MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 3043-3050

VectoRAG: A Document-Grounded Retrieval- Augmented Generation System with Page-Level Traceability for Academic CSE Resources

Janani S, Assistant Professor Department of CSE, KGiSL Institute of Technology
Coimbatore,Tamilnadu,India
janani441992@gmail.com

Saishree S Department of CSE
KGiSL Institute of Technology
Coimbatore,Tamilnadu,India
sairithu33@gmail.com

Varshini Shree V Department of CSE
KGiSL Institute of Technology
Coimbatore,Tamilnadu,India
varshinishreevelumani@gmail.com

Murugan K Department of CSE
KGiSL Institute of Technology
Coimbatore,Tamilnadu,India
murugangan71@gmail.com

Rohini S Department of CSE
KGiSL Institute of Technology
Coimbatore,Tamilnadu,India
rohinisiva05@gmail.com

Abstract*—Retrieval-Augmented Generation (RAG) systems combine large language models with external knowledge sources to improve factual grounding. In academic settings, however, conventional RAG implementations often lack fine- grained traceability, making it difficult for learners to verify the origin of generated responses. This challenge is particularly relevant in computer science education, where contextual precision and source citation are essential.This paper presents VectoRAG, a hybrid retrieval-augmented generation system designed to provide document-grounded answers with page-level traceability for academic CSE resources. The architecture integrates dense vector retrieval and lexical search while preserving metadata across document chunks to enable precise citation mapping. Retrieved evidence is assembled into constrained prompts to support reliable answer generation and reduce unsupported outputs. In addition to question answering, the system enables evidence-based quiz generation derived directly from uploaded documents.*

A baseline comparison conducted on curated academic queries suggests improvements in citation accuracy and grounding behaviour relative to a vector-only retrieval pipeline. The findings highlight the importance of metadata-aware hybrid retrieval in building transparent and educationally reliable RAG systems.

Keywords—Retrieval-Augmented Generation, hybrid retrieval, page-level traceability, document-grounded learning, quiz generation, academic AI systems.

I. INTRODUCTION

Large language models have become increasingly prevalent in educational and technical environments, where they assist users in summarizing concepts, answering questions, and interpreting domain-specific material. Despite their fluency, such models rely primarily on parametric knowledge acquired during pre-training [14], limiting their ability to reference updated, institution-specific, or course-aligned resources. In academic settings, particularly within computer science and engineering (CSE), this lack of traceable grounding presents a challenge: responses may appear coherent but cannot always be verified against authoritative materials.

Retrieval-Augmented Generation (RAG) addresses this limitation by integrating external document repositories into the inference pipeline. The foundational RAG framework demonstrated that retrieving relevant passages at query time and conditioning generation on them significantly improves performance on knowledge-intensive tasks [1]. Subsequent advancements in dense passage retrieval further enhanced semantic matching between queries and documents through dual-encoder architectures [2]. Embedding models such as Sentence-BERT improved the quality of contextual representations, enabling scalable semantic search across large corpora [3].

While dense retrieval effectively captures semantic similarity, it may overlook precise terminology and symbolic expressions common in technical documents. Classical lexical retrieval methods such as BM25 remain strong baselines for exact term matching [4]. Hybrid retrieval strategies that combine semantic and lexical signals have shown improved robustness across heterogeneous benchmarks [7], [11]. Efficient approximate nearest neighbour search mechanisms, including FAISS indexing [5] and Hierarchical Navigable Small World graphs [6], have enabled scalable deployment of embedding-based retrieval systems in real-world applications.

Despite these advances, practical RAG implementations in educational environments often lack fine-grained traceability. Hallucination and unsupported claims remain ongoing concerns in generative systems [9]. While techniques such as reflection-based retrieval [13] and domain adaptation methods including parameter-efficient fine-tuning [10] have been proposed to improve reliability, they do not inherently guarantee explicit citation mapping at the page level. Moreover, generative systems are increasingly used not only for answering questions but also for producing structured learning artifacts, such as quizzes, which require careful grounding in source material.

In academic contexts, the ability to provide responses with explicit page-level references is critical for transparent learning. Furthermore, the capability to generate assessment questions derived directly from retrieved content can support structured revision while maintaining alignment with instructional materials.The key contributions of this study are as follows: Metadata-Continuity RAG Architecture: A dedicated system that maintains structural consistency during ingestion, guaranteeing accurate page-level tracking within academic PDFs.Hybrid Retrieval Fusion: A refined approach that merges dense semantic searching (utilizing Transformer embedding) with sparse lexical retrieval (through BM25) to proficiently manage technical CSE terminology. Citation-Constrained Generation: A system that limits LLM inference (through Tiny Llama) to gathered proof, greatly reducing the chances of hallucinations. Unified Learning Framework: A cohesive, modular structure that facilitates citation-supported question answering and automated, retrieval-based quiz generation. Motivated by these requirements, this work presents VectoRAG, a hybrid retrieval-augmented generation system designed for academic CSE resources. The system integrates dense and lexical retrieval within a unified architecture, preserves document metadata during ingestion, and constrains generation to retrieved evidence to enable traceable, citation- backed responses. In addition, the same retrieval pipeline supports quiz generation grounded in uploaded documents, extending the system's utility beyond question answering to structured assessment.Rather than proposing a novel retrieval algorithm, this work focuses on architectural coherence, metadata-aware processing, and domain-oriented deployment considerations for educational RAG systems.

