CONTEXT IS ALL YOU NEED: EFFICIENT RETRIEVAL AUGMENTED GENERATION FOR DOMAIN SPECIFIC AI

Peixi Xiong, Chaunte W. Lacewell, Sameh Gobriel & Nilesh Jain
Intel Labs
{peixi.xiong, chaunte.w.lacewell, sameh.gobriel, nilesh.jain}
@intel.com

ABSTRACT

Effective Retrieval-Augmented Generation (RAG) pipelines face significant challenges when processing domain-specific technical documents containing diverse content types like text, figures, equations, and tables. We introduce CoRAG, Context-oriented RAG for domain-specific applications, which enhances contextual understanding through a lightweight, two-pipeline architecture: Content Analysis & Enrichment for structured metadata extraction, and Query Processing for context-aware retrieval. Our approach emphasizes preserving structural relationships and semantic connections across different modalities, enabling more precise technical information retrieval. Evaluated on the TSpec- LLM dataset of complex Telco 3GPP technical specifications, CoRAG achieves 77.00% accuracy while using smaller models than current state-of-the-art methods, establishing a new benchmark for **telco-RAG** applications. The system's efficient design and comprehensive context handling make advanced RAG capabilities more accessible for domain-specific use while maintaining high performance across varying levels of technical complexity.

1 INTRODUCTION

The growing complexity of technical, domain-specific documentation, which increasingly incorporates a diverse array of content types, including text, figures, equations, tables, and visualizations, presents significant hurdles for effective information retrieval and comprehension. Although these multimodal elements enhance clarity, they also introduce intricacies in maintaining structural relationships and achieving comprehensive contextual understanding across modalities. The importance of context in technical documentation is paramount, as it is crucial for accurate interpretation, particularly in domain-specific applications where context determines meaning. However, large language models, such as GPT-4V and Claude 3, which are trained on general information, lack the specialized knowledge and nuances inherent to technical, domain-specific data, and also require substantial computational resources (Xu et al., 2024), making them less accessible for widespread adoption. Moreover, these models, which underpin typical Retrieval-Augmented Generation (RAG) pipelines (Lewis et al., 2020), often fail to capture the intricate structural relationships inherent in technical documentation, highlighting the need for more sophisticated approaches to contextual understanding that can effectively address these challenges and provide high-quality context for accurate interpretation.

To address these challenges, we present CoRAG, a structure-aware pipeline that achieves superior contextual understanding through two key stages. The Content Analysis & Enrichment pipeline employs: (1) a Document Parser that preserves hierarchical relationships, (2) a Media Handler that maintains structural integrity through dual-stream processing, and (3) a Visual Summarizer that generates context-rich semantic descriptions. The Query Processing pipeline leverages this enriched structural context to enable precise retrieval and contextually grounded response generation. We developed CoRAG on top of Visual Data Management Systems (VDMS) (Remis et al., 2018) framework, which enables users to define efficient ingestion and retrieval pipelines designed to handle vector embeddings, meta-data store, and multimodal data

To demonstrate the effectiveness of CoRAG in real-world applications, we apply our technology to the TSpec-LLM dataset (Rasoul Nikbakht & Geraci, 2024), a highly specialized collection of Telco domain-specific technical documents that pose significant challenges due to their complex elements, such as acronyms, tables, block diagrams, figures, and equations. The dataset is specifically designed to evaluate the capabilities of RAG systems in the Telco domain, where precise understanding of technical concepts is essential. We assess CoRAG's ability to navigate the intricacies of technical documentation and provide high-quality context for informed decision-making using the TSpec-LLM dataset, which comprises documents with complex structures, tables, and formulas. Notably, our evaluation on this dataset yields a superior accuracy, demonstrating CoRAG's effectiveness in establishing a new benchmark for Telco domain-specific RAG applications and highlighting its potential to advance the state-of-the-art in this field.

Our Key contributions can be summarized as follows:

1. A novel, structure-aware approach to RAG that captures and leverages the intricate relationships between technical concepts, enabling better context and more precise responses.
2. A lightweight and efficient pipeline that achieves superior performance with reduced computational requirements, making it more accessible for widespread adoption.
3. A comprehensive framework for preserving and utilizing rich contextual relationships, including hierarchical relationships, structural integrity, and semantic connections, which provides better context for accurate retrieval and generation of technical information.
4. A demonstration of CoRAG's effectiveness in a real-world application, showcasing how the improved context enabled by our approach leads to more accurate and relevant answers, and establishing a new benchmark for Telco domain-specific RAG applications

2 Related Works

2.1 Retrieval Augmented Generation

Retrieval-Augmented Generation (RAG) has become a key paradigm for enhancing large language models (LLMs) by integrating external knowledge bases. General RAG frameworks, such as (Lewis et al., 2020), combine retrieval systems with neural generation, offering a robust solution for knowledge-intensive tasks. These models retrieve contextually relevant data from external sources, which is then fused with LLM outputs to improve accuracy and reduce hallucinations.

