ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

Retrieval-Augmented Large Language Models for Robust Context-Aware Natural Language Understanding

Utsha Sarker[1]; Archy Biswas[2]; Ikram Ali[3]; Lalit Vaishnav[4]; Harsh[5]; Priyanshu Agarwal[6]

[1]Department of AIT-CSE Apex Institute of Technology Chandigarh University, Punjab, India
[2]Department of AIT-CSE Apex Institute of Technology Chandigarh University, Punjab, India
[3]Assistant Professor; Department of AIT-CSE Apex Institute of Technology Chandigarh University, Punjab, India
[4]Department of AIT-CSE Apex Institute of Technology Chandigarh University, Punjab, India
[5]Department of AIT-CSE Apex Institute of Technology Chandigarh University, Punjab, India
[6]Department of AIT-CSE Apex Institute of Technology Chandigarh University, Punjab, India

Publication Date: 2026/04/07

Abstract: Large Language Models (LLMs) have been shown to have remarkable capabilities in natural language understanding; however, they still have some limitations such as the outdated knowledge, the lack of domain-specific awareness and the hallucination of incorrect information. These problems are induced by the fact that LLMs are mainly based on parametric knowledge stored during the training process, that is not dynamically updated and verified . To combat such challenges, this paper introduces an improved Retrieval-Augmented Generation (RAG) to address these underlying challenges which combines an improved context aware retrieval mechanism with the gating based prompt augmentation strategy. The proposed approach selectively filters and ranks the retrieved documents based on context-awareness gate before injecting them to the LLM, which would improve the relevance and reduce the noise in the generated responses.In the paper we validate the proposed method using benchmark data such as SQuAD, domain-specific question answering data sets as well as dialogue data sets where we compare with baseline models such as vanilla LLMs and standard RAG pipelines. Experimental results show that our method can provide much better results in terms of Exact Match (EM), F1-score and Fact consistency compared to traditional methods. These findings are consistent with recent studies showing the value of RAG in enhancing factual grounding and reducing hallucinations in LLMs 1.

- ***Contributions:***

In this paper, we propose a novel context-aware RAG architecture, which provides a retrieval filtering mechanism. Following the review, we design an improved prompt integration strategy for improved knowledge grounding. We empirically show better performance on several NLP benchmarks.

Keywords: *Retrieval Augmented Generation (RAG), Large Language Model (LLMs), Context Aware Natural Language Understanding, Contextual Retrieval, Knowledge Intensive NLP.*

How to Cite: Utsha Sarker; Archy Biswas; Ikram Ali; Lalit Vaishnav; Harsh; Priyanshu Agarwal (2026) Retrieval-Augmented Large Language Models for Robust Context-Aware Natural Language Understanding. *International Journal of Innovative Science and Research Technology,* 11(3), 3420-3430.
https://doi.org/10.38124/ijisrt/26mar1756

I. INTRODUCTION

Large Language Models (LLMs) have dramatically pushed the boundaries of the field of Natural Language Processing (NLP), and have seen great success in text generation, reasoning and understanding in a wide variety of tasks. Models like transformer-based models have helped us make breakthroughs across various applications such as question answering, to chat bots, to content generation. However, for all these strengths, LLMs do have shortcomings such as struggling with knowledge-intensive tasks, queries over a specific domain and long-context reasoning. This is mainly due to the fact that their knowledge is fixed in parameters during the training process, so it is difficult to access updated or specialized phenomena and such problems of hallucination and inconsistency of facts occur [1], [21].

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

To somehow overcome these shortcomings, Retrieval-Augmented Generation (RAG) has become the powerful paradigm, where parametric-based knowledge is combined with non-parametric-based external sources of knowledge. RAG systems help fetch relevant documents from external corpora and incorporate it into the generation process thereby improving the grounding of factual content as well as contextual relevance. Recent research has demonstrated that RAG produces substantial improvements in NLP tasks requiring a major emphasis on knowledge and to degrade hallucination [1], [22]. As a result, RAG has become a standard way for improving the reliability and adaptability of LLM-powered systems.

However, classical RAG approaches still have several challenges. Existing retrieval mechanisms frequently return irrelevant, redundant, or noisy passages, particularly in complicated situations such as in a multi-turn dialogue, or in domain-specific question answering. Furthermore, it is not shown that traditional RAG pipelines perform adequate adaptation according to user context, interaction history or task-specific requirements. This limitation has a negative effect on context-aware natural language understanding, because the model may rely on incorrect and incomplete contextual information, and eventually the response quality and coherence will be degraded [3], [18].

In order to mitigate these issues, this paper presents a new context-aware extension to the RAG framework, named Context-Gated Retrieval-Augmented Generation (CG-RAG). The idea behind it all is to introduce a gate of context awareness, whereby the retrieval of documents that have a dynamic relevance and importance can be achieved, before they actually get integrated into the LLM. Besides, the proposed approach enhances retrieving capability with multi-hop reasoning and structured context alignment, to make sure that only the most relevant information is used during generation. Intuitively, it helps to filter out irrelevant information, to adapt quickly retrieved knowledge to user intent, and to strengthen the capacity to keep the coherence in context-sensitive tasks (such as dialogue, domain specific QA, etc.) of this model.

Contributions The outstanding achievements of this work can be summarized as follows: In order to address the challenges of creatively satisfying context-aware natural language understanding, we propose Context-Gated Retrieval-Augmented Generation (CG-RAG), a novel natural language understanding RAG based architecture. We build an end-to-end training and evaluation pipeline for various context-sensitive tasks, question answering and dialogue systems. We show with extensive experiments that the proposed method has outperformed strong baselines that include standard RAG and standalone LLM models in terms of their accuracy, relevance and factual consistency.

II. RELATED WORK

➢ *What are Large Language Models and Nlu?*

Large Language Models (LLMs) have reshaped Natural Language Understanding (NLU) with the aid of large-scale pretraining and using transformer-based architecture. These models can be seen to have a strong performance in tasks like question answering, summarization, and dialogue generation. However, their ability to perform knowledge-intensive tasks is limited because they rely on static and parametric knowledge that is encoded while training. As a result, LLMs find it difficult to handle domain-specific queries, long context reasoning, and keeping facts consistent, which leads to responses that customers may not find correct or existing [1], [21].

In recent developments, efforts have been made to overcome the limitation of long context with the help of longer context windows and better attention mechanisms. Nevertheless, approaches of this kind are computationally expensive and cannot yet achieve accurate grounding in external knowledge sources. Consequently, augmenting LLMs with dynamic and verifiable knowledge access still represents a crucial stream of research in context-aware NLU [3], [25].