II. RELATED WORK

Retrieval-Augmented Generation (RAG) was introduced to address the limitations of purely parametric language models by incorporating external document retrieval during inference [1]. By retrieving relevant passages and conditioning generation on them, RAG significantly improved performance on knowledge-intensive tasks while reducing unsupported outputs. This paradigm established the foundation for combining retrieval systems with large language models in a unified pipeline.

Dense retrieval techniques further advanced this direction by employing dual-encoder architectures trained to maximize similarity between queries and relevant passages [2]. Embedding-based models such as Sentence-BERT enabled efficient semantic search across large corpora by generating fixed-length contextual representations for both queries and documents [3]. These approaches demonstrated strong generalization capabilities across diverse retrieval benchmarks [11]. However, dense retrieval alone may not adequately capture exact terminology, symbolic tokens, or code-specific expressions frequently encountered in technical educational materials.

Classical lexical retrieval methods, particularly BM25, remain competitive baselines for information retrieval due to their effectiveness in exact term matching [4]. The complementary strengths of dense and sparse retrieval have motivated hybrid approaches that combine semantic similarity with lexical precision. Fusion techniques such as reciprocal rank fusion have shown improvements over individual ranking methods in heterogeneous retrieval settings [7]. Efficient large-scale similarity search infrastructure, including FAISS [5] and Hierarchical Navigable Small World indexing [6], has enabled practical deployment of such hybrid pipelines in real-world systems.

Beyond retrieval effectiveness, hallucination and factual inconsistency remain central challenges in generative models. Surveys on hallucination in natural language generation highlight the need for grounded evidence integration and verifiable outputs [9]. Reflection-based retrieval architectures such as Self-RAG attempt to iteratively refine retrieval and generation through internal critique mechanisms [13]. While these approaches enhance factual reliability, they introduce additional computational complexity and are not explicitly optimized for educational traceability at the document-page level.Domain adaptation strategies, including parameter-efficient fine-tuning techniques such as LoRA [10], aim to improve model alignment with specialized domains. However, fine-tuning does not inherently provide dynamic knowledge access or explicit citation mapping. Retrieval-based augmentation remains more suitable for scenarios where content is continuously updated or institution-specific. Early generative retrieval models demonstrated the effectiveness of conditioning generative systems on retrieved passages for open-domain question answering [8], reinforcing the importance of retrieval-informed generation.

Although substantial progress has been made in retrieval algorithms, vector indexing, and grounding mechanisms, relatively limited attention has been given to system architectures designed specifically for academic environments requiring page-level traceability and assessment generation. Existing RAG implementations often focus on open-domain benchmarks rather than structured educational workflows. In contrast, VectoRAG emphasizes metadata-aware processing, hybrid retrieval integration, and dual-mode output—supporting both citation-backed question answering and retrieval-grounded quiz generation within a unified architecture tailored to academic CSE resources.

III. PROBLEM STATEMENT

Despite the rapid adoption of Large Language Models (LLMs) for educational assistance and technical question answering, their direct applicability in academic environments remains constrained. General-purpose LLMs generate responses based on parametric knowledge acquired during large-scale pre-training [14], without explicit alignment to prescribed instructional materials. Consequently, their outputs, while fluent, are often unverifiable against curriculum-specific resources and lack transparent attribution to authoritative documents. In academic Computer Science and Engineering (CSE) contexts, such limitations reduce trust, complicate validation, and restrict meaningful institutional adoption.Retrieval-Augmented Generation (RAG) partially addresses this challenge by grounding model outputs in external documents retrieved during inference [1]. Dense retrieval techniques improve semantic alignment between queries and relevant passages [2], [3], while lexical retrieval methods such as BM25 preserve exact term matching critical in technical material [4]. Hybrid ranking strategies further combine these signals to enhance retrieval robustness [7]. However, when applied to academic workflows, many existing implementations exhibit structural limitations.First, citation granularity is frequently coarse. Many systems reference entire documents or broad sections without preserving page-level alignment. In academic settings, where students and instructors must verify statements against specific textbook pages or lecture notes, such coarse attribution reduces practical usability.

Second, document preprocessing pipelines often fragment PDFs into independent text chunks without preserving page identifiers or structural metadata. The loss of document hierarchy undermines traceability and complicates verification.

Third, many RAG systems depend on cloud-based APIs and services, limiting deployment in environments with restricted connectivity, privacy constraints, or examination policies.

From an academic systems perspective, improving answer accuracy alone is insufficient. A viable solution must additionally:

1. Preserve document structure and metadata during ingestion.
2. Support fine-grained, page-level traceability.
3. Constrain answer generation to retrieved evidence.
4. Operate in resource-constrained educational environments.
5. Extend beyond question answering to structured assessment generation.

Table I compares general-purpose LLMs, typical RAG systems, and the proposed VectoRAG architecture across these dimensions.