Recent advancements have tailored RAG frameworks for specialized domains. For example, (Barron et al., 2024) explored vector stores and knowledge graphs for targeted domain retrieval, while (Xu et al., 2024) introduced self-improving RAG frameworks to adapt LLMs for medical and technical domains. These developments highlight the need for customized retrieval strategies and fine-tuning for domain-specific knowledge integration.

Structured RAG has gained traction in applications requiring formalized data representations. Integrating structured data sources, such as knowledge graphs and SQL-like databases, enhances precision in extracting and using domain-specific knowledge, as demonstrated by (Govindharajan & Vijayakumar, 2024). Hierarchical frameworks like (Long et al., 2024) further optimize retrieval by considering the structural context of data, showing effectiveness in medical applications.

3 Methods

3.1 Overview

Our CoRAG system employs a two-stage pipeline (Figure 1): content analysis transforms raw technical documents into contextually-enriched, structured representations, while query processing leverages this enhanced context to identify relevant information and generate comprehensive, context-aware responses.

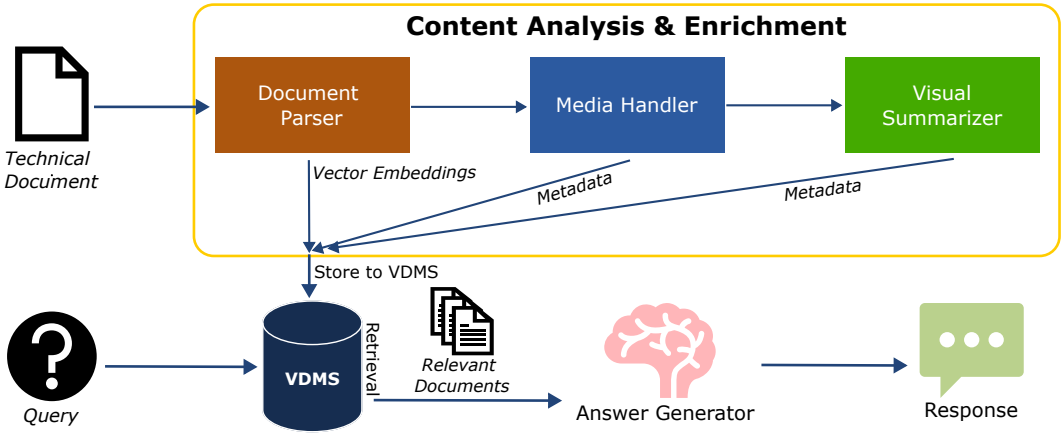

Figure 1: Architecture Overview of CoRAG. The system comprises two main components: (1) the Content Analysis & Enrichment pipeline (yellow), which establishes comprehensive contextual understanding through document parsing, media handling, and visual summarization with rich metadata and semantic relationships; and (2) the Query Processing pipeline, which leverages this enhanced context to retrieve relevant information and generate contextually-grounded responses. Together, these components enable accurate analysis and contextual understanding of complex technical documentation.

3.2 Content Analysis and Enrichment

The Content Analysis and Enrichment section underpins CoRAG, using three modules—Document Parser, Media Handler, and Visual Summarizer—to convert raw documents into structured, metadata-rich, vectorized content. This processed information is stored in VDMS, enabling efficient retrieval and context-aware responses while preserving semantic relationships across content types (The structured representation of pipeline outputs is shown in Appendix Table 3.).

3.2.1 Document Parser

The Document Parser module processes various technical document formats, with a focus on HTML and PowerPoint presentations. It employs a hierarchical chunking strategy to break documents into manageable segments while preserving their structural integrity. The parser extracts textual content and identifies embedded media such as images and tables. To maintain contextual continuity, it uses an overlapping window technique during segmentation, ensuring semantic relationships between chunks are preserved. Additionally, unique identifiers are generated for each extracted element, creating traceable links between related content pieces. This structured approach ensures that relationships between document elements are maintained throughout the processing pipeline, facilitating accurate retrieval and context preservation in subsequent stages.