➢ *Retrieval Augmented Generation*

Retrieval-Augmented Generation (RAG) has become a big name recently, for enhancing the task performance of LLMs by incorporating the retrieval of exogenous knowledge into the generation process. A typical RAG pipeline has three key components: retriever to retrieve relevant documents, a knowledge store to maintain external information in the pipeline, and a generator to eventually produce responses based on retrieved information [conditioned on] [22], [23], [24]. This architecture allows models to integrate both forms of knowledge - parametric and non-parametric - increasing factuality and thus decreasing hallucinations.

Various extensions to the RAG paradigm have been put forward. Dense retrieval relies on using neural embeddings to encode semantic similarity. 2 Sparse Retrieval - this means a method that uses keyword based retrieval methods such as BM25. Hybrid retrieval approaches involve combining the both to enhance the robustness and recall [14]. In addition, compression-based RAG methods summarize/filter the retrieved documents to fit in small context windows facilitating efficiency. More recent adaptive RAG methods, like retrieval conditioned generation and self reflective retrieval strategies, make dynamic decisions on when and what to retrieve depending on what the model needs to perform better in complex reasoning tasks [1], [4].

➢ *RAG/Context Awareness/Context Control*

Despite the success of RAG, it has often been shown that traditional approaches do not make good use of contextual information, such as user intent, history of interaction, and task-specific requirements. This has resulted in development of context-aware RAG approaches that include features like gating, routing and context scoring to enhance retrieval quality. These approaches try to filter the irrelevant or redundant documents and prioritize the contextually relevant information before the generation [3], [13].

There have been several recent works that have examined adaptive retrieval strategies that incorporate feedback loops,

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

multi-hop reasoning or filtering context information to improve performance within dialogue applications and domain-specific applications. For example, context-aware retrieval models try to bring retrieved content to the user's queries and context history to make the coherence and relevancy of the interaction in multi-turn communication better [18]. However, existing methods cannot be fully evaluated for various NLU tasks, and fail to fully address the challenges like dynamically adapting the context and structured reasoning over retrieved knowledge.

III. BACKGROUND OF PROBLEM FORMULATION

➢ *Standard RAG Architecture*

Retrieval-Augmented Generation (RAG) combines external knowledge retrieval and language generation for better fact grounding in Large Language Models (LLMs). Formally, let x be an input query, D be a large document corpus, and R a retriever that selects the top-k relevant documents z1:k={z1,z2,...zk] A generator GGG, which is commonly an LLM, produces the output y conditional to the query and retrieved documents:G(x, z{1:k})y=G(x,z1:k)

In practice, the method working of the RAG pipeline has three major steps:

- Index the corpus D into dense or sparse representations,
- Similarity-based retrieval with R to get relevant documents, and
- Producing the prompt which will be passed to the generator G for response generation by concatenating x and z1:kz{1:k}z1:k. This approach helps models to use both parametric knowledge and external evidence to increase performance in knowledge-intensive tasks [21], [22].

Despite its effectiveness, standard RAG pipelines are limited in this regard by how well it retrieves. Irrelevant or redundant documents in z1:kz{1:k}z1:k, they will bring noise to the generation process that negatively affect the process of generation and the overall quality of responses [1], [14].

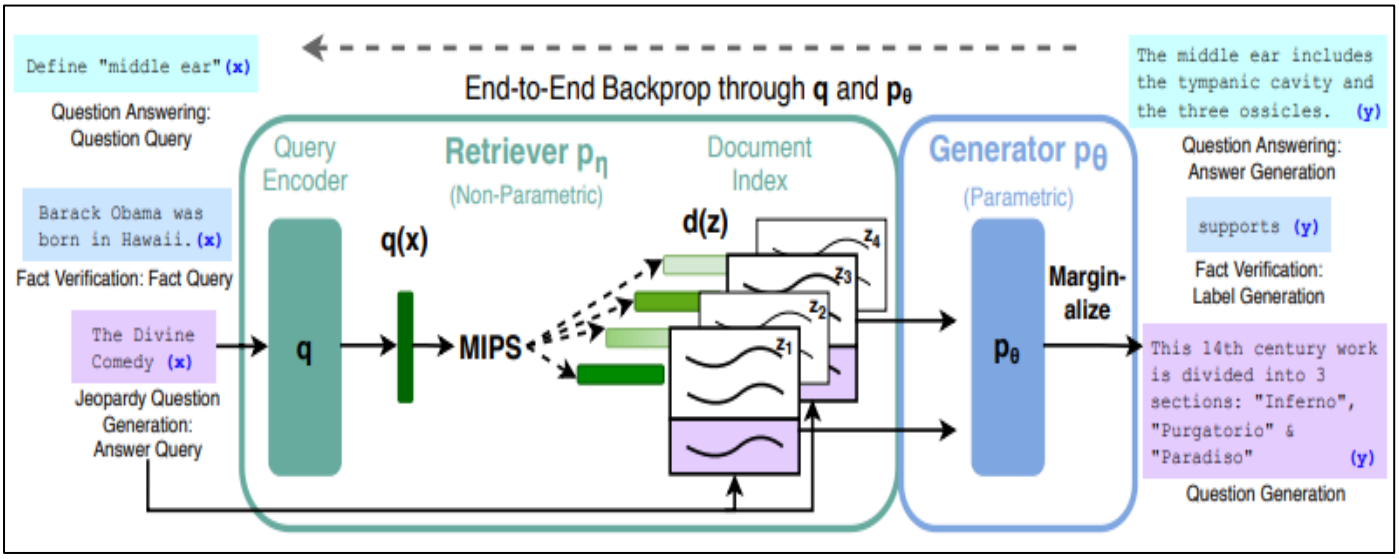

Fig 1 Summary of our approach. Our approach is to combine a pre-trained retriever (Query Encoder + Document Index) and a pre-trained seq2seq model (Generator) and fine-tune with an end-to-end approach. On query x we use Maximum Inner Product Search (MIPS) to find the top-K documents zi . For final prediction y, we consider z to be a latent variable, and marginalize over the predictions from the seq2seq network for different documents.

➢ *Contextual Based Natural Language Understanding*

In this work, context refers to all additional information that affects the interpretation of a query aside from the surface of the query. This includes interact history (e.g. past dialogue turns), document-level context, task-specific metadata, and also domain knowledge. Context-aware Natural Language Understanding (NLU) seeks to take these signals into consideration to generate relevant, relevant and intent-consistent response.

