Grenze International Journal of Engineering and Technology, June Issue

A Hybrid Retrieval-Augmented Generation Framework with Role-based Access Control for Enterprise Knowledge Management

Krisha Chikka[1], Yash Chavan[2], Anjali Gupta[3], Sarwadeep Dhaval[4], Megha Trivedi[5] and Anil Hingmire[6]
[1-6]Vidyavardhini's College of Engineering and Technology, Department of Computer Engineering, Vasai, India
Email: chikkakrisha@gmail.com, yashchavan4628@gmail.com, anjaligupta92958@gmail.com, sarwadeepdhaval@gmail.com, megha.trivedi@vcet.edu.in, anil.hingmire@vcet.edu.in

***Abstract*— Enterprise knowledge systems operate over heterogeneous and distributed data sources, making accurate and secure retrieval challenging. This paper presents a multimodal Retrieval-Augmented Generation (RAG) system that integrates hybrid semantic retrieval, structured query execution, role-based access control, and blockchain-based audit logging. The system introduces a two-dimensional query classification mechanism for joint routing across structured and unstructured pipelines, along with a weighted reranking strategy combining semantic similarity, keyword overlap, and source reliability. Evaluation on a multi-format enterprise dataset demonstrates 87.5% top-5 retrieval accuracy and 92.1% SQL correctness, outperforming a Vanilla RAG baseline. The current evaluation focuses on retrieval and structured query performance, while end-to-end response generation is integrated at the pipeline level and will be evaluated in future work.**

***Index Terms*— Enterprise Knowledge Management, Retrieval-Augmented Generation, Multimodal Retrieval, Hybrid Access Control, Blockchain Auditing and Semantic Search.**

I. INTRODUCTION

Enterprise knowledge is distributed across heterogeneous systems, including cloud storage, relational databases, email platforms, and document repositories, and exists in both structured and unstructured formats. Retrieving relevant information across these sources remains challenging due to differences in data representation, schema, and access mechanisms. Traditional keyword-based retrieval methods fail to capture semantic relationships across such heterogeneous sources, limiting their effectiveness in real-world enterprise environments. RAG improves retrieval by grounding responses in relevant documents [1], but existing implementations typically assume homogeneous and preprocessed data, which does not reflect practical enterprise settings [8].

Three limitations persist in enterprise retrieval systems. First, large language models often produce unreliable outputs when applied to domain-specific data, leading to hallucinations [1], [7]. Second, existing RAG pipelines do not enforce strict access control, potentially exposing sensitive information [3]. Third, most systems lack built-in mechanisms for auditability and compliance, which are essential in regulated environments such as General Data Protection Regulation (GDPR) and Health Insurance Portability and Accountability Act (HIPAA) [4].

Grenze ID: 01.GIJET.12.2.595

A. Research Gap Analysis

Current solutions provide partial capabilities but fail to address enterprise requirements holistically. While semantic retrieval improves search quality, existing systems do not simultaneously support multimodal data ingestion, query-aware routing across structured and unstructured sources, and tamper-proof audit logging [3], [4], [9]. This gap highlights the need for an integrated architecture where retrieval, access control, and auditability are treated as core system components rather than post-deployment additions.

B. Primary Objectives of Proposed system

- Enable unified retrieval across structured and unstructured enterprise data
- To improve response reliability by grounding generated outputs in retrieved evidence from multiple data sources. [1], [15]
- Enforce secure access through fine-grained permission control [7]
- Ensure transparency via immutable audit logging [4]
- Provide an interactive interface for query-driven knowledge access with structured and visual outputs.

The system processes enterprise data through a multimodal ingestion pipeline, generates semantic embeddings, and retrieves relevant information using a hybrid RAG approach. Structured and unstructured retrieval pipelines operate in parallel and are combined through a reranking mechanism. Security is enforced via access control, and are recorded through an immutable audit layer [3], [4].

II. PROPOSED SYSTEM

A. System Architecture

The system integrates multimodal retrieval, access control, and audit logging within a unified pipeline. It consists of three layers: (i) ingestion, which processes heterogeneous enterprise data and generates embeddings while preserving structured schema; (ii) retrieval and reasoning, which performs hybrid retrieval across vector search and schema-aware query execution; and (iii) audit, which records all for traceability.

