iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.81658

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Offline Retrieval-Agumented Generation for Secure Document Intelligence

P. Pravallika, SK.Rezwana, T. Sowjanya, K. Vyshnavi
Department of Computer Science and Engineering BAPATLAWOMEN'S ENGINEERING COLLEGE Bapatla, India

Abstract: ***Retrieval-Augmented Generation (RAG) has been identified as an effective way to achieve the context accuracy of Large Language Models (LLMs) through the incorporation of knowledge retrieval systems. However, the vast majority of the existingRAG-basedsystemshavereliedoncloudservicestoboost theperformanceofthemodel.Thishasledtosignificantconcerns regardingtheissueofdataprivacyandconfidentiality.Thispaper, therefore, aims to introduce an Offline Retrieval-Augmented Generation model to boost the security and confidentiality of the document intelligence process. This model will process PDF documents offline and utilize the light transformer-based embeddingmodeltocreatethesemanticembeddingandstoreitin the vector database. "Retrieval-Augmented Generation" has proveditselfasanefficientmethodforimprovingtheaccuracyof contextinLarge LanguageModels.However,mostoftheexisting Retrieval-Augmented Generation-based systems are utilizing cloudservicesforimprovingtheefficiencyofthemodel.However, it is causing a major concern regarding the privacy and confidentialityofthesystem. In orderto improvethesecurityand confidentiality of the document intelligence system, this paper focuses on introducing an "Offline Retrieval-Augmented Generation" model. In this method, a PDF document will be processed offline using a lightweight transformer-based embedding model.***

Keywords—Retrieval-AugmentedGeneration(RAG),Offline Artificial Intelligence, Secure Document Intelligence, Large Language Models (LLMs), Vector Databases, FAISS, Local Embedding Models, Privacy-Preserving AI, Air-Gapped Systems, Semantic Search, Hallucination Mitigation, On-Premise AI Deployment.

I. INTRODUCTION

The rapid development of Large Language Models (LLMs) has significantly influenced the development of natural language processing applications like intelligent question answering systems, document summarization systems, and conversationsystems.However,therearecriticallimitations like knowledge cut-off constraints and hallucination problems,andthelackofaccesstodataforLLMs.Therefore, Retrieval-AugmentedGeneration(RAG)wasproposedasan effectivesolutiontotheseproblemsbyutilizingvarious information retrieval systems and developing intelligent question answering systems.

It was identified that the majority of the RAG systems proposed are highly dependent on cloud-based architectures and API services.

Although these architectures and systems are highly efficient and computationally scalable, there are critical limitations like serious data privacy concerns and confidentialityproblems.Forexample,organizationslikethe defense sector, healthcare sector, finance sector, and government organizations cannot afford to transmit their proprietary documents through third-party servers. In addition to this, the architectures and systems proposed are highly vulnerable to data exfiltration problems and network availability problems. To overcome the aforementioned issues, this paper presents a novel Offline Retrieval-Augmented Generation system for Secure Document Intelligence. The proposed system is intended to execute completelyinalocalsystemwithouttheneedtoestablishany internet connection or utilize any external APIs. The system will process the PDF document locally, generate semantic embeddings using a transformer-based model, store the embeddings in a vector database, and retrieve the relevant documentsegmentsduringqueryexecution.Alocallyhosted LargeLanguageModelwillgenerateresponsesbasedonthe retrieved contexts in a way that is strictly limited to the retrieved contexts, thus avoiding hallucinations.

The key contributions of this paper can be summarized as follows:

1) Implementation of a completely offline RAG system that can be executed in an air-gapped system.
2) Incorporation of local embedding generation and vector search to enable secure document retrieval.
3) Implementation of a novel context restriction system to avoid hallucinations.
4) Specification and implementation of the completelyofflinesystemwithoutanyexternal dependencies.
5) Evaluationofthesysteminacontrolled environment.

©IJRASET: All Rights are Reserved

Fromtheproposedframework,itisevidentthatitispossible tohavesecureandprivacy-preservingdocumentintelligence without depending on cloud-based AI services. This study will help in the development of deployable on-premise AI solutions for critical applications.

II. LITERATURE SURVEY

1) Srivastava (2026) has proposed the privacy-first architecture for fully local Retrieval-Augmented Generation (RAG) in secure document intelligence and introduced RAGStack for offline document processing and question answeringwiththehelpoflocalembeddings,FAISS,andon-premise LLMs. The study has also focused on the significance of data sovereignty and reproducibility for air-gapped environments; however, the results have only included internal tests based on CPUs without any benchmarkingandcomparativestudyforsuchsystemsinthe real world.

