparameters such Top-K search and similarity thresholds.

Notably, the adoption of sentence-based chunking and semantic similarity embeddings proved instrumental in improving both the coherence and accuracy of generated responses, particularly for complex analytical queries. Evaluation results across multiple financial documents and configurations highlight the system's robustness and adaptability. The optimal configuration-characterized by a balanced Top-K setting, moderate similarity cutoff, and sentence-based splitting-achieved superior performance in retrieving contextually relevant information and generating factually grounded answers. This not only streamlines the traditionally labor-intensive process of financial analysis but also minimizes errors and enhances decision-making for stakeholders such as analysts, investors, and auditors.

Through this project, it is evident that LLMs-when combined with robust retrieval strategies and multimodal data processing-can move beyond mere support tools to play a central role in financial analysis and decision-making. However, the work also acknowledges the current limitations of LLMs, particularly in precise numerical reasoning and complex forecasting, where experienced human analysts still maintain an edge. Nonetheless, the demonstrated improvements in accessibility, efficiency, and reliability mark a significant advancement in the field of financial information systems.

References

[1] Y. Nie, Y. Kong, X. Dong, J. M. Mulvey, H. V. Poor, Q. Wen, and S. Zohren, "A Survey of Large Language Models for Financial Applications: Progress, Prospects and Challenges," *CoRR*, arXiv:2406.11903, Jun. 2024.

[2] J. Lee, N. Stevens, S. C. Han, and M. Song, "A Survey of Large Language Models in Finance (FinLLMs)," *CoRR*, arXiv:2402.02315, Feb. 2024.

[3] A. J. Yepes, Y. You, J. Milczek, S. Laverde, and L. Li, "Financial Report Chunking for Effective Retrieval Augmented Generation," *CoRR*, arXiv:2402.05131v3, Feb. 2024.

[4] S. Shah, S. Ryali, and R. Venkatesh, "Multi-document Financial Question Answering Using LLMs," arXiv preprint, Nov. 2024.

[5] Z. Duan and J. Wang, "Exploration of LLM Multi-Agent Application Implementation Based on LangGraph+CrewAI," arXiv preprint, 2023.

[6] LlamaIndex, "Q&A with LlamaIndex: Efficient Querying for Financial Reports," LlamaIndex Documentation, Nov. 25, 2024.

[7] "Multi-modal RAG: Enhancing Financial Data Analysis with Multimodal Retrieval-Augmented Generation," LlamaIndex Blog, Nov. 12, 2024.

[8] L. Espinosa-Anke, et al., "RAGAS: Automated Evaluation of Retrieval Augmented Generation," arXiv preprint arXiv:2309.15217, Sep. 2023.