Queries are processed through a hybrid pipeline where structured and unstructured retrieval paths operate in parallel and are merged via a reranking module based on semantic similarity, keyword overlap, and source reliability. Access control is enforced during query execution to prevent unauthorized retrieval. All interactions are logged using a blockchain-backed mechanism to ensure auditability compliance [3], [4].

B. Core Features of the System

- Multimodal Data Ingestion: Connects to cloud and internal enterprise systems for syncing of documents, images, spreadsheets, PDFs and multimedia [1],[5].
- Smart Semantic Search: Embedding-based search enables context-aware retrieval beyond keyword matching [2],[3].
- RAG Answer Generation: Combines dense, sparse retrieval to produce grounded, domain-specific responses for natural language queries [4][6].
- Hybrid Access Control: Enforces fine-grained permissions based on roles, data sensitivity, query context, and past access patterns [7].

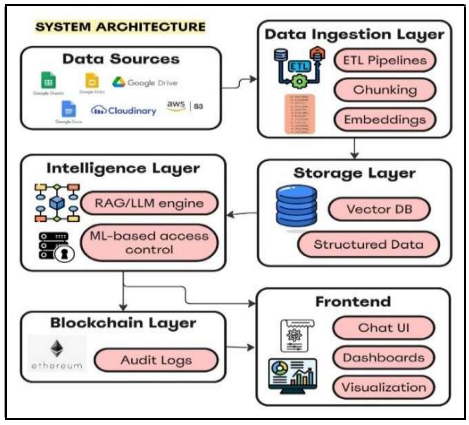

Figure 1. Technical System architecture diagram illustrating the flow

- Audit Logging: Logs all queries and system interactions using Hyperledger or Polygon for tamper-proof records and compliance [8].
- Interactive Conversational UI & Visualizations: Chat-based interface with dynamic tables, charts, graphs, and document previews for actionable insights [6][9].
- Hybrid Retrieval and Query Translation: Queries are dynamically routed across structured and unstructured pipelines. Structured queries are translated into SQL/NoSQL, while unstructured queries trigger semantic retrieval, with both results combined when required [5], [10].
- Admin & Compliance Management: Provides conversational querying and system monitoring [2], [9], [11].

The system uses React.js and TailwindCSS for the frontend, with FastAPI and Node.js services supported by Redis and Celery for backend processing. Data is managed across PostgreSQL, MongoDB, and FAISS/Milvus for vector indexing. Sentence-BERT is used for embeddings, while LLMs handle response generation. Security is enforced via JWT, and audit logging is implemented using Hyperledger Fabric. Deployment is containerized using Docker and Kubernetes.

III. Literature Review

A. Introduction to Enterprise Knowledge Management

Enterprise Knowledge Management (EKM) focuses on organizing and retrieving knowledge across heterogeneous environments. Modern systems handle both structured and unstructured data distributed across multiple platforms, making unified retrieval challenging [2], [5]. Traditional approaches rely on keyword or metadata-based indexing, which fails to capture semantic context [1]. Recent methods incorporate semantic embeddings to improve retrieval quality [1], [8], but often assume centralized, preprocessed data and do not adequately address real-time integration, multimodal inputs, or enterprise constraints such as access control and compliance [2], [9], [12], [13].

B. Existing Knowledge Retrieval Systems

Conventional enterprise retrieval systems are optimized for structured or document-centric workflows, but their performance degrades when applied to multimodal and cross-platform data [2], [6]. Keyword-based and metadata-driven approaches fail to generalize across formats such as scanned documents, images, and mixed data sources [5], [6].

Retrieval-Augmented Generation (RAG) integrates document retrieval with language model generation, enabling responses to be grounded in relevant data sources [1], [8], [15]. This approach has been shown to reduce hallucinations and improve performance in tasks such as question answering and summarization [16], [11]. Comparative studies indicate that RAG systems provide better accuracy than standalone language models, particularly in domain-specific applications [15], [16].

C. Multimodal and AI-Powered Enterprise Solutions for Complex Enterprise Data

Recent work has explored multimodal retrieval by combining text, visual, and structured data representations within a unified framework. Systems such as eSapiens demonstrate improved performance on complex documents containing tables, scanned content, and mixed formats [9]. Hybrid retrieval approaches further extend this by integrating structured query execution with semantic search, enabling more comprehensive query handling [10], [14].