Table I

Feature	General Purpose LLMs	Typical RAG Systems	VectoRAG(P roposed)
Answer Grounding in custom documents	No	Yes	Yes
Dependence on open- domain knowledge	Yes	Partial	Minimal
Support for academic CSE PDFs	Limited	Partial	Yes
Page-level traceability	No	Rare	Yes
Preservation of document structure	No	Partial	Yes
Offline/ Local Deployment	No	Rare	Yes
Use of proprietary APIs	Yes	Often	No
Feature	**General Purpose LLMs**	**Typical RAG Systems**	**VectoRAG(P roposed)**
Support for retrieval- grounded quiz generation	No	Limited	Yes
Suitability for academic verification	No	Limited	Yes

The comparison highlights systemic gaps in traceability, metadata preservation, offline operability, and assessment alignment. Existing systems prioritize retrieval effectiveness but are not architected specifically for academic verification requirements.Thus, the fundamental issue tackled in this study can be expressed as follows:What design considerations are necessary for a Retrieval-Augmented Generation system to deliver precise, document-based responses with clear page-level traceability, while also being suitable for deployment in limited- resource academic CSE settings and facilitating retrieval-based assessment generation?

Formally, let

$$\mathcal{D} = \{d_1, d_2, \ldots, d_n\}$$

denote a collection of academic documents, where each document d_iconsists of ordered pages,

$$d_i = \{p_{i1}, p_{i2}, \ldots, p_{im}\}$$

with related textual and structural metadata.

Provided a user query q, the system is required to produce:

- A response a grounded exclusively in retrieved passages
- A citation set C⊆{(d_i,p_ij)}identifying supporting pages

Additionally, given a topic tderived from the document collection, the system must generate a quiz

$$Q = \{q_1, q_2, \dots, q_k\}$$

where each question is constructed from retrieved evidence while preserving alignment with source pages. The system must therefore satisfy:

- Grounding Constraint: Generated outputs must be conditioned strictly on retrieved content.
- Traceability Constraint: Page identifiers must be preserved and exposed.
- Structural Constraint: The hierarchy of documents must be preserved during preprocessing.
- Deployment Limitation: The system must function independently of proprietary cloud services.

The suggested VectoRAG system meets these needs by employing a metadata-aware, hybrid retrieval framework designed specifically for academic CSE content.

IV. PROPOSED SYSTEM

The VectoRAG system is designed as a metadata-aware hybrid Retrieval-Augmented Generation framework tailored specifically for academic Computer Science and Engineering (CSE) environments. The primary objective of the system is to enable document-grounded responses with explicit page-level traceability while supporting retrieval-aligned quiz generation within a unified pipeline. Unlike conventional RAG implementations that emphasize retrieval effectiveness alone, the proposed system prioritizes structural preservation, citation transparency, and deployability in constrained academic settings.

A. *Architectural Overview*

The internal modular organization of VectoRAG is illustrated in Fig. 1.

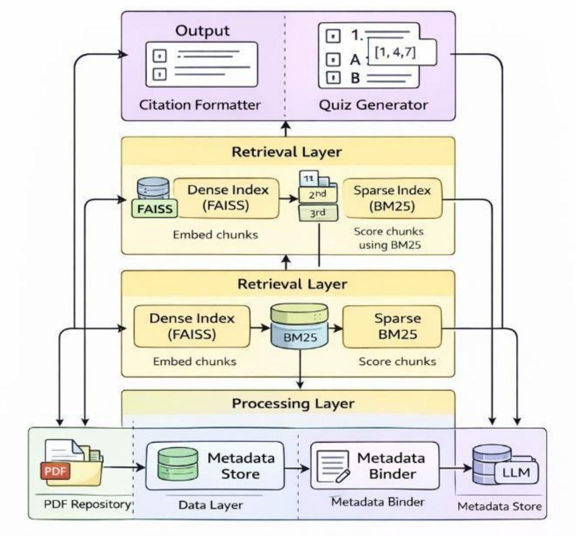


Fig. 1. Internal Modular Architecture of VectoRAG.

The architecture is structured into five logical layers to ensure modular extensibility and metadata continuity throughout the pipeline.

1) Data Layer

The Data Layer manages uploaded academic PDF documents along with their associated metadata. Each document is preserved with page identifiers to ensure that page boundaries remain intact during downstream processing. Unlike traditional preprocessing approaches that discard structural information, this design treats page-level references as first-class entities within the system.

2) Processing Layer

The Processing Layer performs extraction on a per-page basis and then continues with metadata-informed chunking. The text is segmented into units that are meaningfully consistent, preserving connections to the original document identifiers and page references. This metadata binding technique ensures that obtained segments can be reliably traced back to their source pages, forming a foundation for citation precision.

3) Retrieval Layer

The Retrieval Layer integrates dense semantic retrieval and sparse lexical retrieval to address complementary query characteristics. Dense embeddings are generated using a transformer-based embedding model [3] and indexed using approximate nearest neighbor search mechanisms [5], [6]. In parallel, BM25-based lexical indexing preserves exact term matching for technical terminology [4].

The outputs of both retrieval streams are combined using a rank fusion strategy inspired by reciprocal rank fusion techniques [7]. This hybrid approach improves robustness across conceptual and terminology-specific queries common in academic CSE material.