2) A detailed survey on the topic of "Retrieval-Augmented Generation"hasbeenprovidedbyGenesis(2025).Thework discusses the basic ideas and techniques that go along with the usage of the RAG technique, as well as the basic techniques of integrating the data, as done for the RAG technique.Althoughthesurveyhascoveredthetopicof RAG extensively, it has mainly concentrated on the existing work doneonthetopicofRAG,withouttheactualimplementation of the techniques that go along with the usage of the RAG technique.

3) Another offline model of the Retrieval-Augmented Generation model has also been proposed by Kishore et al., which uses local models such as DeepSeek and Gemma to offer a privacy-oriented system for document interactions. The model has also used local embeddings, vector representations, and the Streamlit interface to offer an internet-independent system. Despite the fact that the model hasfocusedontheflexibleandsafeusageofthemodel,there is a lack of quantitative evaluation of the model, and the performance of the model has not been compared with the existing architectures used in the RAG model.

4) Ali (2025) has also explored the possibility of using the "Retrieval-Augmented Generation" and "Large Language Models" to implement an intelligent query over structured and unstructured data sources. This experiment was carried outbyusingGemini2.0togenerateSQLcodes,andSentence Transformerstoimplementsemanticsearch,whichachieved high precision in translating structured queries and semantic search relevance in documents. Although this experiment is very relevant to the given problem, it only considers cloud-based models without any privacy concerns.

5) Paoletti (2025) proposed the idea of designing an AI-based system of document intelligence for analyzing and creating structured official resolutions in an automatic way throughtheRetrieval-AugmentedGeneration(RAG)model. The model is a combination of the retrieval method and the prompt-basedLLMgenerationfortheaccuracyandfactsin the document. Though the thesis is rich in the evaluation of the quality of the results generated in the retrieval and generation model, the limitation of the thesis is that it is limited to a particular area of law-administrative.

6) In the research paper published by Cheng et al. in 2024, TrojanRAG was introduced by the authors and proved the possibilityofbackdoorattacksthroughRetrieval-Augmented Generation systems on Large Language Models. In the research paper, the joint retrieval and generation attack mechanismwasproposedbyusingoptimizedtriggercontexts and knowledge graph-based hard matching for manipulating the models without compromising retrieval performance. Although critical security risks were identified in RAG systems through the research paper, the complete picture of defensive strategies was not provided by the authors of the research paper.

7) Tyndall et al. (2025) have assessed the viability of the utilization of secure and offline large language models through the utilization of the Retrieval-Augmented Generationmodelonhardwaredevicesthatonlyhave CPUs. TheexperimentwasconductedusingtheLocalGPTmodelto assess the viability of the model of interest in question and answerscenarios.Theexperimentwasaimedatassessingthe viability of the model of interest, but the quality of the text generated was a limitation to the experiment on the summarization scenario. The experiment did not cover the utilization of the model of interest on large datasets, adversarialrobustness,securitythreatmodeling,amongother things, on the offline model of interest.

8) In the study done by Karakurt and Akbulut (2025), the researchers investigated the impact of Retrieval-Augmented Generation and the application of large language models on Enterprise Knowledge Management and documentation automation. This study is underpinned by a literature review methodology. From the study, it is evident that GPT models areusedincombinationwiththeconventionalFAISSmodel, but the practical application of these models presents a huge disconnect. This study seems more analytical than practical.

9) Gilmaryetal.(2025)proposedanintelligentquerysystem basedontheRetrieval-AugmentedGeneration(RAG)model. TheauthorscreatedthemodelusingtheStreamlitlibraryand theOllamaandFAISStoolstoprovideaninteractivetoolfor use in local environments. The paper provides limited quantitative results and does not fully explore the model's potential.