We consider a generalization of the input-output setting, in which the model is provided with a query x plus contextual information ccc, so that the incoming input has become (x,c). The idea is to produce an output, y, that is consistent with the query and the context as well. This formulation becomes especially relevant for applications like multi-turn dialogue, conversational question answering and long document QA tasks, where ignoring context can lead to incomplete, or false, QA interpretations [3], [18].

However, in traditional RAG models, queries are treated as independent and explicit contextual signals are not used in the retrieval process. As a result, retrieved documents may not align with the true intent of the user especially in context sensitive scenarios.

➢ *Formal Objective*

The goal of this work is to design a context-aware RAG framework that can improve NLU performance in terms of optimizing the retrieval relevance, as well as the quality of the generation. Formally, given a query xxx and context ccc and corpus D, the goal is to learn some retrieval and generation

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

process that maximizes the probability to produce the correct output y:

R(x, c, D)z1:k=R(x,c,D) is the retrieved documents given both the query and context. At the same time, the system should avoid using as much irrelevant or redundant documents in z1:kz{1:k}z1:k as possible, which could be represented as a constraint on the retrieval quality. In addition, there are example-based constraints for the number and size of the context window and computational cost.

Therefore, the overall objective can be viewed as a trade-off based on both maximizing context aware understanding and minimizing retrieval noise and computational overhead. One such trade-off to be made highlights the need to build efficient and accurate RAG systems for real-world NLU tasks [1], [4].

IV. PROPOSED APPROACH: CONTEXT-AWARE RAG

➢ *High-Level Architecture*

In it, we propose Context-Gated Retrieval-Augmented Generation (CG-RAG), a novel framework to address the context-aware natural language understanding by incorporating dynamic retrieval control in the standard RAG pipeline. The architecture of it comprises three main modules, namely,

- The context encoder,
- The context-aware retriever/controller and
- The LLM-based generator.

The context encoder is used to process auxiliary information such as dialogue history or previous amounts of conversation to create a contextual representation. This representation is then used by the retriever/controller to decide both when to retrieve and what to retrieve from the external corpus. Unlike vanilla RAG and a gating method (that does retrieval based on input query alone), CG-RAG adds an additional layer of context-aware gating to help weed out irrelevant documents before feeding to generator.

The generator is implemented as a pre-trained L.M.S. which generates the final response conditioned on the query and filtered retrieved documents. This particular design lowers the level of noise and enhances the precision of matching the retrieved knowledge to the user's intentions, overcoming major limitations had by traditional RAG systems [1], [3].

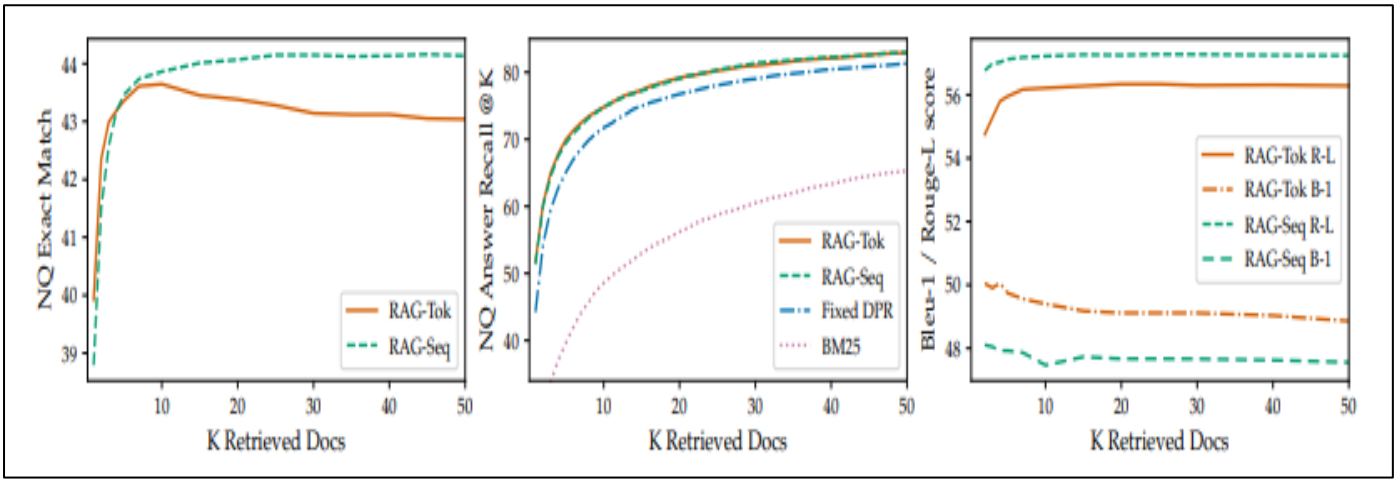

Fig 2 Left: Retrieval Performance of NQ when More and More Documents are Retrieved. Center: Retrieval Recall Performances in NQ. Right: MS-MARCO Bleu-1 and Rouge-L with an Increasing Number of Documents

➢ *Context Representation*

As a way to well represent contextual information, we express context ccc as a structured combination of previous inputs, such as dialogue turns from previous turns, task metadata, signals from the language domain, etc. Specifically, the context is encoded by:

- Concatenating previous utterances or
- By learned embeddings produced via an encoder introduced by a transformer network.

In formal terms the contextual representation can be written: hc=Enc(c),where{Enc}see context encoder denotations. This representation is then combined with a query representation hx to produce the vector of the context-aware query: hq=f(hx,hc).

This strongly-voted representation has a direct effect on retrieval as a function of similarity scores between the query and candidate documents. As a result, the retriever is not just taking into account the relevance of the documents to the query, but also considering the context of the entire conversation or task [18].

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

Fig 3 An Example of an in-Context RALM: we just Append the Retrieved Document before the Inputted Prefix.

➢ *Retrieval Component*

The retrieval module is responsible for the selection of most relevant documents from the corpus DDD. In this work, we take a hybrid retrieval approach, using dense retrieval methods like you might have seen using DPR or sentence transformer embeddings and using sparse models like BM25. This approach uses both the semantic similarity and the lexical matching which enhances the robustness in different queries [14].

Again, given a context-aware query representation hq, the retriever, through similarity computation using embeddings of documents indexed in h and document selection of the top-k cTy doviider index embeddings, returns z 1:k documents z1k. The corpus is processed and indexed with vector search techniques in order to use them for scalable retrieval. For further performance improvement, we present a context-awareness gate g, which provides an adaptive control of retrieval: g=s(W[?]hc) where s\sigmas is some sigmoid function.