D. Secure and Compliant Knowledge Systems

Security and compliance have become critical considerations in enterprise knowledge systems due to the sensitive nature of organizational data [3], [4], [7]. Recent approaches incorporate access control mechanisms within retrieval pipelines to restrict data exposure based on user roles and context [3], [7]. Blockchain-based audit logging has also been proposed to ensure tamper-proof tracking of system interactions, supporting regulatory requirements such as GDPR and HIPAA [4].

IV. Methodology

The proposed system integrates retrieval-augmented generation with direct database querying to support hybrid retrieval across enterprise data. The pipeline is organized into three phases:

- Data Ingestion and Preprocessing
- Query Understanding and Hybrid Retrieval
- Verification and Audit Logging

The workflow (Figure 2.) captures the end-to-end flow from multi-source data ingestion to response generation. Data from cloud storage, databases, and document repositories is processed into a unified retrieval pipeline, enabling queries to be executed across heterogeneous sources.

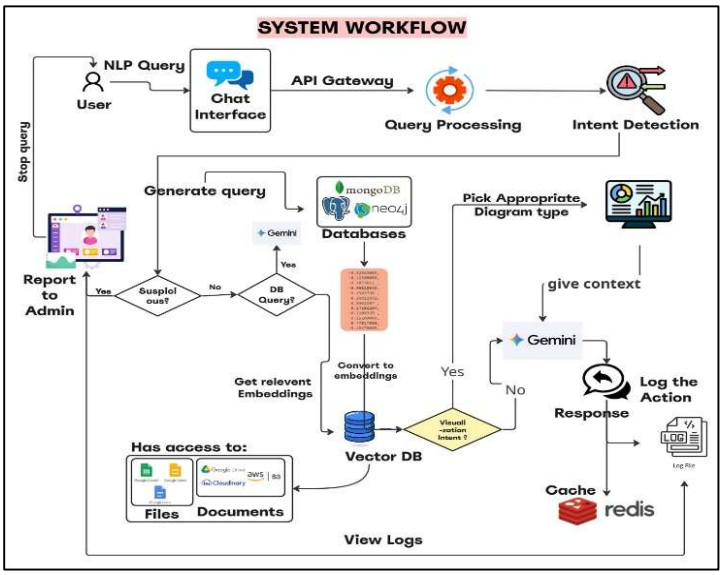

Figure 2. System workflow depicting the integration pipeline from multi-source data ingestion

Phase 1 – Data Ingestion & Preprocessing

Enterprise data is collected from structured databases, unstructured repositories, and external cloud platforms. Each source undergoes preprocessing for normalization and metadata extraction. (as shown in Figure 3.). For scanned documents, Optical Character Recognition (OCR) converts content into machine-readable text. A Schema Snapshot captures database tables, fields, relationships, and collection structures, providing a semantic vocabulary for query interpretation.

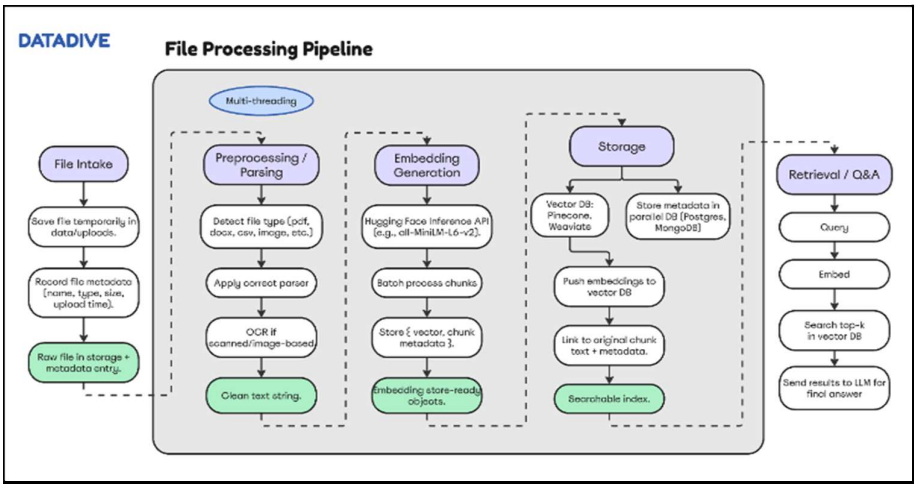

Figure 3. File processing pipeline: ingestion, OCR, embedding, and hybrid indexing.