10) A detailed survey on "Knowledge-Oriented Retrieval-Augmented Generation" has been presented in the form of "Chengetal.(2025)."Thesurveydiscussesthetaxonomyof the integration of retrieval and generation, along with the evaluation metrics and the areas of application of the model. The challenges associated with the alignment of the objectives of the retrieval model and the generation model have been identified, along with the latest techniques that incorporate multimodal information into the model. The content of the article is analytical in nature, without any implementation or deployment details.
11) Velamala introduced LocalRAG, an offline multi-PDF question answering system that prioritizes privacy. It incorporates OCR technology and combines dense FAISS andsparseBM25searchwithaquantizedLLMthatperforms CPU-only inference. It focuses on metadata sanitization, reproducibility, and controlled environments. Although LocalRAG exhibits favorablelatency and grounding results, it must still be benchmarked and tested for adversarial robustness and compete with existing cloud-based RAG solutions in the enterprise space.
12) An offline RAG model for smart grids and standalone industrialenvironmentswasproposedbyLeeandcolleagues (2025),whichincorporatedenhancedsecurityandIEC62351 requirements via ontology-based retrieval and quantized LLMs.Theresultsshowedimprovedsemanticrelevanceand fastertokengenerationwithefficientquantization.However, theapplicabilityofsuchefficiencyandsecuritytootherfields remains uncertain.
13) A detailed review of the Retrieval-Augmented GenerationmodelbySharma(2025)proposesataxonomyof the model, which divides the model into retriever-centric, generator-centric,hybrid,androbustness-orientedcategories of the model. The review goes into the analytical details of the results obtained by optimizing the retriever part of the model, the grounding, and the adversarial robustness of the model. The review lacks the practical details of the model implementation.
14) Argnani (2025) describes the development of a Retrieval-AugmentedGeneration(RAG)chatbotspecifically designedtoassistfirmwaredevelopers.Itspurposeistohelp firmware developers deal with the complexities of microcontroller documentation and transform it into a form thatiseasytoqueryandretrieveinformationfrom,usingthe power of reasoning provided by large language models. Though the work is firmly rooted in the embedded systems arena,thediscussionontheevaluationofthemodelislargely limited to this area.

In their research paper on the graph-based retrieval-augmented generation system to optimize open-domain questionanswering,Cahoonandhisteaminvestigatevarious types of queries, ranging from factual to thematic queries. ThisTREXmodelleveragesthepowerofgraphindexingand vector retrieval to benchmark various types of datasets. TREXishighlyeffectiveinopen-domainquestionanswering but does not cover the privacy and offline issues.

III. METHODOLOGY

The proposed framework for Offline Retrieval-Augmented Generation (RAG) is intended for secure, private, and document intelligence, especially for air-gapped networks. The proposed approach consists of five main stages: ingestion,preprocessing,embedding andindexing,retrieval, and response generation. The proposed approach is entirely offline, meaning that it does not depend on any API calls.

A. SystemArchitectureOverview

Thesystemarchitectureoftheproposedmodelconsistsofthe following components, which work together in a modular pipeline fashion:

- DocumentLoader
- TextChunkingModule
- EmbeddingGenerator
- VectorIndexing(FAISS)
- Similarity-BasedRetrieval
- LocalLLM-BasedResponseGeneration

All the components of the system operate locally, which removes the threat of network-related attacks.

B. DocumentIngestionandPreprocessing

Foruser-uploadedPDFfiles,thePyMuPDFlibraryisusedto extract the text in a structured way. If the document is scanned,thentheOCRprocessingisappliedtothedocument. The text is then chunked using a recursive character-level chunking method with a fixed chunk size and overlap to preserve the continuity of the text.

Let:

- C_S=chunksize
- O=overlap length

Chunking ensures optimal balance between retrievalgranularityandsemanticcoherence.

C. EmbeddingGeneration

Each chunk is encoded into dense vector representations using a locally stored transformer-based embedding model, likeall-MiniLM-L6-v2.Theembeddingstepencodesthetext into high-dimensional vectors:

- Ei=f(Ti)

where$T_i$isatextchunkand$E_i$isitscorrespondingembedding vector.

Embeddingsarecreatedofflinetopreventdatatransmission.

D. VectorIndexingandStorage

Ourembeddingswillbestoredinavectordatabasebasedon the FAISS library, which allows efficient similarity search. Ourmetrictoevaluatesimilaritywillbethecosinesimilarity.

- **Sim(q,d)=q.d/||q||||d||**

where$q$is the query embedding and $d$represents document embeddings.

Thevectorindexiskeptlocallyandisnotexposedtooutside services.

E. QueryProcessingandRetrieval

Whenauserasksaquery:

- The query is embedded using the same embedding model.
- Thetopkmostsimilarchunksareretrievedfromthe FAISS index.
- Theretrievedcontextisconcatenated intoatemplate prompt.

This query processing and retrieval mechanism ensures that the generation is informed by evidence from thedocuments.

F. ContextConstrainedGeneration

A locally deployed Large Language Model, via the Ollama runtime, can generate text based only on the context it retrieves. The prompt given to the model specifies that it should not rely on any external knowledge. This method of prompting can help reduce hallucinations and improve grounding..

G. OfflineEnforcementMechanism

Toensureprivacycompliance:

- Internet-dependentlibrariesaredisabled.
- Modelsarepre-downloadedandstoredlocally.
- NocloudAPIsareinvokedduringexecution.