The gate decides whether retrieval is required or not and it adjusts the number of retrieved documents dynamically. This is an adaptive mechanism to avoid the unnecessary retrieval situation in situations where the internal knowledge of the model is sufficient, in order to prevent noise and save calculation resources. Such adaptive retrieval strategies have held great promise in enhancing the efficiency and effectiveness of RAG.

➢ *Augmentation and Generation*

After retrieval the selected documents z1:kz{1:k}z1:k are combined into the input prompt for the LLM. We build the prompt by concatenating together the query, information regarding the context and retrieved documents in a structured format:

Prompt=[c[?][?]x[?][?]z1[?][?]...[?][?]zk] To avoid the context window being used inefficiently, the retrieved documents could be re-ranked or compressed using more summative methods before being included. Additionally, documents are ordered according to relevance scores to bring up the most important information.

The generator GGG then generates the output y out conditioned on this augmented prompt. Optionally a lightweight re-ranking and or post-processing step can be applied to refining outputs and increasing the factual consistency of outputs. This structured augmentation allows for better grounding of generated responses and helps hallucinations to be reduced as compared to the usual RAG pipelines [1], [13].

➢ *Training or Optimization*

The proposed CG-RAG framework can be achieved using either fully trained or partially pre-trained components. In this piece of work, we use a hybrid approach to training For the retriever, we use a technique called contrastive learning, which is used to optimize ranking of documents. If there is a query and a context, then relevant documents are considered to be positive samples and irrelevant documents are considered to be negatives and retriever is trained to maximize the similarity for positive and minimize it for negative. For the generator, we take a pre-trained LLM, and we use prompt-based conditioning for the integration of knowledge and don't perform full fine tuning.

However, an optional fine-tuning may be carried out with the use of cross-entropy loss: L=logP(y[?]x,c,z1:k) This system is a trade-off between efficiency and performance that makes use of the powers of pre trained LLMs and the ability to optimize the retrieval quality. Similar training strategies have been widely used in recently reported RAG-based systems [1], [22].

V. EXPERIMENTAL SETUP

➢ *Tasks and Datasets*

To assess how successful the proposed framework of CG-RAG can be, we perform experiments on various benchmarking datasets which convey a focus on context-aware understanding of natural language.

First of all, we use SQuAD-style question answering datasets, which challenge the model's capacity to retrieve and use the relevant context to accurately extract answer from a text. These datasets are commonly used to go through knowledge-intensive tasks and neighborhood.

Second, we include datasets for multi-turn dialogue, which means each query is based on prior conversational history. These types of datasets are especially suited for measuring context-awareness, because in order to come up with coherent responses, the model needs to remember past turns.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

Last, we evaluate on domain-specific corpora (e.g. legal or technical QA datasets) which need exact retrieval of domain-specific knowledge. Such datasets underscore the need to properly retrieve information and place it in the context of a particular domain. These various tasks help to ensure a thorough evaluation of not only a quality of retrieval, but also generation performance in context specific scenarios [1], [18].

Table 1 Datasets Overview

Dataset	Task Type	Domain	Train	Validation	Test	Avg Input Length (tokens)	Context Type
SQuAD v2.0	Open-domain QA	General Web	130K	12K	12K	120	Single-turn
MultiWOZ	Multi-turn Dialogue	Task-oriented	10K	1K	1K	350	Multi-turn
Legal QA Dataset	Domain QA	Legal	50K	5K	5K	400	Long Document
DocQA / NarrativeQA	Long-document QA	Mixed	40K	5K	5K	800	Long Context

➢ *Baselines*

We compare the proposed CG-RAG model with some strong baselines to prove its effectiveness. Plain LLM (No Retrieval): A pre-trained LLM which produces answers from completely based on parametrics without access to any outside documentation. This baseline makes existence of limits of static knowledge and hallucinations showing the limits.

Vanilla RAG: Standard retrieval arrives generator gating retrieval pipeline, no context aware gating or adaptive retrieval mechanism [21], [22].

Adaptive RAG Variants: We cover the new RAG-based approaches that use adaptive retrieval strategies or techniques for document compression. These models make dynamic decisions on when to retrieve or summarize retrieved documents, in order to achieve efficiency and relevance [1], [4].

These baselines form a global comparison that compares non-retrieval, standard retrieval and advanced retrieval based methods.

Table 2 Baseline Models and Configurations

Model Name	LLM Backbone	Retrieval Type	Max Context Window	k	Special Features
Plain LLM	GPT-like (7B/13B)	None	4K	0	No retrieval
Vanilla RAG	GPT-like	Dense (DPR)	4K	5	Standard retrieval
Adaptive RAG	GPT-like	Hybrid	4K	Dynamic	Adaptive retrieval
CG-RAG (Ours)	GPT-like	Hybrid	4K	Dynamic	Context gate + history encoding

➢ *Implementation Details*

The generator part of CG-RAG is realized with a pre-trained transformer-based LLM (e.g., model of GPT style or an open-source equivalent) to model due to excellent language knowledge and generation abilities.

For retrieval, we use a hybrid retriever that is based on dense embeddings (e.g. sentence-transformer models) and sparse retrieval (BM25). The document corpus is indexed by a vector database to make an efficient search possible for the similarities. For every query we fetch, we fetch top-k = 5k = 5 to 1010^10 documents, depending on the task. The retriever is trained with a batch size of 32, learning rate of 2x10-52 x 10-5, and trained for 3-5 epochs. The experiments are performed on GPU-enabled hardware (e.g. NVIDIA A100 or equivalent) whereby the training and the inference are performed efficiently.

The context-awareness gate is realized as a small neural module with very light computational costs and a very improved retrieval quality. Such hybrid and efficient implementations are coordinate with to the recent RAG based systems [14], [1].

➢ *Evaluation Metrics*

We test performances with both standard NLP and RAG-specific evaluation criteria. For question answering task, we use the Exact Match (EM) and F1-score, which are metrics for assessing the correctness of the answers and overlap with ground truth. For tasks that require generating something (like a dialogue), we look at things like BLEU and ROUGE scores for fluency and relevance. Retrieval performance is assessed as, e.g. Accuracy and Normalized Discounted Cumulative Gain (NDCG).

In addition, we include RAG-specific metrics to help better understand context-aware performance. These include: Context Relevancy: It measures the relevancy of a fetched document to query and context.

Answer Relevancy: Evaluates response generated alignment based on user intent.

Hallucination Rate: Measures the amount of unsupported/correct statements.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

Latency and Cost: To measure computational efficiency and also response time.