Following preprocessing, data is transformed into dense vector embeddings using transformer-based models such as BERT and Sentence-BERT. Embeddings and metadata are indexed in a scalable vector database (FAISS/Milvus) to facilitate semantic search. Structured data remains accessible through SQL/NoSQL connectors, enabling hybrid queries across structured and unstructured sources. Structured data is not flattened into text prior to indexing, as this leads to loss of relational semantics and reduced query accuracy in hybrid retrieval scenarios.

Phase 2 – Query Understanding & Hybrid Retrieval

User queries are routed through a secure API Gateway, where they are authenticated using JWT and validated against role-based access policies. The system applies a two-dimensional classification strategy: (i) intent type (structured, unstructured, hybrid) and (ii) response type (summary, data, visualization, conversational). This classification determines both the execution path and the expected output format.

Algorithm 1: Query Routing

Input: Query Q, user role R

Output: execution path P

- Classify Q into intent type T
- Validate access permissions for R
- Route execution:
  - If T = Structured → SQL Engine
  - If T = Unstructured → Vector Store
  - Else → Hybrid Pipeline
- Return execution path P

For structured queries, schema-aware SQL is generated using semantic matching between query terms and schema embeddings. For unstructured queries, relevant document chunks are retrieved from the vector store using cosine similarity (1):

$$\text{similarity}(A, B) = \frac{A \cdot B}{\|A\|\|B\|} \quad (1)$$

In hybrid scenarios, results from both pipelines are combined and reranked using a weighted scoring function:

$$\text{FinalScore}(r_i) = \alpha \cdot \text{SemanticScore} + \beta \cdot \text{KeywordScore} + \gamma \cdot \text{SourceScore} \quad (2)$$

where α = 0.6, β = 0.2, and γ = 0.2. This formulation (2) balances semantic relevance, keyword overlap, and source reliability.

Reranking plays a critical role in hybrid retrieval, as similarity-based methods alone often return results that are relevant in isolation but incomplete in context. The reranked outputs are passed to the RAG pipeline for response synthesis, ensuring that generated answers remain grounded in retrieved evidence.

Phase 3 – Verification & Audit Logging

A verification layer cross-checks generated responses against retrieved evidence, assigning a confidence score. Low-confidence responses trigger re-retrieval.

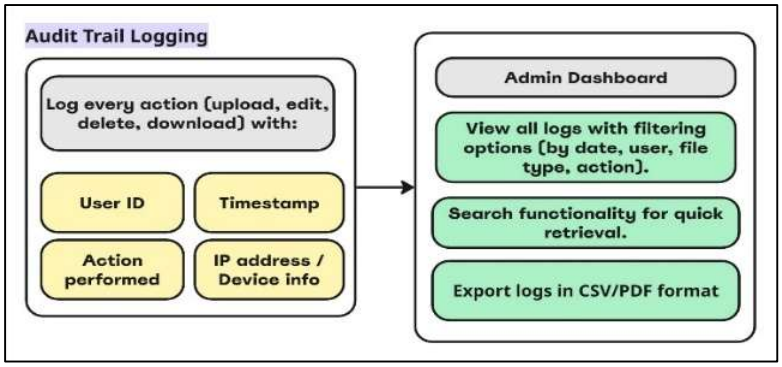

Figure 4. Blockchain audit trail: logging queries, access decisions, and verification outcomes for compliance.

All queries, results, and access decisions are logged in a blockchain-enabled audit trail (as illustrated in Figure 4.). Each record captures user and tenant IDs, executed SQL statements, model versions, verification results, and confidence scores. This ensures tamper-proof traceability and compliance with enterprise regulations.

IV. METHODOLOGY

A. Evaluation Setup

Evaluation was conducted on a curated enterprise-like dataset comprising 47 documents across 12 file types, including PDF, DOCX, PPTX, XLSX, CSV, JSON, images, and email records. The dataset combines publicly available enterprise-style content with synthetic data simulating HRMS, CRM, and ERP systems.