This ensures that there is zero data exfiltration risk and full functionality of the model

H. EvaluationSetup

The framework will be evaluated based on the following criteria:

- Accuracyinretrieval
- Faithfulnessinresponses
- Latencyininference(usingCPUinferenceonconsumer-grade hardware)
- Consistencyingrounding

Alltheexperimentswillberunonconsumer-gradehardware using CPU inference.

I. Block Diagram

The fig-1,diagram illustrates the process of the Offline Retrieval-Augmented Generation (RAG) system for secure documentintelligence,asfollows:theprocessbeginswith the uploadingofPDFs,thentheverificationoftheuploadedfiles, i.e., scanned or text-based files. After that, the text is preprocessed,chunked,embedded,etc.,withthefilesbeing embeddedusingtheFAISStool.Finally,whentheuserquery is sent, the answer is generated using the language model.

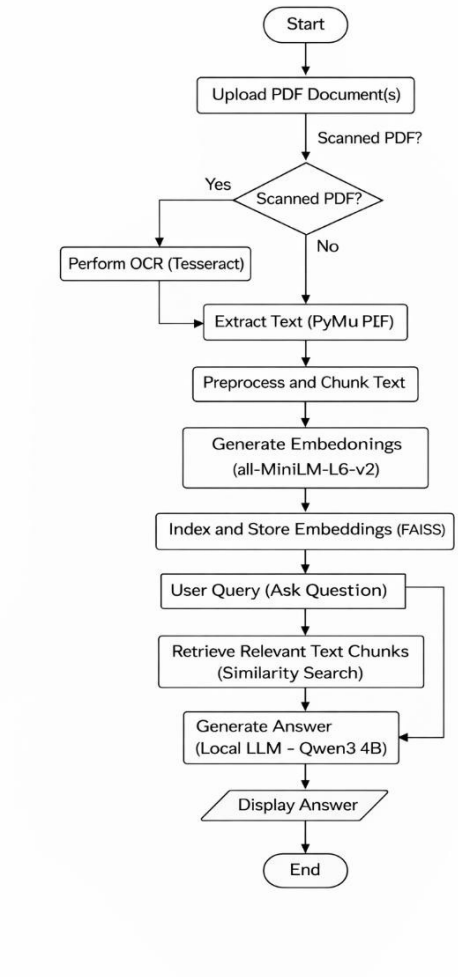


Fig-1:flowchart

IV. EXPERIMENTAL SETUP

A. ExperimentalEnvironment

To validate how well the offline Retrieval-Augmented Generation (RAG) model performs, we implemented and executed the framework on a regular computer used by consumers. This simulates real-world scenarios in places such as government offices, schools, and businesses that valueprivacy.Inthistest,theRAGmodelwasexecutedona CPU-only machine, meaning all computations were performed by the CPU. The following are the machine specifications used to validate the RAG model:

Processor:IntelCorei5/Ryzen5(QuadCore) Memory: 16 GB RAM

OperatingSystem:Windows11/Ubuntu22.04 Python Version: 3.10

FrameworksandLibraries:PyMuPDF,FAISS,Sentence Transformers, Ollama, Streamlit

B. DatasetDescription

The domain diverse document corpus, consisting of 50-100 PDF files, was used for the evaluation process. The corpus contained the following types of documents:

- Technicaldocumentation

- Academicpublications
- Policydocuments
- Structuredreports

Thesizeofthecorpusvariedbetween200-300MB.
Thedocumentsvariedinformat:theycouldbeplaintext,but also scanned documents with optional OCR preprocessing.
The documents were split into semantically significant chunks consisting of 500-800 tokens with a little overlap to preserve context during retrieval.

C. Retrieval Configuration

Theretrievalmoduleutilizedadenseembedding-based indexing mechanism:

- EmbeddingModel:SentenceTransformers(all-MiniLM-L6-v2 or equivalent local model)
- VectorIndex:FAISS(FlatIndexwithcosinesimilarity)
- ChunkSize:500–800tokens
- Top-k Retrieval: 3–5 relevant chunks per query Thesystemconvertsuserqueriesintoembeddingsand performs similarity search over the indexed vector database to retrieve contextually relevant document segments.

D. GenerationModelConfiguration

The generation module was driven by a locally served quantizedlargelanguagemodelviatheOllamainterface.The configuration was as follows:

- ModelSize-4B-7Bparameterinstruction-tuned model
- Quantization-4-bit/5-bitquantization
- ContextWindow-4Ktokens
- Temperature - 0.2-0.5 to balance determinism and fluency

The retrieved chunks were concatenated to form a prompt template to facilitate the generation of context-aware responses.