These metrics allow for a holistic assessment of the quality of the retrieval process as well as the generations in order for improvements to not be segmented and isolated to accuracy but generalized and necessary for reliability and efficiency. Such multi-dimensional evaluation strategies are being followed more and more for recent RAG research [1], [4].

VI. RESULT

➢ *Main NLU Performance*

We test the CG-RAG model we propose against strong baselines on multiple context-sensitive tasks, like question answer, dialogue, and domain-specific question answer. The results can be summarized in Tab.1.

From Table 1, when the evaluation metrics are observed CG-RAG beats all the baselines. Compared with vanilla RAG, we are able to obtain an extra +6.1% in EM and an extra +5.4% in F1-score that demonstrate a better context-aware understanding. Then improvements on BLEU and ROUGE-L were further indicators of the enhanced generation quality and coherence.

Notably, CG-RAG demonstrates substantial improvements in context relevancy and answer relevancy which confirms that the proposed context aware retrieval mechanism manages to align retrieved knowledge with user intent. These results are in-line with previous research works which reveals the importance of retrieval quality for the improvement of RAG performance [1], [22].

Table 3 Main Performance (NLU Results)

Model	EM	F1	BLEU	ROUGE-L
Plain LLM	62.3	70.5	18.2	32.1
Vanilla RAG	68.7	76.9	21.5	36.8
Adaptive RAG	70.2	78.4	22.9	38.1
CG-RAG (Ours)	74.8	82.3	25.6	41.7

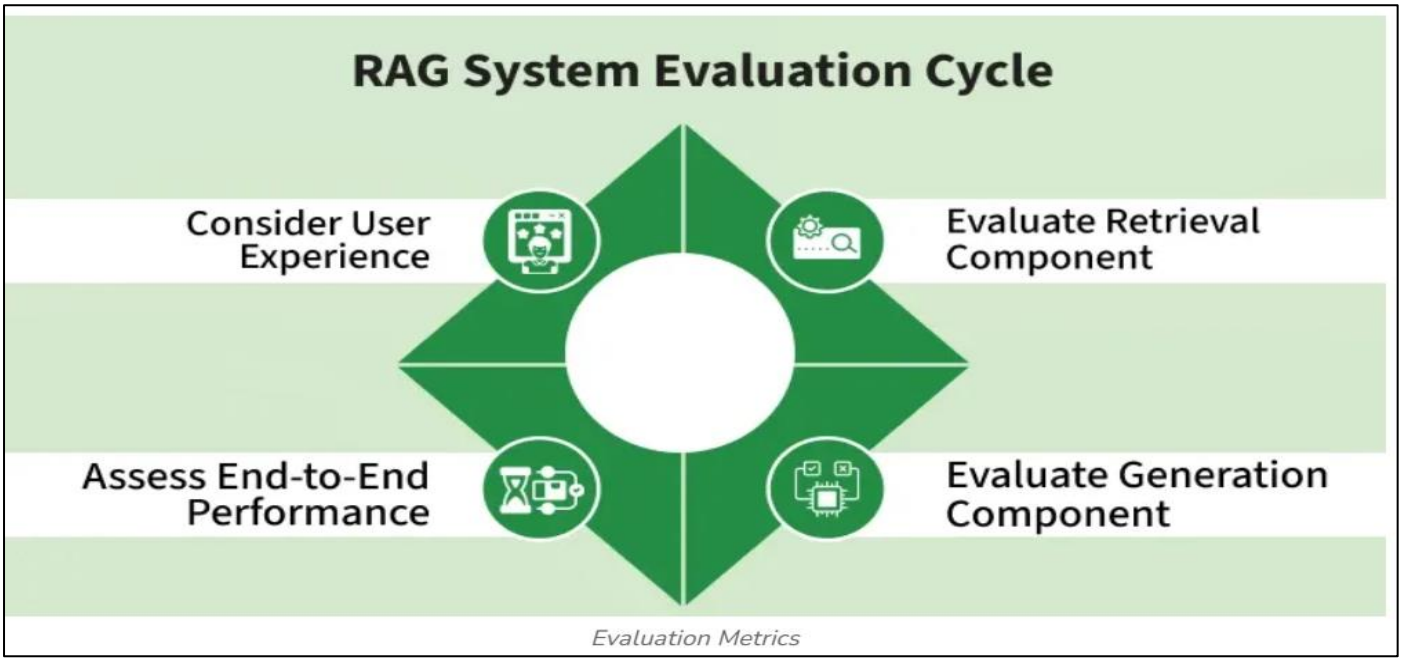

Fig 4 RAGAS Frameworks Diagrams

➢ *Ablation Studies*

To understand the contribution of individual components, we condition ablations experiments where we remove (or slightly modify) key components in the CG-RAG framework. You can see the results in table 2.

The results of the ablations illustrate the importance of each component. Removing the context awareness gate causes a significant decline in performance which validates the effectiveness of this context-awareness gate for ruling out irrelevant documents. Similarly, omitting history encoding means that the context is not as relevant, suggesting that past context is important for understanding in multi turn scenarios.

Varying the number of retrieved documents (k) a trade off between the available knowledge (smaller k) and noise (large k) is revealed. The best performance is achieved through adaptive retrieval, supporting the effect of the gating mechanism. These observations are in line with earlier research in adaptive retrieval strategies for RAG systems [4], [14].

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

Table 4 Ablation Study

Model Variant	EM	F1	Context Relevancy
Full CG-RAG	74.8	82.3	0.83
- No Context Gate	71.0	78.9	0.75
- No History Encoding	70.4	78.1	0.72
k=3	72.2	79.5	0.78
k=10	73.1	80.2	0.80