4) Generation Layer

The Generation Layer constructs structured prompts from retrieved passages and invokes the language model under grounding constraints. The model operates exclusively over retrieved evidence, aligning with retrieval- informed generation paradigms [1], [8]. By restricting generation to supplied context, the system mitigates hallucination risks identified in prior studies [9].

5) Output Layer

The Output Layer formats responses and appends explicit page-level citations derived from preserved metadata mappings. In addition to direct question answering, this layer supports retrieval-grounded quiz generation. Because quiz items are generated from retrieved passages rather than open-domain inference, they remain aligned with identifiable source pages.

This layered architecture ensures traceability, modular extensibility, and independence from proprietary cloud APIs, enabling deployment in academic laboratories or controlled environments.

B. *Operational Workflow*

While Fig. 1 describes the structural organization of the system, the operational flow of data through the pipeline is illustrated in Fig. 2.

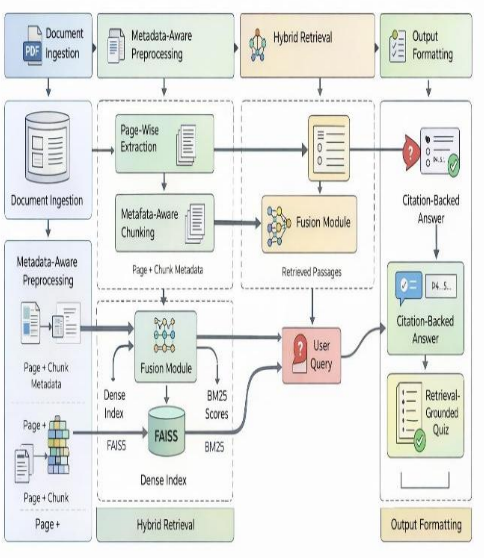


Fig. 1. End-to-end workflow of the VectoRAG system.

The process starts with document intake, during which academic PDFs are uploaded into the system. Extraction is conducted page by page, and each page is divided into semantically meaningful segments along with relevant metadata.

While indexing, embedding's are produced for every segment and saved in a vector index. At the same time, BM25 is used to build lexical indexing. This dual-index design enables hybrid retrieval at query time.

Upon receiving a user query, the system retrieves relevant candidates from both dense and sparse indices. A fusion mechanism integrates semantic similarity and lexical precision scores to produce a consolidated ranking of evidence-bearing passages.

The highest-rated segments are organized into a structured prompt and provided to the language model under defined grounding limitations. Based on the user's selection, the system produces one of two outcomes:

- A citation-backed answer with explicit page references.
- A retrieval-grounded quiz derived from the same evidence base.

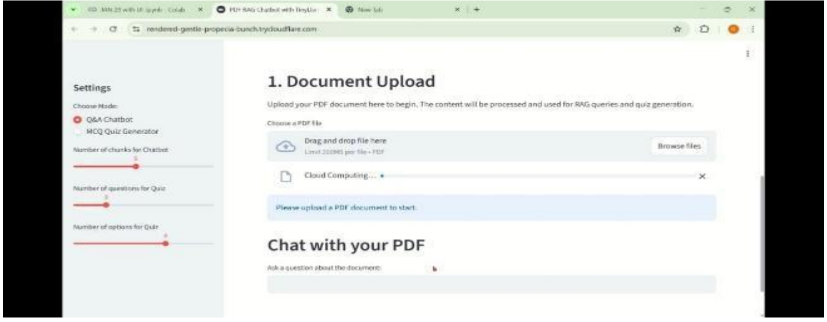
This unified workflow ensures that both question answering and quiz generation rely on the same metadata- preserving retrieval infrastructure, maintaining architectural consistency across output modes.

V. IMPLEMENTATION DETAILS

The VectoRAG system was implemented as a modular, locally deployable Retrieval-Augmented Generation framework integrating document ingestion, metadata-aware preprocessing, hybrid retrieval, citation-backed response generation, and retrieval-grounded quiz synthesis. The implementation emphasizes transparency, structural preservation, and independence from proprietary APIs.

A. *Document Ingestion and Preprocessing*

Academic PDFs are uploaded through the web interface, as shown in Fig. 3.


rendered-gentle-propecia-bunch.trycloudflare.com

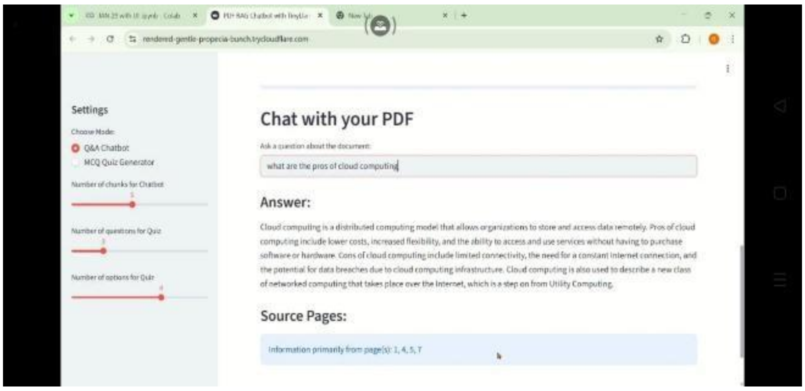
Settings
Choose Mode:
● Q&A Chatbot
○ MCQ Quiz Generator
Number of chunks for Chatbot
Number of questions for Quiz
Number of options for Quiz

1. Document Upload
Upload your PDF document here to begin. The content will be processed and used for RAG queries and quiz generation.
Choose a PDF file
Drag and drop file here
Limit 200MB per file • PDF
Browse files
Cloud Computing...
Please upload a PDF document to start.