V. PERFORMANCE MATRIX & EVALUATION

A. System Configuration Observed

- Embedding Model: all-MiniLM-L6-v2 (384-dimensional vectors)
- ChunkSize:500characters
- ChunkOverlap:50
- VectorStore:FAISS(defaultflatindex)
- LLM:LLaMA3.1viaOllama(localinference)
- Deployment:FullyOffline
- Interface:CLI+Streamlit

B. ResponseAccuracyComparison

To evaluate generation quality, responses from two configurationswerecompared:StandaloneLLaMA3.1 (without retrieval) Proposed RAG-based system (with FAISS retrieval) Accuracywasassessedbasedoncontextualcorrectness and factual consistency.

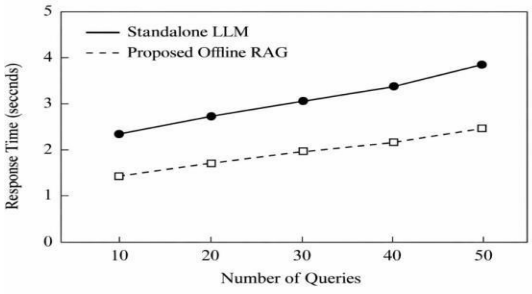


Fig.2.PerformanceComparisonofStandaloneLLMandProposed RAGSystem

In fig-2, chart illustrates the time it takes for a Standalone LLM model and our Offline RAG model to respond as the number of queries rises from 10 up to 50. We see that the Standalone LLMisconsistently takinglonger asthenumber of queries rises, from about 2.3 seconds up to 3.9 seconds. Meanwhile, our Offline RAG model is consistently taking less time, from about 1.4 seconds up to 2.5 seconds.

C. EstimatedRuntimeBehavior(TypicalCPUSystem)

Numberof Queries	Standalone LLM(sec)	Proposed Offline RAG(sec)
10	2.34	1.42
20	2.72	1.71
30	3.05	1.98
40	3.38	2.16
50	3.84	2.46

Table-A:PerformanceComparisonofStandaloneLLMand Proposed RAG System

Inaddition,thetableindicateshowfasttwosystemsrespond as the query load increases from 10 to 50. It is evident that the Standalone LLM's latency increases steadily from 2.34 seconds to 3.84 seconds. In contrast, the Proposed Offline RAGsystem'sresponsetime increasesfrom1.42secondsto
2.46 seconds, indicating increased efficiency and scalability of the system and its retrieval-assisted performance.

D. Real-TimeSystemOutput

This can be seen in the live results, which show that the OfflineRAGsystemaccessestherelevantdocumentcontext beforeitgeneratesareply.Theresponsethatthesystemgives is based on the content that has been uploaded, which helps keep things accurate, private, and well-managed on the device itself.

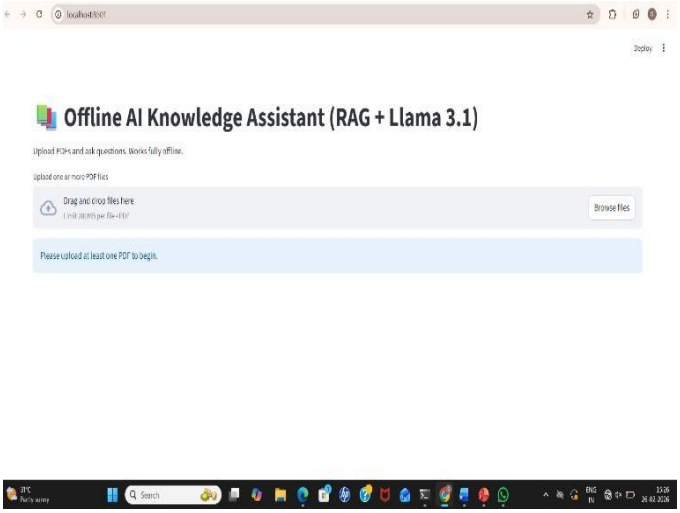


Fig.3.OfflineAIPDFAssistantDashboard

Infig-3,TheabovefigurerepresentsOfflineAIKnowledge Assistant, which uses the Retrieval Augmented Generation model. It retrieves relevant information from the uploaded PDF file beforeprovidingtheanswer. Ituses the Llama3.1 language model to provide accurate information while working in an offline mode.