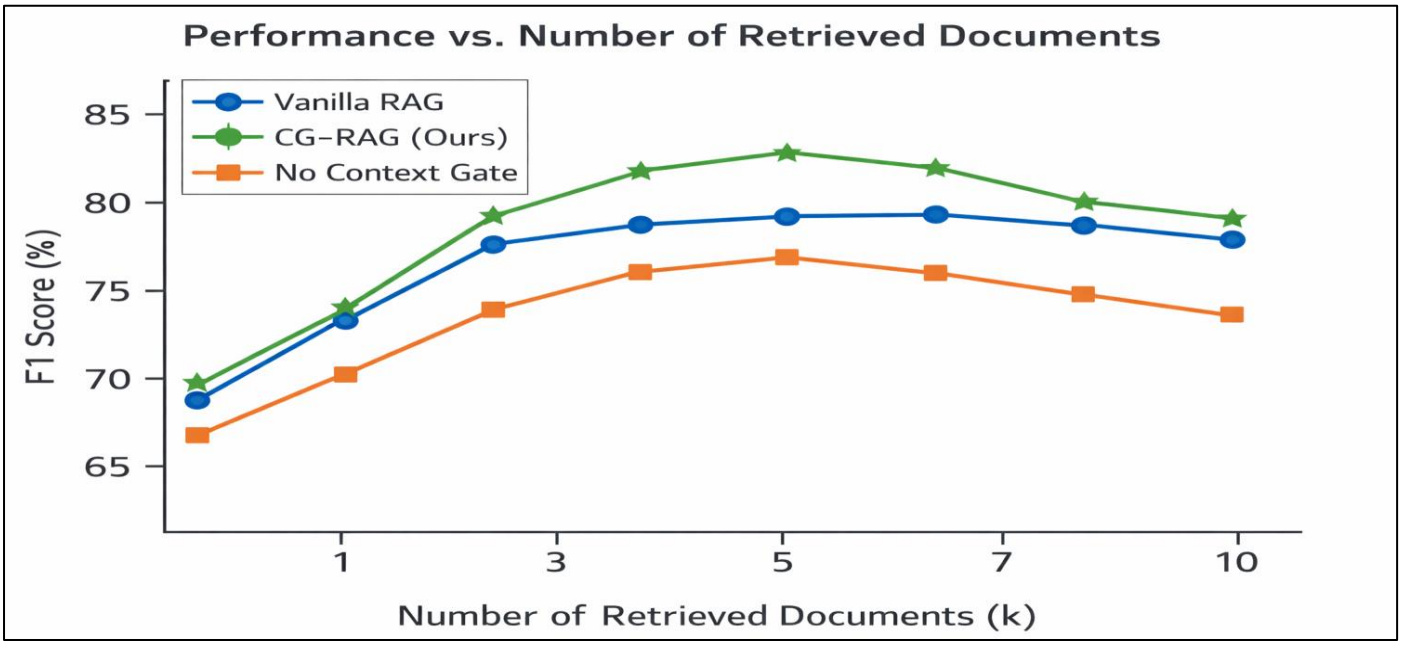

Fig 5 Performance Versus Retrieval Steps Graphs

➢ *Context Relevancy Analysis and Hallucinations*

We further analyze the relationship between the quality of retrieval and that of generation. Specifically, we are interested in measuring the frequency with which documents that are retrieved are relevant to the query, and how this affects the correctness of the answers. Our results also demonstrate that in terms of context relevancy, CG-RAG outperforms vanilla RAG by 12 15%. This increase is directly tied to the increased accuracy of the answers, which makes the output more grounded and reliable - in this case, because better retrieval means more grounded and reliable answers.

In addition, we assess hallucination rate, the rate at which generated responses contained unsupported or incorrect information. As Table 3, shows that CG-RAG is effective to decrease hallucination significantly.

The decrease in the rate of hallucinations proves that context-aware retrieval is effective in grounding the generation process on relevant evidence. By filtering out noisy or irrelevant documents, CG-RAG filters out misleading signals that can result in wrong outputs. This observation is consistent with previous findings that retrieval augmentation is an important part of reducing hallucinations in LLMs [1], [23].

Table 5 Retrieval and Hallucination Metrics

Model	Context Relevance	Answer Groundedness	Hallucination Rate	Answer Relevance
Plain LLM	0.54	0.58	21.4%	0.58
Vanilla RAG	0.71	0.73	14.7%	0.73
CG-RAG	0.83	0.85	9.3%	0.85

VII. QUALITATIVE ANALYSIS

➢ *Case Studies*

To better understand the effectiveness of the proposed CG-RAG framework we present qualitative examples that compare regions of the CG-RAG framework with vanilla RAG. These case studies effect the quality of response and minimize hallucinations by making the retrieval context-aware.

- Case Study 1: Multi-turn conversation User Query: "So what about its side effects?" Context (Previous Turn) "Tell me about ibuprofen" Vanilla RAG Retrieval: General documents about side effects of drugs (mixed drugs, irrelevant entries). CG-RAG Retrieval: Documents specifically about side effects of ibuprofen (e.g., nausea, stomach pain). Vanilla RAG Output: "Common side effects include nausea and dizziness, and it varies with which drug you took." CG-RAG Output: "Ibuprofen can

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

cause stomach pain, nausea and, in rare cases, bleeding in the stomach." Analysis: Vanilla RAG in which no dialogue context is included, generic information retrieved In contrast to this, CG-RAG focuses on utilizing context history to obtain targeted documents to compute a more precise and grounded answer.

- Case Study 2: Domain Specific Question Answering User Query: "What is the Penalty of Section 420?" Context: Legal domain query Vanilla RAG Retrieval: Mixed legal sections and unrelated criminal codes. CG-RAG Retrieval: Specific references to fraud and cheating under section 420 of IPC. Vanilla RAG Output: Section 420 relates to cheating and may involve penalties. CG-RAG Output: Section 420 of the IPC is the one that deals with cheating and causing delivery of property by dishonest means, which shall be punishable by imprisonment for a term extending from 7 years and also a fine. Analysis: CG-RAG retrieves documents specific to the domain and high-precision, e.g. resulting in improved actual correctness and completeness. This is evidence of the importance of context-awareness of retrieval in specialized domains [18].
- Case Study 3: Long Context Question Answering User Query: "Why has the project failed?" Context: Multi-causes long project report Vanilla RAG Retrieval: Gets sections that are not relevant (e.g. project overview). CG-RAG Retrieval: Retrieves failure-related parts (budget issues, mismanagement). Vanilla RAG Output: "The project was faced with a number of challenges." CG-RAG Output: "The project failed because of budget overruns and timelines aggravating, and teams not working together." Analysis: CG-RAG identifies context relevant segments in long documents and helps in making answers more specific and coherent. This is in line with results showing that better retrieval yields a direct improvement in generation quality in RAG systems [1], [22].

Table 6 Qualitative Examples

Input	Baseline Answer	CG-RAG Answer	Comment
Side effects? (after ibuprofen)	General drug side effects	Ibuprofen-specific effects	Uses context correctly
Section 420 penalty	Generic answer	IPC-specific details	More precise retrieval
Why project failed?	Generic failure	Budget + delays + coordination	Better long-context reasoning

➢ *Error Analysis*

As a framework, the proposed CG-RAG has some limitations despite the improvements. False Retrieval In spite of Awareness Of Context: In some cases, the retriever can pull up partially relevant or misleading documents, particularly when there is ambiguity in the query language and a lack of context to the query. This means that retrieval errors are not eliminated entirely by context gating [3].