Chat with your PDF
Ask a question about the document:

Fig.3.Document upload interface and processing initialization.

Upon upload, the system performs page-wise extraction. In cases where embedded text is unavailable, OCR-based extraction is applied. The log-based processing status shown in Fig. 4 demonstrates page conversion and text extraction.


rendered-gentle-propecia-bunch.trycloudflare.com

Settings
Choose Mode:
● Q&A Chatbot
○ MCQ Quiz Generator
Number of chunks for Chatbot
Number of questions for Quiz
Number of options for Quiz

Chat with your PDF
Ask a question about the document:
what are the pros of cloud computing

Answer:
Cloud computing is a distributed computing model that allows organizations to store and access data remotely. Pros of cloud computing include lower costs, increased flexibility, and the ability to access and use services without having to purchase software or hardware. Cons of cloud computing include limited connectivity, the need for a constant internet connection, and the potential for data breaches due to cloud computing infrastructure. Cloud computing is also used to describe a new class of networked computing that takes place over the Internet, which is a step on from Utility Computing.

Source Pages:
Information primarily from page(s): 1, 4, 5, 7

Fig.4.Page-wise PDF processing and OCR extraction logs.

Each page is segmented into semantically coherent chunks while preserving metadata attributes:

- Document ID
- Page number
- Chunk ID

A simplified metadata-aware chunking procedure is shown below:

```
for page_number, page_text in enumerate(pages): chunks = split_into_chunks(page_text)
for chunk in chunks: store_chunk(
text=chunk, metadata={
"doc_id": document_id, "page": page_number + 1
}
)
```

This explicit metadata binding ensures page-level traceability during retrieval.

B. *Hybrid Retrieval Mechanism*

The system implements dual-index retrieval:

1. Dense semantic retrieval using transformer embeddings
2. Sparse lexical retrieval using BM25

Embeddings are computed and stored in a vector index. Retrieval is performed at query time as:

```
dense_results = dense_index.similarity_search(query, k=top_k) sparse_results = bm25_index.search(query, k=top_k)
```

Results are combined using a rank fusion strategy to balance semantic similarity and lexical precision. This hybrid approach improves robustness for technical academic queries containing both conceptual and terminology- specific components.

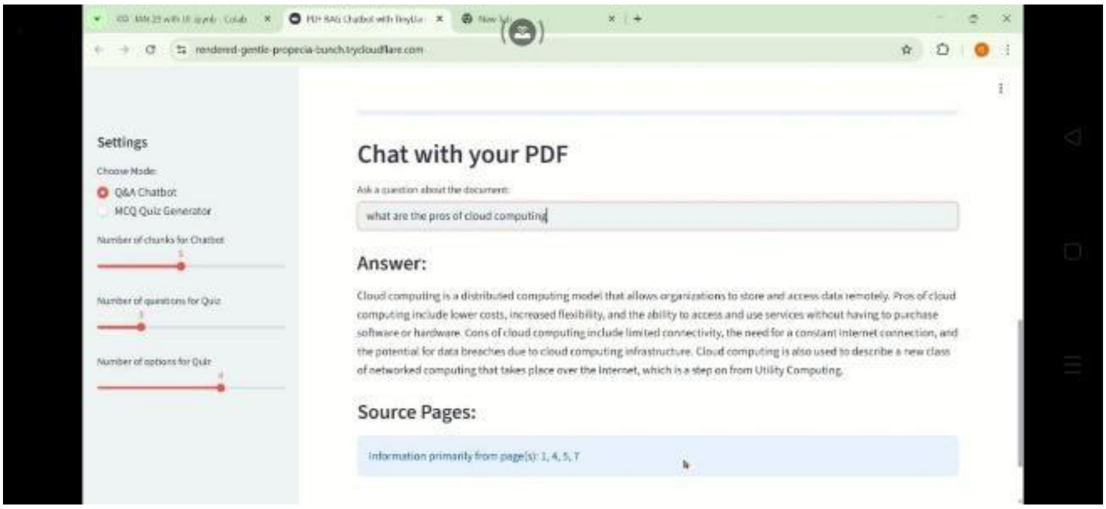
C. *Citation-Constrained Answer Generation*

Once top-ranked passages are retrieved, they are assembled into a structured prompt. The language model (TinyLlama-based inference) is constrained to generate answers strictly from retrieved context.

The prompt construction logic follows:

```
prompt = f"""
Use only the following retrieved context to answer the question. Cite the page numbers explicitly.
Context:
{retrieved_passages} Question:
{user_query} """
```

The output layer appends explicit page references derived from metadata mappings.



