Hybrid RAG for Factual and Reliable Question Answering Using Quantized LLMs

Team Members

Mahesh Babu Akula

Naga Goutham Nidumukkala

Surya

Problem Statement

The rapid adoption of Large Language Models (LLMs) in question answering (QA) and information retrieval (IR) systems has revolutionized how users access knowledge by enabling natural, context-aware responses. Despite these advancements, a persistent challenge remains: LLMs frequently generate hallucinated, unverifiable, or factually inconsistent content, especially when operating without explicit grounding in external data. This limits their reliability for knowledge-intensive tasks such as research assistance, healthcare, or enterprise knowledge discovery.

Traditional Retrieval-Augmented Generation (RAG) frameworks attempt to alleviate this issue by incorporating external document retrieval into the response generation process. However, most RAG systems depend on a single retrieval paradigm — either sparse (lexical) methods like BM25 or dense (semantic) embeddings — each of which has inherent limitations. Sparse retrievers struggle with semantic generalization and synonymy, while dense retrievers may overlook exact lexical matches, leading to partial or incomplete evidence sets. As a result, current RAG implementations often fail to retrieve comprehensive and contextually aligned evidence, which directly impacts the factual grounding of generated answers.

To address these shortcomings, this project proposes a Hybrid RAG architecture that combines sparse retrieval (BM25) and dense retrieval (Chroma vector embeddings) to leverage the complementary strengths of both methods. The hybrid retrieval layer ensures a richer and more contextually complete evidence set for the LLM to reason over. Additionally, the system integrates a verification-driven fusion mechanism, wherein the quantized LLM not only generates responses but also performs cross-evidence verification to validate factual claims against retrieved documents. This dual-phase generation and verification process aims to minimize hallucinations, improve factual consistency, and enhance interpretability in general-domain question answering.

By adopting this approach, the project seeks to contribute toward trustworthy and resource-efficient generative retrieval systems, bridging the gap between classical information retrieval and modern neural language models.

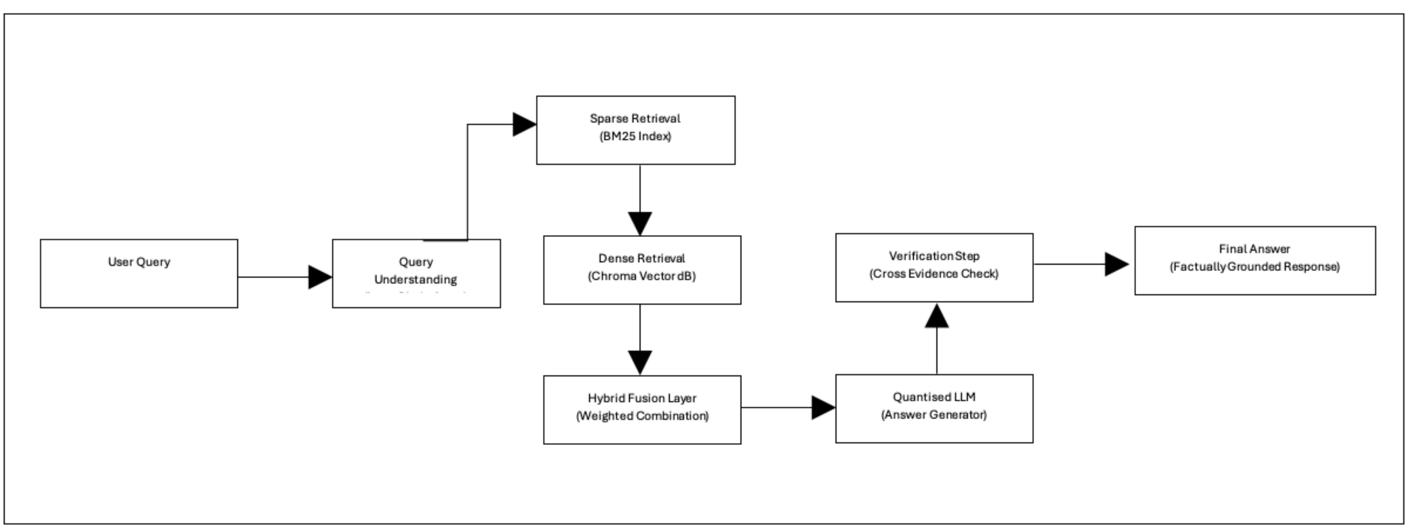
Data Sources

- **Primary Dataset:** Public QA datasets such as **Natural Questions**, **SQuAD**, or **MSMARCO** (static corpus for initial experiments).
- **Scaling Phase:** Expand to real-world datasets (e.g., Wikipedia or news articles) after validating prototype performance.