A test set of 40 queries was constructed with manually verified ground-truth answers, covering structured (14), unstructured (16), and hybrid queries (10). Retrieval accuracy is reported at top-5, where a query is considered correct if the relevant source appears within the top retrieved results. SQL correctness is measured by comparing generated queries against manually validated outputs.

B. Prototype Performance Metrics

Performance benchmarking was conducted on individual subsystems rather than the full end-to-end RAG pipeline. Measurements were taken across multiple runs under both warm and cold cache conditions to reflect realistic deployment behaviour.
Table I summarizes performance across authentication, ingestion, embedding, schema management, and retrieval.

TABLE I. SYSTEM PERFORMANCE RESULTS

Experiment	Dataset/Subsystem	Result	Notes
Authentication Pipeline	/auth/google	~0.9-1.3s	Includes OAuth token issuance
RAG Accuracy	Ingested files (SQL, JSON, DOCX, PDF, PNG, JPEG, etc.)	87.5%	Top 5 retrieval
SQL Correctness	Structured data from ingested files	92.1%	Validated against manually verified queries
File Ingestion (per file)	/ingest/single	2.4-4.9s	File Dependant
Embedding Generation	/generate-embeddings	1.7-2.9s	512–2048 tokens
Snapshot Retrieval (cached)	/api/v1/database/cache	40-85 ms	Redis optimized
Schema Snapshot (cold)	/api/v1/database/		
snapshot	2.2–4.1 s	No cache	

Observations from Table I.:

- Caching reduces schema access latency by nearly two orders of magnitude, making repeated enterprise queries significantly faster.
- Variations in ingestion time are primarily driven by OCR overhead and file size.
- Schema snapshotting confirms interoperability across multiple database types and supports downstream ranking tasks.

C. Performance Comparison with Baselines

To evaluate the contribution of individual components, the system was compared against simplified configurations including Vanilla RAG, removal of schema-aware SQL, reranking, and preprocessing.
Observations from the baseline comparison Table II

TABLE II. BASELINE COMPARISONS

Configuration	RAG Accuracy	SQL Correctness
Full System	87.5%	92.1%
Vanilla RAG	74.2%	-
Without Schema-aware SQL	-	81.5%
Without Reranking	81.3%	92.1%

- The hybrid architecture improves retrieval accuracy by 13.3 percentage points over Vanilla RAG, indicating the importance of combining structured and semantic retrieval.
- Removing reranking reduces performance, highlighting that relevance ordering plays a critical role beyond initial retrieval.
- Removing reranking reduces performance, highlighting that relevance ordering plays a critical role beyond initial retrieval.

D. Scalability Potential, and Practical Impact

The system is implemented using a microservice architecture with Docker-based orchestration, enabling independent scaling of ingestion, embedding, and retrieval services. Redis caching enables sub-100 ms response times for repeated queries, which is critical for enterprise workloads with high query locality.
From a practical perspective, the system reduces fragmentation across enterprise data sources and enables unified access to structured and unstructured knowledge.
These results indicate that the architecture is not only functionally valid but also scalable and deployable in real-world enterprise environments.

V. Results and Evaluation

A. Conclusion

The proposed work presents a hybrid retrieval-augmented generation framework designed for enterprise knowledge management across structured and unstructured data sources. The system integrates semantic retrieval, schema-aware query execution, reranking, and audit logging within a unified pipeline to enable accurate and traceable responses.

Experimental results demonstrate that the proposed approach improves retrieval accuracy to 87.5% (top-5) and SQL correctness to 92.1%, outperforming baseline configurations such as Vanilla RAG. The inclusion of schema-aware SQL and reranking contributes significantly to these gains, while caching mechanisms reduce query latency in repeated access scenarios.

Overall, the results validate that combining structured query execution with semantic retrieval leads to more reliable and context-aware enterprise search compared to standalone approaches. The system provides a scalable and extensible foundation for deploying secure and efficient knowledge retrieval solutions in real-world enterprise environments.