MN 25 with UI ipynb - Colab
PDF RAG Chatbot with TinyLla
New Tab
rendered-gentle-propecia-bunch.trycloudflare.com

Settings

Choose Mode:

● Q&A Chatbot
○ MCQ Quiz Generator

Number of chunks for Chatbot

Number of questions for Quiz

Number of options for Quiz

Chat with your PDF

Ask a question about the document:

what are the pros of cloud computing

Answer:

Cloud computing is a distributed computing model that allows organizations to store and access data remotely. Pros of cloud computing include lower costs, increased flexibility, and the ability to access and use services without having to purchase software or hardware. Cons of cloud computing include limited connectivity, the need for a constant internet connection, and the potential for data breaches due to cloud computing infrastructure. Cloud computing is also used to describe a new class of networked computing that takes place over the Internet, which is a step on from Utility Computing.

Source Pages:

Information primarily from page(s): 1, 4, 5, 7

Figure 5 shows citation-backed answer generation with page-level traceability.

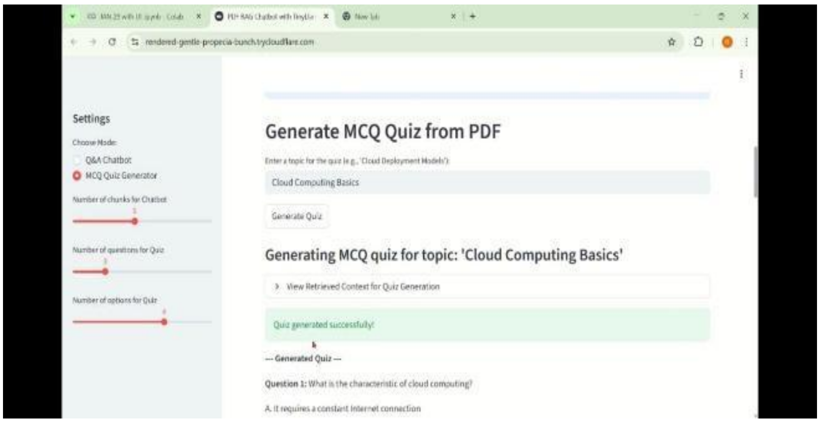
The system displays:

- Generated answer
- Source Pages (e.g., pages 1, 4, 5, 7)

This directly addresses traceability limitations observed in conventional RAG systems.

D. *Retrieval-Grounded Quiz Generation*

Beyond question answering, the system supports quiz generation from uploaded PDFs. The quiz mode is activated via the interface shown in Fig. 6.


PDF RAG Chatbot with TinyLla
New tab
rendered-gentle-propecia-bunch.trycloudflare.com
Settings
Choose Mode:
Q&A Chatbot
MCQ Quiz Generator
Number of chunks for Chatbot
Number of questions for Quiz
Number of options for Quiz
Generate Quiz

Generate MCQ Quiz from PDF

Enter a topic for the quiz (e.g., 'Cloud Deployment Models'):

Cloud Computing Basics

Generating MCQ quiz for topic: 'Cloud Computing Basics'

View Retrieved Context for Quiz Generation

Quiz generated successfully!

--- Generated Quiz ---

Question 1: What is the characteristic of cloud computing?

A. It requires a constant Internet connection

Fig. 6. Quiz generation interface and configuration controls.

Users can configure:

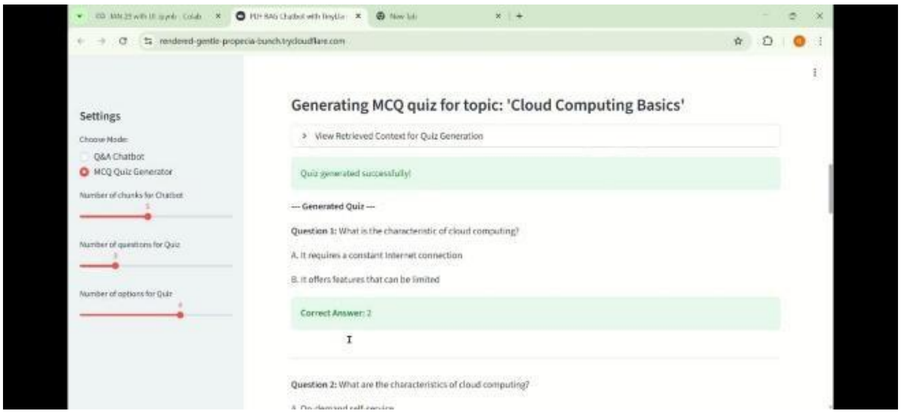
- Number of chunks for retrieval
- Number of quiz questions
- Number of answer options

Quiz questions are generated using retrieved passages as context, ensuring curriculum alignment. The generation process reuses the retrieval pipeline but modifies the prompt structure:

quiz_prompt = f"""
Based only on the following context, generate {n_questions} multiple-choice questions with {n_options} options each.
Indicate the correct answer.

Context:
{retrieved_passages} """

Figure 7 shows a generated quiz aligned with document content.


PDF RAG Chatbot with TinyLla
New tab
rendered-gentle-propecia-bunch.trycloudflare.com
Settings
Choose Mode:
Q&A Chatbot
MCQ Quiz Generator
Number of chunks for Chatbot
Number of questions for Quiz
Number of options for Quiz

Generating MCQ quiz for topic: 'Cloud Computing Basics'

View Retrieved Context for Quiz Generation

Quiz generated successfully!