3.2.2 Media Handler

The Media Handler module processes multimedia elements from technical documents through a dual-stream approach for images and tables, focusing on contextual relationships and structural integrity. The module implements quality assessment criteria that evaluate image resolution and clarity, filtering out low-quality visual content while preserving figure titles, captions, and related textual elements. For tables, it utilizes a dual-processing methodology - processing them as machine-readable text for semantic analysis and as visual screenshots to maintain structural information. The module associates each media element with contextual metadata, including figure titles, captions, and cross-references, establishing semantic connections between visual and textual components. Image standardization processes ensure format consistency while maintaining technical fidelity, while the table parsing methodology preserves both semantic content and layout structure. This integrated approach maintains technical precision while supporting contextual analysis, enabling accurate information retrieval and response generation within the technical domain.

3.2.3 VISUAL SUMMARIZER

The Visual Summarizer module enhances processed content by generating detailed descriptions of visual elements in technical documentation. It employs machine learning models to produce comprehensive descriptive summaries and semantic keywords for images and complex visualizations, with configurable parameters for detail level and technical specificity. This enrichment operates in two modes: direct image analysis or text-based analysis of generated descriptions, providing systematic handling of diverse visual content. The module maintains contextual relationships by extracting and preserving figure titles, related text, and technical metadata during the summarization process. The resulting summaries and keywords are vectorized alongside the original content, establishing a semantic embedding space that integrates visual and textual relationships. This integrated approach enables context-aware retrieval of visual content in response to technical queries, while maintaining the precise technical nature of the documentation.

3.3 QUERY PROCESSING

The query processing component uses a RAG approach to transform user queries into dense vectors, enabling semantic consistency with document embeddings for efficient similarity-based retrieval.

3.3.1 VECTOR-BASED RETRIEVAL IN VDMS

VDMS enables efficient retrieval of multimodal technical content by leveraging dense vector representations of text, tables, and visuals stored during content analysis. Using inner product distance metrics, it performs similarity computations between query vectors and stored vectors. A configurable top-K retrieval strategy selects the most semantically relevant documents, handling both unimodal and multimodal queries effectively. The retrieved content provides comprehensive context for answer generation, integrating insights from both textual and visual sources.

3.3.2 CONTEXT-AWARE ANSWER GENERATION

The answer generation component synthesizes responses by combining retrieved context and multimodal information. Retrieved documents are processed to preserve textual structure and semantic relationships, while descriptions and metadata of visual elements enrich the context. This preprocessed data is fed into a large language model using tailored prompts that integrate textual and visual evidence. For structured queries, an additional analysis aligns responses with provided options, ensuring accuracy and grounding in the source material with explicit links to supporting evidence.

4 EXPERIMENTS

4.1 DATASET

We evaluate our system on the TSpec-LLM dataset (Rasoul Nikbakht & Geraci, 2024), specifically chosen for its domain-specific technical complexity and comprehensive structural preservation. It maintains original document structures, including tables and formulas, making it ideal for evaluating both contextual understanding and processing efficiency. The dataset provides 100 technically complex questions from 3GPP Releases 15–17, enabling rigorous evaluation of accuracy—defined as the percentage of correct answers—across models like GPT-3.5, GPT-4, and Gemini Pro 1.0, with and without naive-RAG integration.

4.2 QUANTITATIVE RESULTS

4.2.1 OVERALL PERFORMANCE

Our experimental results demonstrate the effectiveness of CoRAG across different difficulty levels of technical queries. As shown in Table 1, our system achieves competitive performance with an overall accuracy of 77.00%, surpassing larger models like GPT and Gemini-1.0. Notably, our approach performs particularly well on challenging queries, achieving 73.68% accuracy on hard questions compared to 16.00% for GPT-4.0 and 36.00% for Gemini-1.0. While other RAG-enhanced models show improved performance, our system maintains superior results despite utilizing smaller-scale

Difficulty	Methods						
	GPT-3.5	Gemini-1.0	GPT-4	RAG + GPT-3.5	RAG + GPT-4	RAG+ Gemini-1.0	Ours
Easy	70.00%	67.00%	80.00%	-	-	**93.00%**	80.00%
Interm.	39.00%	37.00%	47.00%	-	-	65.00%	**76.47%**
Hard	16.00%	36.00%	26.00%	-	-	66.00%	**73.68%**
Overall	44.00%	46.00%	51.99%	71.00%	72.00%	75.00%	**77.00%**

Table 1: Performance accuracy of various methods across difficulty levels (Easy, Intermediate, Hard) and overall performance. The "Interm." column represents the Intermediate difficulty category.