B. Future Scope

Building on the current prototype, several directions are identified for extending the system's capabilities:

- Response Faithfulness and Evaluation: Future work will focus on evaluating response quality, factual consistency, and explainability using frameworks such as RAGAS and automated verification mechanisms.
- Advanced Hybrid Retrieval: Retrieval performance can be improved through cross-encoder reranking and query-adaptive weighting of semantic, keyword, and source-based scores.
- Intelligent Audit Monitoring: Future enhancements may include smart-contract automation and anomaly detection to strengthen compliance tracking and identify suspicious access patterns [4].
- Multimodal Retrieval and Visualization: The framework can be extended to support diagrams, tables, handwritten notes, and multimedia content, along with knowledge-graph-based data exploration.
- Multilingual and Cross-Domain Adaptation: Future versions may incorporate multilingual retrieval and cross-domain adaptation to improve accessibility across global enterprise environments.

References

[1] L.-C. Chen, Y. Zhang, M. Li, and P. Wang, "Application of retrieval-augmented generation for interactive industrial knowledge management via a large language model," Comput. Stand. Interfaces, vol. 94, p. 103995, 2025.

[2] X. Wang, "AI for Optimizing Enterprise Knowledge Management," Int. J. Artif. Intell. Mach. Learn., vol. 4, 2021.

[3] G. Byun, H. Kim, J. Lee, and S. Park, "Secure Multifaceted-RAG for Enterprise: Hybrid Knowledge Retrieval with Security Filtering," arXiv:2504.13425, 2025.

[4] A. Ahmad, "Blockchain-Driven Secure and Transparent Audit Logs," Ph.D. dissertation, Dept. Comput. Sci., Univ. Central Florida, Orlando, FL, USA.

[5] J. Wang et al., "Towards Operationalizing Heterogeneous Data Discovery," arXiv:2504.02059, 2025.

[6] D. W. Feyisa et al., "The Future of Document Indexing: GPT and Donut Revolutionize Table of Content Processing," arXiv:2403.07553, 2024.

[7] M. N. Nobi, M. Gupta, R. Krishnan, M. S. Rana, L. Praharaj, and M. Abdelsalam, "Machine Learning in Access Control: A Taxonomy," in Proc. 30th ACM Symp. Access Control Models Technol. (SACMAT), New York, NY, USA, 2025, pp. 145–156.

[8] N. Chidipothu et al., "Improving Large Language Model (LLM) Performance with Retrieval Augmented Generation (RAG): Development of a Transparent Generative Artificial Intelligence University Support System for Educational Purposes," J. Big Data Artif. Intell., vol. 3, no. 1, 2024.

[9] Z. Shi, L. Wang, L. He, Y. Yang, and T. Shi, "eSapiens: A Real-World NLP Framework for Multimodal Document Understanding and Enterprise Knowledge Processing," arXiv:2506.16768, 2025.

[10] C. Cheerla, "Advancing Retrieval-Augmented Generation for Structured Enterprise and Internal Data," arXiv:2507.12425, 2025.

[11] A. Singh, A. Ehtesham, S. Kumar, and T. T. Khoei, "Agentic Retrieval-Augmented Generation: A Survey on Agentic RAG," arXiv:2501.09136, Feb. 2025.

[12] A. A. Khan, M. T. Hasan, K. K. Kemell, J. Rasku, and P. Abrahamsson, "Developing Retrieval Augmented Generation (RAG)-Based LLM Systems from PDFs: An Experience Report," arXiv:2410.15944, Oct. 2024.

[13] S. Packowski, I. Halilovic, J. Schlotfeldt, T. Smith, "Optimizing and Evaluating Enterprise Retrieval-Augmented Generation (RAG): A Content Design Perspective," in Proc. Int. Conf. Adv. Artif. Intell., Oct. 2024, pp. 162–167.

[14] C. Min, S. Bansal, J. Pan, A. Keshavarzi, R. Mathew, and A. V. Kannan, "Towards Practical GraphRAG: Efficient Knowledge Graph Construction and Hybrid Retrieval at Scale," arXiv:2507.03226, Jul. 2025.

[15] H. S. Patel, P. Mohite, and S. Dhamdhere, “Maximizing RAG Efficiency: A Comparative Analysis of RAG Methods,” in Proc. Int. Conf. Commun., Compute. Ind. 6.0 (C2I6), Bengaluru, India, 2024, pp. 1–6.
[16] A. Patel, R. Kale, and A. Patil, “Evaluation of RAG Metrics for Question Answering in the Telecom Domain,” arXiv:2407.12873, Jul. 2024.