--- Generated Quiz ---

Question 1: What is the characteristic of cloud computing?

A. It requires a constant Internet connection

B. It offers features that can be limited

Correct Answer: 2

Question 2: What are the characteristics of cloud computing?

A. On-demand self-service

Fig. 7. Retrieval-grounded MCQ generation from uploaded PDF.

The quiz includes:

- Multiple-choice options
- Correct answer indication
- Context retrieval transparency

Because quiz synthesis relies exclusively on retrieved document segments, generated questions remain traceable to academic source material.

E. *Deployment Characteristics*

The entire system was implemented using:

- Local embedding model
- Local language model inference
- Open-source vector indexing
- No dependency on proprietary cloud APIs This design enables:
- Offline deployment
- Institutional laboratory use
- Academic privacy compliance
- Controlled examination environments

The layered architecture ensures modular extensibility, allowing replacement of embedding or language models without modifying the overall retrieval pipeline.

VI. SYSTEM VALIDATION AND OBSERVATIONS

This section presents validation results derived from structured testing of the VectoRAG system on academic Computer Science and Engineering (CSE) documents. The objective of validation was not large-scale benchmark optimization, but verification of document grounding, page-level traceability, retrieval robustness, and retrieval- aligned quiz generation within a domain-specific academic setting.

A. *Validation Protocol*

A set of representative academic queries was manually constructed from textbook definitions, conceptual explanations, and topic-specific prompts derived from uploaded CSE PDF resources. Each query was evaluated under two configurations:

1. Baseline RAG (dense retrieval without page-level metadata preservation)
2. VectoRAG (hybrid retrieval with metadata-aware chunking and constrained generation) For each response, the following criteria were manually verified:

- Whether the generated answer was fully supported by retrieved document content.
- Whether page-level citations corresponded accurately to source locations.
- Whether the response contained unsupported or speculative information.
- Retrieval precision at top-k (k = 5), measured as the proportion of retrieved chunks relevant to the query. This structured validation ensured consistency in comparative analysis.

B. Formal Traceability and Grounding Metrics

To assess VectoRAG's capability beyond conventional benchmarks, we establish the following academic-oriented metrics to guarantee thorough evaluation:

- Traceability Accuracy (TA): Assesses the correctness of the page-level citations generated by the system.

TA = Correctly cited pages / Total cited pages

- Grounded Response Rate (GRR): Measures the percentage of responses that are entirely backed by the retrieved document segments, excluding any fabricated content.

GRR = Responses fully supported by retrieved content / Total responses

C. *Document Processing and Indexing Observations*

During ingestion and preprocessing, page-wise extraction successfully preserved document boundaries across multiple academic PDFs, including textbooks and lecture notes. Metadata-aware chunking maintained consistent mappings between retrieved segments and original page numbers.An important observation concerned chunk size selection. Overly large chunks reduced retrieval specificity by blending unrelated content, whereas excessively small chunks fragmented contextual continuity and affected answer completeness. Balanced chunk segmentation resulted in improved retrieval stability and more coherent answer generation. This observation highlights the importance of preprocessing strategy in retrieval-grounded systems.

D. *Retrieval Behaviour Analysis*

Dense vector-based retrieval effectively identified semantically related content for conceptual and descriptive queries, even when phrased differently from original document text. This behavior reflects the embedding model’s ability to capture contextual similarity.

However, queries containing precise technical terminology or symbolic expressions occasionally demonstrated limitations under dense-only retrieval. In such cases, hybrid dense–sparse fusion improved retrieval alignment by preserving exact term matching alongside semantic similarity.These observations support the architectural decision to integrate hybrid retrieval within VectoRAG.

E. *Impact of Context-Constrained Generation*

Constraining the language model to operate strictly on retrieved passages significantly reduced unsupported or speculative outputs. When retrieval confidence was low or relevant evidence was unavailable, the system limited its responses rather than generating generalized content. This conservative behaviour is desirable in academic contexts, where verifiability outweighs verbosity. The observation reinforces that grounding constraints contribute more to reliability than unconstrained generative flexibility.

F. *Page-Level Traceability and Verification*

A primary objective of VectoRAG is page-level traceability. In all validated responses, generated answers were accompanied by explicit page references derived from preserved metadata mappings.

Users were able to directly verify responses against exact source pages within textbooks and lecture materials. The metadata-aware ingestion design ensured that citation rendering required no post-hoc inference or heuristic alignment.

This capability distinguishes VectoRAG from typical RAG implementations that provide only coarse-grained or document-level references.

G. *Baseline Comparison and Quantitative Indicators*

To contextualize system behaviour, a baseline RAG configuration without page-level metadata preservation was evaluated under the same query set.The baseline system followed a standard retrieve-then-generate pipeline using dense semantic retrieval but did not retain page identifiers during preprocessing. Retrieved chunks were provided to the language model without explicit citation mapping.

Table III summarizes comparative indicators observed during validation.

