RAG Implementaions

By 

Corrective RAG (CRAG)

CRAG was designed because **vector search sometimes gives irrelevant hits** even if the user query is clear.

CRAG adds a *corrector model* that:

- Filters out irrelevant retrieved chunks
- Re-ranks the relevant ones
- Optionally routes to keyword search or a different retriever

Example Workflow
User Query:
“Explain the formula for interest rate in the finance policy document.”

Vector Retriever Outputs (Messy):
Chunk 1: “Company holidays 2023” (irrelevant)
Chunk 2: “Interest rate formula …” (relevant)
Chunk 3: “Finance team org chart” (irrelevant)
Corrective Model Output:
Removes chunk 1 & 3
Re-ranks chunk 2 as top candidate

LLM Answer:
Uses clean and accurate context.

Self-RAG with Reflection (Self-Evaluating RAG)

LLM validates its own output using **reflection** and **self-correction** loops.

This is different from CRAG because here the **LLM examines its own answer**, not retrieval.

Example Flow

Query: “What does Section 4 of the policy document say?”

Draft Answer:
“Section 4 is about Employee Benefits.”

Reflection Step:
LLM reviews the retrieved text → notices Section 4 is actually “Quality Standards”.

Corrected Answer:
“Correction: Section 4 describes Quality Standards…”

Hybrid RAG (Semantic + Keyword Search)

Combine semantic search (vectors) + keyword search (BM25/Elastic)
Because vectors **lose exact-match precision**, and BM25 misses semantic meaning.

Example

User Query:
“What is the compensation limit under clause 11.2?”

- **Vector search** → finds chunks about “compensation policies”
- **Keyword search (BM25)** → finds the **exact clause 11.2** text containing numbers

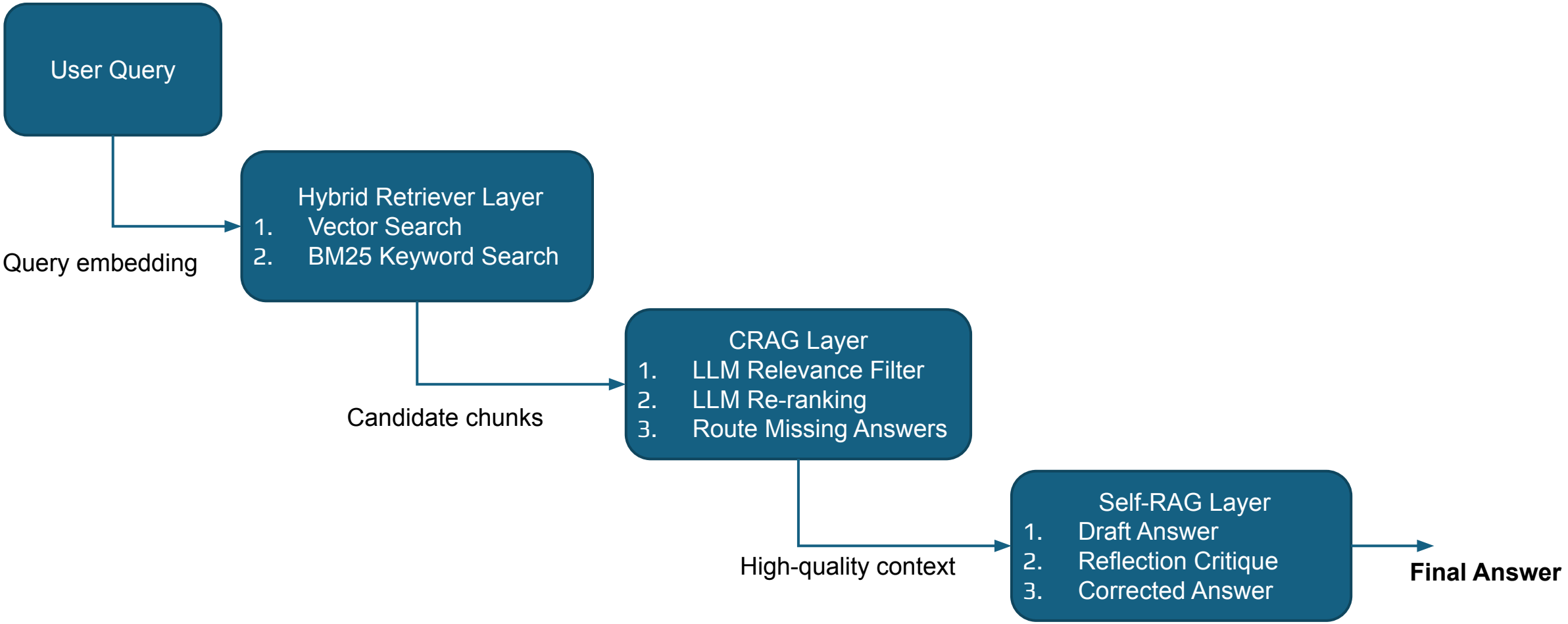
Hybrid result:
LLM gets both:
- Semantically similar data
- Exact-number precise clause

RAG Implementations

Technique	Purpose	Main Strength	Example Use
Corrective RAG	Fix retrieval errors	Filters irrelevant chunks	Policies, PDFs with noise
Self-RAG w/ Reflection	Fix LLM answer hallucination	Self-correcting answers	Q&A needing high accuracy
Graph RAG	Structured querying & reasoning	Uses relationships	Org charts, legal docs
Hybrid RAG	Combine semantic + keyword search	Exact + semantic accuracy	Clauses, numbers, terms

Unified RAG Pipeline (Hybrid + CRAG + Self-RAG)

Layer 1 → Hybrid Retrieval (Semantic + Keyword)
Layer 2 → CRAG (Corrective Filtering + Reranking)
Layer 3 → Self-RAG (Answer Refinement + Hallucination Removal)



Thank you