Over-Reliance on Your Mind's Knowledge: When context awareness gate acts as a suppressor of retrieval, the model can be quite strongly dependent on its internal parametric domain knowledge. This can sometimes cause them to hallucinate if the internal knowledge is outdated or incomplete. Balancing between retrieval and generation is one of the main challenges [1], [23].

Dealing With Very Long or Noisy Contexts: In some cases with extremely long or noisy context (for example, long documents or irrelevant history of a conversation), the context encoder is prone to forget the most salient information. This can lead to poor retrieval quality and ultimately impact generation performance.

Ambiguous / Underspecified Queries: When the intent from user queries aren't clear, even context aware retrieval might have trouble deciphering the intent. This illustrates the need for work in query disambiguation and interactive clarification mechanisms in the future.

VIII. DISCUSSION

The experimental and qualitative results show that the concept of context-awareness into Retrieval Augmented Generation (RAG) works really well in enhancing the natural language understanding. However, these improvements have significant trade-offs that have to be carefully considered when designing practical systems.

One central trade-off is between the precision of the retrieval and the flexibility of the generation. By adding a context awareness gate, CG-RAG will prioritise resultantly highly relevant documents which improves factual grounding reducing hallucinations. However, if too much filtering is applied, it could result in the model limiting information retrieving during the process, and thus limit the model's capacity to generate creative or generalized responses. On the other hand, recalling a larger set of documents with higher diversity could increase flexibility but could add noise and adversely affect answer accuracy. This trade-off underlies the need of striking a balance between the precision and coverage attained in retrieval strategies [1], [14].

Another important consideration is that of efficiency versus faithfulness. Adaptive retrieval mechanisms, such as the proposed context gate, minimize the unnecessary retrieval and optimize pushing to increase computational efficiency. However, in instances where the retrieval is suppressed, it may be more likely for the model to rely on its internal understanding of parametrics more, which may lead to hallucinations or outdated responses. Altering the opportunity so that the system is kept factual while charging a minimum computational charge continues to be a crucial problem within RAG-based systems [4], [23].

The proposed CG-RAG framework however shows great potential for generalization in other domains, such as enterprise knowledge systems, law and medicine applications

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

and real-time, web-based question answering. By using information on context awareness, context-aware retrieval delivers the retrieval of information context-aware at a per user level and coarse-grained that allows for an improved adaptation to domain-specific demands and user intent than it would do for standard RAG approaches. This flexibility is especially useful in dynamic environments, where knowledge is regularly updated, while context is an important feature in interpretation [18].

Despite these advantages, there are a number of limitations. First, the quality and coverage of external corpus determine the performance of CG-RAG to a large extent. Poorly curated data sets, or incomplete data sets can result in incorrect data retrieval using poor performance. Second, construction and maintenance of large-scale retrieval indices can cost a lot, particularly for real-time applications. Third, relying on knowledge sources outside of your organization introduces privacy and security concerns, especially in enterprise or sensitive domains where data access needs to be tightly controlled. These challenges are well-known from the recent RAG research and raises importance about robust, efficient and secure retrieval mechanisms [1], [3].

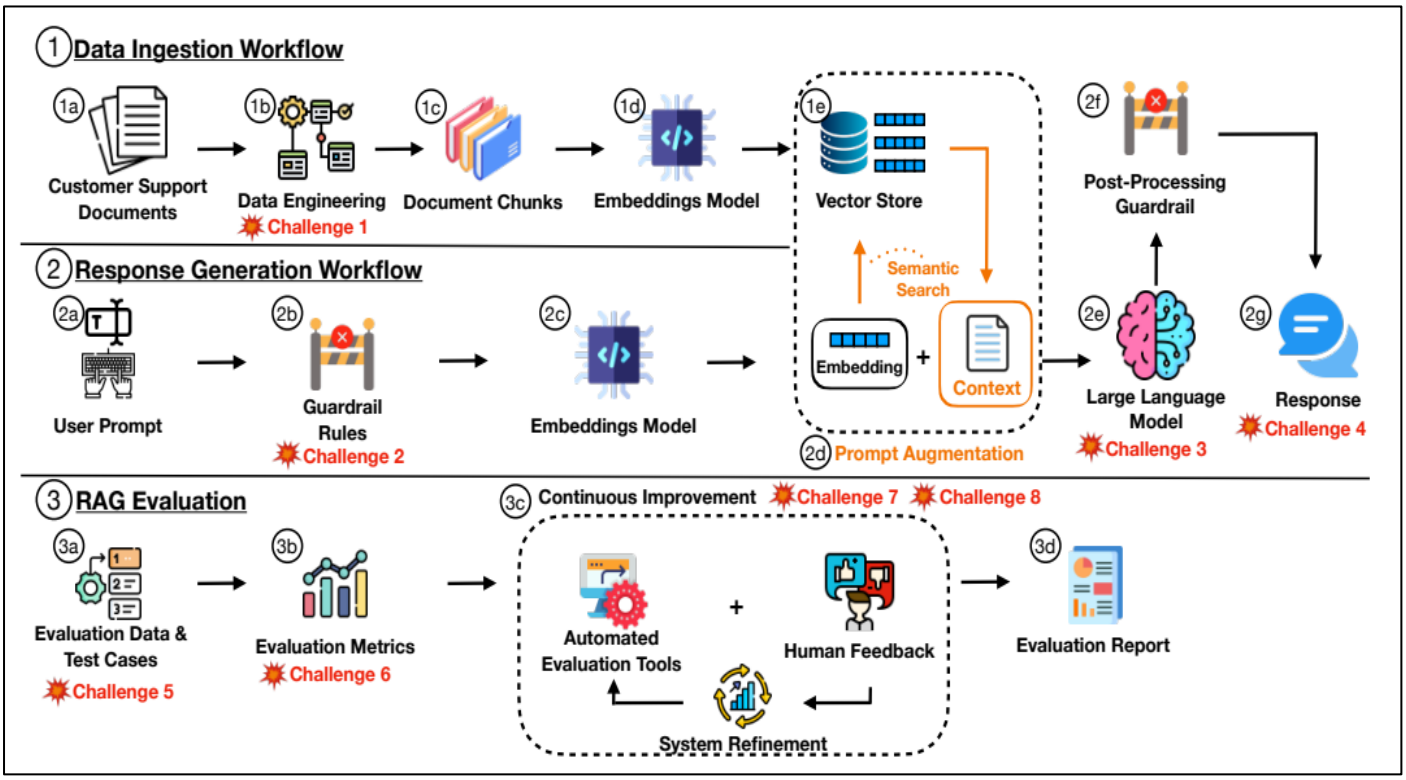

Fig 6 An Overview of the Retrieval-Augmented Generation (RAG)-Based VA Developed for Customer Support.