Tools/Frameworks

- **Quantized Open-Source LLM:** For efficient local inference and response generation.
- **LangChain:** To orchestrate the RAG pipeline, manage agent behavior, and structure prompt templates.
- **Chroma Vector Database (FAISS backend):** For dense vector retrieval and similarity search.
- **BM25 (via Whoosh or Elasticsearch):** For sparse text retrieval and evidence ranking.
- **Python, Hugging Face Transformers, OpenAI Evaluation Toolkit (optional):** For implementation and analysis.

Methodology



Evaluation Plan

The evaluation emphasizes **qualitative human assessment** of generated answers:

- **Criteria:**
  - **Relevance** – How well the answer addresses the query.
  - **Faithfulness/Factuality** – Whether the response is grounded in retrieved evidence.
  - **Hallucination Severity** – Graded as None, Minor, or Major.
  - **Usefulness** – Overall helpfulness and completeness of the response.
- **Process:**
  - Human annotators (3 per query) will rate 300–500 QA pairs.
  - Measure inter-annotator agreement (e.g., Fleiss' kappa).
  - Compare against baselines (BM25-only, dense-only, and standard RAG).
- **Supplementary Metrics:** Recall@k for retrieval accuracy and entailment-based factuality scores for verification.

Timeline

Phase	Duration	Tasks
Weeks 1–2	Literature review, dataset setup, indexing with BM25 & Chroma	• Conduct a literature review on Retrieval-Augmented Generation (RAG), hybrid retrieval (BM25 + dense), and hallucination mitigation in LLMs. • Study existing frameworks like DPR, ColBERT, and RAG (Lewis et al.). • Select benchmark QA datasets (SQuAD, Natural Questions, MSMARCO). • Preprocess dataset: clean, tokenize, and segment documents. • Build BM25 index (using Whoosh or Elasticsearch).
Weeks 3–4	Implement hybrid retrieval fusion, integrate quantized LLM using LangChain	• Implement hybrid retrieval fusion (BM25 + Chroma). • Tune retrieval weights (e.g., $\alpha \cdot$BM25 + $(1–\alpha) \cdot$dense similarity). • Integrate LangChain agent to manage query flow and retrieval orchestration. • Connect quantized LLM (e.g., Mistral, Llama 2 7B Q4) to generate answers using retrieved documents. • Implement RAG baseline (without verification) for comparison. • Design modular prompts for retrieval and generation.
Weeks 5–6	Develop verification-driven generation and prompt templates	• Implement **cross-evidence verification module** — comparing generated claims to retrieved passages. • Integrate re-query or correction mechanism for hallucinated responses. • Develop prompt engineering strategies for fact-grounded generation (e.g., chain-of-thought + citation prompts). • Build a logging mechanism to track model responses and verification outcomes. • Conduct initial tests on small sample queries.

Weeks 7–8	Run experiments, collect human evaluation data, refine system	• Conduct systematic experiments on 300–500 QA pairs. • Collect human evaluation ratings for: Relevance, Faithfulness, Hallucination severity, and Usefulness. • Prepare annotation guidelines and recruit 2–3 evaluators per sample. • Measure inter-annotator agreement (Fleiss' kappa). • Compare performance with baselines: BM25-only, Dense-only, and Standard RAG. • Tune fusion weights, retrieval size (Top-k), and prompt design based on results.
Weeks 9–10	Analyse results, document findings, finalize research report and demo	• Analyze results qualitatively and quantitatively (retrieve precision, hallucination reduction rates, factuality improvement). • Perform ablation studies (e.g., without verification, without hybrid fusion). • Summarize key findings and insights. • Prepare visualizations: score distributions, comparison graphs, and architecture diagram. • Write final research report (abstract, methodology, results, limitations, future work). • Develop demo application using python and presentation slides for submission.