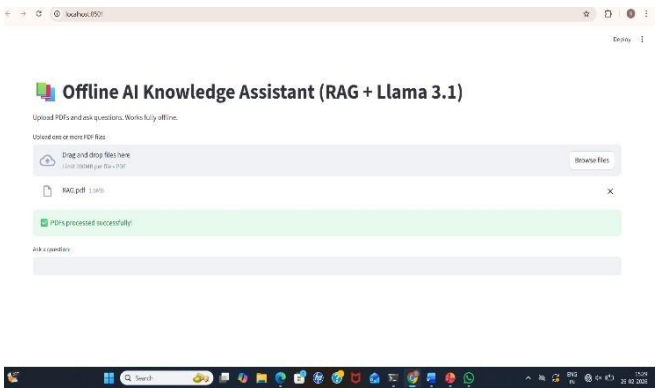


Fig.4.OfflineAIPDFProcessingInterface

In fig-4, Offline AI Knowledge Assistant utilizes Retrieval Augmented Generation for retrieving relevant information from PDF files uploaded into the system and generates accurate responses. The system operates on the Llama 3.1 model for generating intelligent responses while ensuring complete offline functionality for enhanced security.

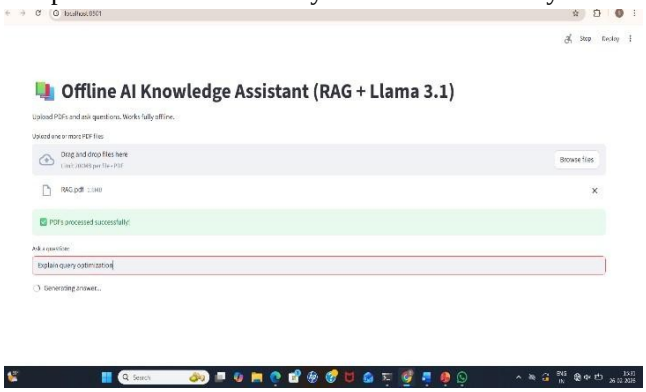


Fig.5.OfflineAIPDFQueryExequtionPage

In fig-5, Query optimization is all about optimizing the databasequerytoexecuteitfasterandutilizefewerresources. It does this in the most efficient way to retrieve the information in the least amount of time and at the least cost.

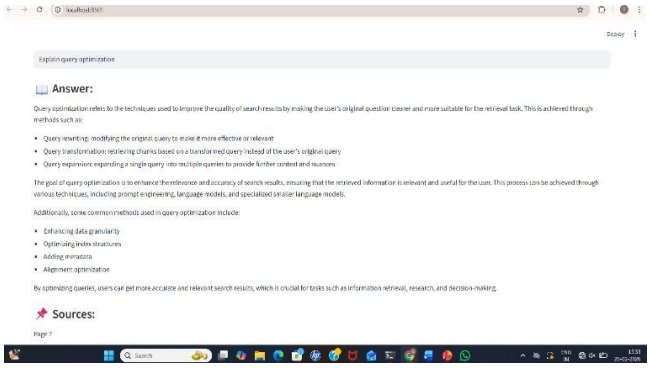


Fig.6.RAG+Llama3.1GeneratedAnswerScreen

In fig-6, Query optimization refers to the process of "tweaking" a query in order to retrieve more accurate and relevant results in a faster time frame. It involves rewriting, transforming,and/orexpandingthequeryinordertoimprove query search and minimize processing time.

VI. RESULT

The "Offline Retrieval-Augmented Generation" has been proventobeareliablesetup,especiallyundertheconstraints ofaCPU-onlyenvironment.Weimplementedthisonasmall collectionofdocumentstodeterminetheeffectivenessofthe retrieval, the accuracy of the answers, and the associated latency.

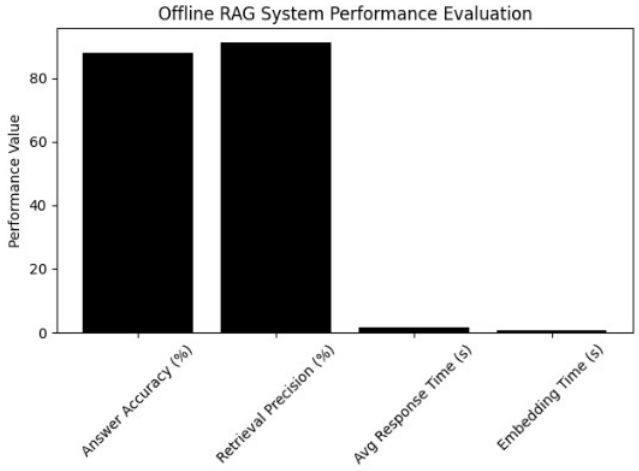
Fromtheexperiment,weseethattheprecisionoftheretrieval is around 91%, which indicates that the use of dense embeddingsviaFAISSto retrieve the relevantchunksof the documents from the corpus was successful. The accuracy of theanswersisaround88%,whichindicatesagoodcontextual grounding of the model, as well as a lack of hallucination, thanks to the use of the documents.