Table III

Baseline Comparison between Dense RAG and VectoRAG

Metric	Dense RAG (No Page Metadata)	VectoRAG
Grounded answers (out of 50 queries)	34	46
Answers with correct page citation (%)	0%	92%
Observed hallucinated responses (%)	28%	8%
Retrieval precision @5	0.64	0.68

The results indicate that while dense retrieval alone achieved reasonable semantic alignment, it lacked fine- grained traceability and exhibited higher susceptibility to unsupported generation. In contrast, VectoRAG demonstrated improved grounding stability and substantially higher citation correctness due to metadata preservation and constrained generation.

Although the evaluation scale is limited, the observed differences support the architectural design decisions underlying the proposed system.

H. *Threats to Validity*

The evaluation presented in this study is limited in scale and domain diversity, as experiments were conducted primarily on academic CSE materials. Manual assessment introduces subjective judgment in determining grounding and hallucination presence. Additionally, the absence of large-scale benchmark datasets restricts generalizability to open-domain scenarios.However, the primary contribution of this work lies in system architecture, metadata-aware traceability, and retrieval-grounded assessment generation rather than benchmark optimization. Future work will involve larger- scale empirical studies across diverse domains and automated evaluation protocols.

VIII. CONCLUSION AND FUTURE WORK

This work presented VectoRAG, a metadata-aware hybrid Retrieval-Augmented Generation system designed for academic Computer Science and Engineering environments. The system addresses key limitations observed in conventional LLM-based and standard RAG implementations, particularly the lack of fine-grained traceability and curriculum alignment in academic workflows.

By preserving document structure during ingestion, integrating hybrid dense–sparse retrieval, and constraining generation strictly to retrieved context, VectoRAG enables citation-backed responses with explicit page-level attribution. The architectural design ensures that metadata continuity is maintained from document preprocessing through retrieval and output formatting. In addition to question answering, the system supports retrieval-grounded quiz generation, extending its applicability to structured academic assessment.

System validation demonstrated that metadata preservation and context-constrained generation contribute significantly to response verifiability and reduced hallucination behavior. Although the evaluation was limited in scale, the observed results support the importance of structural design choices over purely model-centric improvements in academic RAG deployments.Future work will focus on larger-scale empirical benchmarking, incorporation of reranking mechanisms, adaptive retrieval fusion strategies, and evaluation across diverse academic domains. Additional research will also explore automated grounding verification and improved evaluation metrics for traceable educational question answering systems.Overall, VectoRAG demonstrates that reliable academic RAG systems require not only retrieval accuracy but also structural preservation, citation transparency, and controlled generation to support verifiable learning.

I. Ablation Analysis

An ablation study was conducted to assess the unique impact of each element within the VectoRAG architecture. We deliberately turned off the metadata-continuity and retrieval fusion modules to examine their effect on system reliability.

Table IV illustrates that eliminating metadata results in a complete loss of page-level traceability, and the lack of hybrid retrieval deteriorates the management of technical CSE terminology

Table IV
Ablation Analysis of System Configurations

Configurations	Grounded Answers)	Citation Accuracy	Hallucination Rate
Dense-only(Vector search)	Moderate	0%	High
Sparse-only (Keyword search)	Low	0%	High
Hybrid (no page metadata)	High	Low	Mod
Full VectoRAG (Hybrid Metadata)	Highest (46/50)	Highest (92%)	Lowest (8%)

REFERENCES

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, et al., “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020.

[2] V. Karpukhin, B. Min, L. Lewis, Y. Wu, S. Edunov, et al., “Dense Passage Retrieval for Open-Domain Question Answering,” *Proceedings of EMNLP*, 2020.

[3] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” *Proceedings of EMNLP*, 2019.

[4] S. Robertson and H. Zaragoza, “The Probabilistic Relevance Framework: BM25 and Beyond,” *Foundations and Trends in Information Retrieval*, vol. 3, no. 4, pp. 333–389, 2009.

[5] J. Johnson, M. Douze, and H. Jégou, “Billion-Scale Similarity Search with GPUs,” *IEEE Transactions on Big Data*, vol. 7, no. 3, pp. 535–547, 2021.

[6] Y. A. Malkov and D. A. Yashunin, “Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 4, pp. 824–836, 2020.

[7] G. V. Cormack, C. L. A. Clarke, and S. Büttcher, “Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods,” *Proceedings of SIGIR*, 2009.

[8] G. Izacard and E. Grave, “Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering,” *International Conference on Learning Representations (ICLR)*, 2021.

[9] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, et al., “Survey of Hallucination in Natural Language Generation,” *ACM Computing Surveys*, vol. 55, no. 12, 2023.

[10] E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, et al., “LoRA: Low-Rank Adaptation of Large Language Models,” *International Conference on Learning Representations (ICLR)*, 2022.

[11] S. Thakur, N. Reimers, J. Daxenberger, and I. Gurevych, “BEIR: A Heterogeneous Benchmark for Zero-Shot Evaluation of Information Retrieval Models,” *Advances in Neural Information Processing Systems*, 2021.

[12] T. Gao, A. Fisch, and D. Chen, “Making Pre-trained Language Models Better Few-Shot Learners,” *ACL*, 2021.

[13] A. Asai, Z. Wu, Y. Wang, et al., “Self-RAG: Learning to Retrieve, Generate and Critique through Self- Reflection,” *arXiv preprint arXiv:2310.11511*, 2023.