Settings					Performance			
No.	Answer Gen. Module	Pre-processing	Metadata: Title	Metadata: Visual Fea	Easy	Interm.	Hard	Overall
1	**llava**	✗	✗	✗	60.00%	58.82%	57.89%	59.00%
2	**llava**	✓	✗	✗	63.33%	56.86%	57.89%	59.00%
3	**llama3.2-vis**	✓	✗	✗	63.33%	62.75%	68.42%	68.42%
4	**llama3.2-vis**	✓	✓	✗	73.33%	66.67%	68.42%	69.00%
5	**llama3.2-vis**	✓	✓	✓	80.00%	76.47%	73.68%	77.00%

Table 2: Quantitative analysis and ablation study comparing the performance of different settings for the Answer Generation Module under varying conditions. The "Interm." column represents the Intermediate difficulty category. Performance is evaluated across Easy, Intermediate, Hard, and Overall difficulty levels, with metadata contributions and pre-processing variations highlighted.

models. This efficiency-performance balance is particularly evident in the intermediate category, significantly outperforming baseline models. These results suggest that our pipeline, combining efficient document processing with context-aware response generation, can effectively handle complex technical documentation queries while maintaining computational efficiency.

4.2.2 Ablation Study

We conducted ablation studies to evaluate how each component contributes to contextual understanding (Table 2). The complete system (Exp. 5) achieved the highest performance at 77.00%, demonstrating the importance of comprehensive context capture. The hierarchical document processing and contextual metadata enrichment showed consistent improvements, with enhanced cross-modal context boosting performance from 68.42% to 69.00% (Exp. 3 vs. 4), particularly for complex technical queries. Integrating all contextual components resulted in an 8-point gain over the baseline (Exp. 1), validating our context-centric approach.

4.3 Qualitative results

We provide a qualitative analysis of CoRAG's performance across query difficulty levels (Figure 2). For easy queries, the system retrieves precise answers, such as minimum elevation angles. In intermediate cases, like Doppler shift calculations, it integrates graphical and textual data effectively. For complex queries, such as extracting room dimensions from tabular data, the system processes detailed technical specifications accurately. These examples highlight CoRAG's ability to handle diverse data sources with high accuracy across varying query complexities.

5 Conclusions

We presented CoRAG, a novel context-oriented approach that advances technical document understanding through comprehensive contextual analysis and efficient Retrieval-Augmented Generation. Our evaluation on the TSpec-LLM dataset demonstrates superior performance compared to larger models, validating our strategy for processing complex technical documentation. The system's success stems from its ability to preserve and leverage rich contextual relationships through hierarchical document parsing, dual-stream media processing, and sophisticated cross-modal understanding. Fu-

ture work could extend this context-aware approach to other specialized domains while enhancing the framework to capture even more nuanced technical relationships and domain-specific semantic connections.

Question: What is the typical minimum elevation angle for aeronautical satellite and aerial systems?
Ground Truth Answer: 20 degrees
Difficulty: Easy

VDMS-RAG Response: 20 degrees
VDMS-RAG Retrieved:
Table 4.5-2: Typical elevation angles in aerial and satellite based systems

Satellite & aerial Systems	Typical minimum Elevation Angle for terminals	Rationale/remarks
International GEO (Trunking)	5 degrees	Serving earth stations equipped with very large antennas
Regional GEO	10 degrees	Addressing regions in lower and medium latitude
International (GEO) Maritime	5 degrees	Addressing large ships
Aeronautical	20 degrees	Taking into account aero-dynamic constraints prevents operation at lower angles
Vehicles	15 degrees	Taking into account road conditions, terrain, and vehicle mechanics
Non GSO	10 to 30 degrees	Ensuring service continuity optimising the number of satellites
Aerial	In the range of 10 degrees	Maximising the service area

Question: What is the maximum rate of change of Doppler shift for a high-speed train travelling north from the Equator?
Ground Truth Answer: -23 mHz/s
Difficulty: Intermediate

VDMS-RAG Response: -23 mHz/s
VDMS-RAG Retrieved:
In the figure below, we consider a train travelling North from the Equator at a constant speed of 500 km/h. The two extreme points in the figure are 3600 seconds apart. The Doppler shift plotted as a function of the latitude. The maximum rate of change of Doppler shift is approximately -23 mHz/s.