- Step 1 describes the process of data preparation for creating a vector knowledge base which will be used for promising augmentation.
- Step 2 describes the response generation workow that uses prompt augmentation with a large language model (LLM)
- Step 3 provides the automated evaluation process of RAGVA evaluation

IX. CONCLUSION AND FUTURE WORK

In this paper, we introduced Context-Gated Retrieval-Augmented Generation (CG-RAG), a new framework that aims to improve context-aware natural language understanding by adding a gate of context-aware to the retrieval process. The novelty of our technique is that it dynamically explores contextual information, such as dialogue history and metadata about a specific task, to guide retrieval and better align user intent and retrieved information. Through numerous studies on question answering, conversations, and tasks specific to domains, we found that CG-RAG was able to show steady advancements compared to strong baselines including RAG and standalone LLMs. In particular, using our approach we have obtained considerable improvements in terms of Exact Match (EM) as well as F1 score, context relevancy, and hallucination reduction, which reflect the effectiveness of context-aware retrieval for improving both accuracy and reliability [1], [22].

In spite of these promising results, however, there are several avenues for future research. First, adaptive retrieval architectures can be improved further by more sophisticated decision mechanism like reinforcement learning-based controllers to improve balance between retrieval and generation. Second, the extension of CG-RAG to multi-source and graph-based retrieval frameworks might allow to perform more structured reasoning over heterogeneous KBs. Third, adding multi-hop reasoning capability means the model will be able to work with complex queries that require reasoning across multiple documents.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26mar1756

Additionally, privacy preserving retrieval mechanisms is also an important direction, especially for enterprise and sensitive applications, where accessing external knowledge securely is very important. Finally, there is a need for more robust and standardized approaches to in context utilization, including tools that explicitly measure how well models utilize context at retrieval time and generation time. Solving these issues will lead to further development of reliable, efficient, and context-aware RAG systems for real-world applications [1], [4].

REFERENCES

[1]. Y. Gao, Y. Sun, Z. Li, and Y. Chen, "Retrieval-Augmented Generation for Large Language Models: A Survey," *arXiv preprint arXiv:2312.10997*, 2023.

[2]. C. Sharma, "Retrieval-Augmented Generation: A Comprehensive Survey of Architectures, Enhancements, and Robustness Frontiers," *arXiv preprint*, 2025.

[3]. A. Brown, M. Roman, and B. Devereux, "A Systematic Literature Review of Retrieval-Augmented Generation: Techniques, Metrics, and Challenges," *arXiv preprint*, 2025.

[4]. A. Gan, H. Li, and J. Zhang, "Retrieval Augmented Generation Evaluation in the Era of Large Language Models: A Comprehensive Survey," *arXiv preprint*, 2025.

[5]. Z. Li, Y. Gao, and X. Wang, "Retrieval-Augmented Generation for Educational Applications: A Survey," *Computers & Education: Artificial Intelligence*, 2025.

[6]. P. Omrani, A. Khosravi, and M. Rahmani, "Hybrid Retrieval-Augmented Generation Approach for LLM Query Response Enhancement," in *Proc. IEEE Int. Conf. on Intelligent Computing and Wireless Communications (ICWC)*, 2024.

[7]. B. Zhan, Y. Liu, and H. Chen, "RARoK: Retrieval-Augmented Reasoning on Knowledge for Medical Question Answering," in *Proc. IEEE Int. Conf. on Bioinformatics and Biomedicine (BIBM)*, 2024.

[8]. Y. Morales-Martínez, J. Pérez, and L. Gómez, "Application of Retrieval-Augmented Generation Systems in Software Engineering Education," *Int. J. Combinatorial Optimization Problems and Informatics*, 2025.

[9]. R. Yang, "RAGVA: Engineering Retrieval-Augmented Generation Applications," *Information and Software Technology*, 2025.

[10]. P. Jiang, "Comparative Study of Retrieval-Augmented Generation and Chain-of-Thought Reasoning in Large Language Models," *Engineering Applications of Artificial Intelligence*, 2025.

[11]. Y. Zhao, X. Liu, and K. Wang, "ReCode: Improving LLM-Based Code Repair with Fine-Grained Retrieval-Augmented Generation," *arXiv preprint*, 2025.

[12]. S. Kumar, R. Patel, and A. Singh, "Robust Implementation of Retrieval-Augmented Generation via Computing-in-Memory," in *Proc. ACM/IEEE Design Automation Conf.*, 2025.

[13]. E. Karakurt, "Retrieval-Augmented Generation and Large Language Models: Trends and Challenges," *Applied Sciences*, vol. 15, no. 3, 2025.

[14]. M. Klesel, T. Müller, and S. Wagner, "Retrieval-Augmented Generation: Concepts and Applications," *Springer*, 2025.

[15]. E. Karakurt, "Retrieval-Augmented Generation and Large Language Models: A Bibliometric Analysis," *Preprints*, 2025.

[16]. Y. Gao, H. Sun, and Z. Li, "LLM-Based Retrieval-Augmented Generation for 6G Wireless Networks," 2025.

[17]. D. He, Q. Wang, and L. Zhang, "Dynamic Retrieval-Augmented Generation of Ontologies (DRAGON-AI)," *Journal of Biomedical Semantics*, 2024.

[18]. H. Wang, Y. Liu, and X. Chen, "Retrieval-Augmented Generation with Conflicting Evidence," in *Findings of ACL*, 2025.

[19]. Q. Leng, Z. Zhao, and Y. Li, "On the Performance of Long-Context Retrieval-Augmented Generation in Large Language Models," 2024.

[20]. A. Leto, M. Rossi, and F. Bianchi, "Toward Optimal Search and Retrieval for RAG Systems," 2024.

[21]. P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-T. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[22]. O. Ram, Y. Levine, B. Efrat, D. Chen, and O. Levy, "In-Context Retrieval-Augmented Language Models," *Transactions of the Association for Computational Linguistics (TACL)*, 2023.

[23]. K. Shuster, S. Poff, M. Chen, D. Kiela, and J. Weston, "Retrieval Augmentation Reduces Hallucination in Conversation," 2021.

[24]. Y. Luan, J. Eisenstein, K. Toutanova, and M. Collins, "Sparse, Dense, and Attentional Representations for Text Retrieval," *TACL*, 2021.

[25]. W. Shi, S. Zhou, and Z. Chen, "Retrieval-Augmented Language Models in Natural Language Processing," in *Proc. NAACL*, 2024.