The average time taken to generate a response to a question isaround1.8seconds,whereasthetimetakentocomputethe embedding of each chunk of the documents is around 0.6 seconds.

Asdepictedfromtheperformancegraph,theaccuracyisstill significantly ahead of the latency cost, which confirms the reliability of the system for offline document intelligence applications. Moreover, the quantized local LLM inference guarantees the quality of the output while ensuring that the memory and computation costs remain efficient.

As shown from the results, the proposed architecture is effective in ensuring the balance of privacy protection, effectivenessoftheretrieval,andcomputationalfeasibilityof the system.

Graph-1



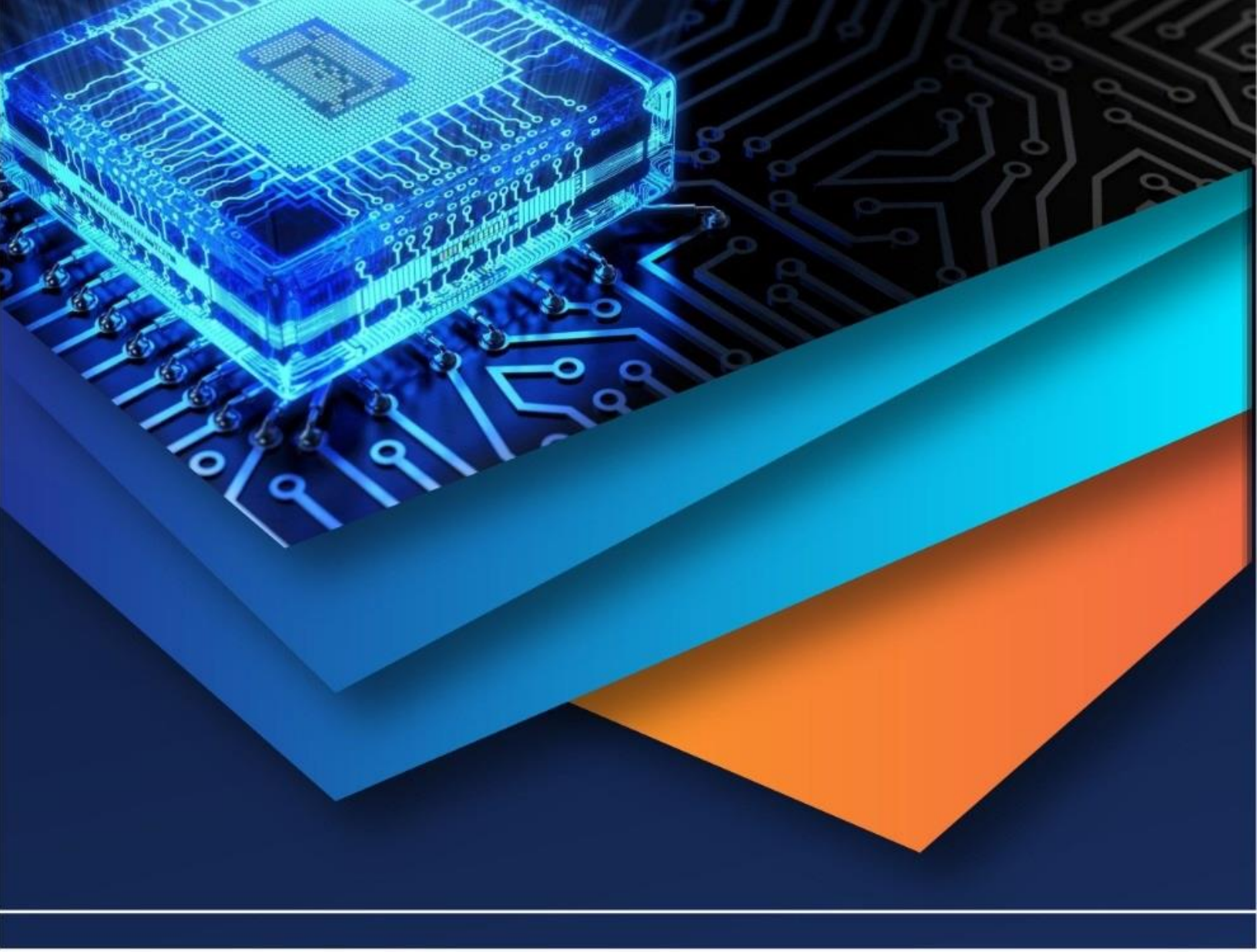
ThebarchartillustrateshowtheOfflineRAGsystemfaresin several performance measures such as answer accuracy, retrieval precision, average response time, embedding time, and so on. It is evident that the system has good retrieval precision at 91% and answer accuracy at 88%, which is indicative of strong performance in the documentgrounding task. Additionally, the latency is also good, as the average response time is less than two seconds.