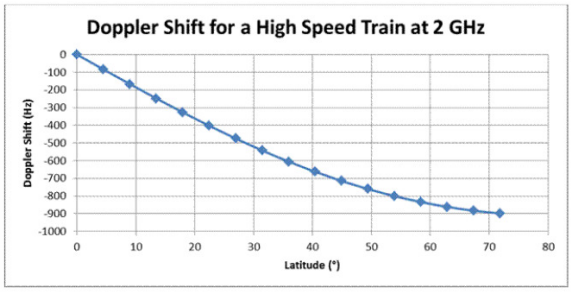

Figure 5.3.2.3-3: Doppler Shift at 2 GHz for a High Speed Train travelling along a longitude (North direction)

Question: In the Indoor-office scenario, what is the room size for the open office?
Ground Truth Answer: 120mx50mx3m
Difficulty: Hard

VDMS-RAG Response: 120mx50mx3m
VDMS-RAG Retrieved:

Table 7.2-2: Evaluation parameters for indoor-office scenarios

Parameters		Indoor - office open office	Indoor - office mixed office
Layout	Room size (WxLxH)	120mx50mx3m	
	ISD	20m	
BS antenna height h_{BS}		3 m (ceiling)	
UT location	LOS/NLOS	LOS and NLOS	
	Height h_{UT}	1 m	
UT mobility (horizontal plane only)		3 km/h	
Min. BS - UT distance (2D)		0	
UT distribution (horizontal)		Uniform	

Figure 2: Qualitative results. Due to page limitation, we only show top-1 retrieve results.

REFERENCES

Ryan C Barron, Ves Grantcharov, Selma Wanna, Maksim E Eren, Manish Bhattarai, Nicholas Solovyev, George Tompkins, Charles Nicholas, Kim Ø Rasmussen, Cynthia Matuszek, et al. Domain-specific retrieval-augmented generation using vector stores, knowledge graphs, and tensor factorization. *arXiv preprint arXiv:2410.02721*, 2024.

Hariharan Govindharajan and Senthilkumar Vijayakumar. A framework for automated selective fine-tuning of domain-specific large language models using graph-based retrieval augmented generation. In *2024 IEEE 15th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pp. 431–439. IEEE, 2024.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474, 2020.

Cui Long, Yongbin Liu, Chunping Ouyang, and Ying Yu. Bailicai: A domain-optimized retrieval-augmented generation framework for medical applications. *arXiv preprint arXiv:2407.21055*, 2024.

Mohamed Benzaghta Rasoul Nikbakht and Giovanni Geraci. Tspec-llm: An open-source dataset for llm understanding of 3gpp specifications, 2024.

Luis Remis, Vishakha Gupta-Cledat, Christina R. Strong, and Ragaad AlTarawneh. VDMS: an efficient big-visual-data access for machine learning workloads. *CoRR*, abs/1810.11832, 2018. URL `http://arxiv.org/abs/1810.11832`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. C-pack: Packaged resources to advance general chinese embedding, 2023.

Ran Xu, Hui Liu, Sreyashi Nag, Zhenwei Dai, Yaochen Xie, Xianfeng Tang, Chen Luo, Yang Li, Joyce C Ho, Carl Yang, et al. Simrag: Self-improving retrieval-augmented generation for adapting large language models to specialized domains. *arXiv preprint arXiv:2410.17952*, 2024.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024.

A APPENDIX

A.1 IMPLEMENTATION DETAILS

Our system implementation incorporates several key components optimized for technical document processing and response generation. For retrieval, we employ a top-5 setting in the Visual Data Management System (VDMS) to balance comprehensive context gathering with computational efficiency. The visual summarization component utilizes MiniCPM-V-2.6 (Yao et al., 2024), a compact yet powerful vision-language model, to generate descriptive summaries of technical figures and diagrams. For generative tasks, we leverage Llama 3.2 Vision model (11B) (Touvron et al., 2023) through the Ollama framework, which demonstrates robust capabilities in handling both textual and visual inputs. The system employs carefully crafted prompts that guide the model to consider both document structure and visual elements. For example, when processing tables, the prompt instructs the model to analyze structural layout, data relationships, and key patterns, ensuring comprehensive understanding of tabular information. The prompt engineering also incorporates specific guidance for handling technical terminology and maintaining consistency with 3GPP documentation standards. Document chunking is implemented with a 500-token chunk size and appropriate overlap to preserve context continuity while maintaining efficient processing. To enhance retrieval precision, we utilize the BGE base embeddings (v1.5) (Xiao et al., 2023) for textual vector representation, which has shown strong performance on technical domain tasks. Our current version only vectorizes the textual summary, not the image.

A.2 DATA STRUCTURE IN CORAG

Content Type	**Data**	**Metadata/Features**
Text	Document content	source path title chunk_id text_id
Image	Base64 encoded image	image_path image_id figure_title description (vectorized) keywords source path
Table	Table content Table screenshot	table_path table_id table_title format_type (text/screenshot) description (vectorized) source path

Table 3: Data and Metadata Structure in CoRAG