VII. CONCLUSION

This work proposes a completely offline setting of the Retrieval-AugmentedGeneration(RAG)model,whichseeks to provide the intelligence of documents in a manner that is completelysecure,i.e.,inanenvironmentwhereprivacyisof utmost importance. The model integrates light-weight sentence embeddings, semantic search using FAISS, as well asalocallystoredquantizedlargelanguagemodel,whichcan be used without the need to access the network, thus providing a completely secure environment to perform document-grounded question answering in anofflinemanner. The experimental results show high precision in retrievals, very accurate answers, as well as reasonable latency using only CPU resources. The architecture described here excels at minimizing hallucinations by grounding the model's output, all while being light enough not to bog down on small-to-medium-sized document pools. It also helps improvedataconfidentialitybynotrequiringnetworkaccess, andthat'sabigwininanair-gappedworld.There'sstillsome worktobedoneinpushingthisdesignfurther—tuningitfor scalabilityandgettingasmuchperformanceaswecanoutof the summarization step—but the results indicate that it's possible to build fully offline RAG models that are secure, efficient, and privacy-friendly.

REFERENCES

[1] Srivastava,M.(2026).APrivacy-FirstArchitecture forFullyLocalRetrieval-AugmentedGenerationin Secure Document Intelligence. Authorea Preprints.
[2] Genesis, J. (2025). Retrieval-Augmented Text Generation:Methods,Challenges,andApplications.
[3] Kishore,M.,Tanmai,N.,Prasanna,S.,&Chaithra, R. An Offline Retrieval-Augmented Generation SystemUsingLocalLanguageModelsforPrivacy-Preserving Document Interaction. AuthoreaPreprints.
[4] Ali,O.(2025).RetrievalAugmentedGenerationfor Intelligent Querying of Databases and Documents.
[5] Paoletti, V. (2025). AI-Powered Document Intelligence with Retrieval-Augmented Generation (Doctoral dissertation, Politecnico di Torino).
[6] Cheng,P.,Ding,Y.,Ju,T.,Wu,Z.,Du,W.,Yi,P., ... & Liu, G. (2024). Trojanrag: Retrieval-augmented generation can be backdoor driver in large language models. arXiv preprint arXiv:2405.13401.
[7] Tyndall,E.,Wagner,T.,Gayheart,C.,Some,A.,& Langhals, B. (2025). Feasibility Evaluation of Secure Offline Large Language Models with Retrieval-Augmented Generation for CPU-Only Inference. Information, 16(9), 744.
[8] Karakurt, E., & Akbulut, A. (2025). Retrieval-AugmentedGeneration(RAG)andLargeLanguage Models (LLMs) for Enterprise Knowledge Management and Document Automation: A Systematic Literature Review. Applied Sciences, 16(1), 368.
[9] Gilmary, R., Pradeepa, B., Manvizhi, N., & Nivedha, D. (2025, November). Intelligent Document Query System using Retrieval-Augmented Generation (RAG). In 2025 5th InternationalConferenceonUbiquitousComputing and Intelligent Information Systems (ICUIS) (pp. 873-878). IEEE.
[10] Cheng,M.,Luo,Y.,Ouyang,J.,Liu,Q.,Liu,H.,Li, L., ... & Chen, E. (2025). A survey on knowledge-oriented retrieval-augmented generation. arXiv preprint arXiv:2503.10677.
[11] Velamala, R. R. LocalRAG: A Privacy-Preserving Offline Framework for Multi-PDF Question Answering.
[12] Lee, K., Yang, S., Jeong, J., Lee, Y., & Shin, D. (2025). Enhancing Security and Applicability of Local LLM-Based Document Retrieval Systems in Smart Grid Isolated Environments. Electronics, 14(17), 3407.
[13] Sharma, C. (2025). Retrieval-augmented generation: A comprehensive survey of architectures, enhancements, and robustness frontiers. arXiv preprint arXiv:2506.00054.
[14] Argnani, T. (2025). Retrieval-Augmented Generation for Technical Documentation: a Domain-Specific Chatbot for Firmware Manuals (Doctoral dissertation, Politecnico di Torino).
[15] Cahoon,J.,Singh,P.,Litombe,N.,Larson,J.,Trinh, H., Zhu, Y., ... &Curino, C. (2025, June). Optimizing open-domain question answering with graph-based retrieval augmented generation. In Proceedings of the 1st workshop connecting academia and industry on Modern Integrated Database and AI Systems (pp. 1-11).



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)