Biomedical Evidence Retrieval with Agentic RAG and Dual Text Encoders

Dhruv Goyal, Ema Seibert, Ryan Ding, Matteo Migliarini, Kevin Zhu

NEURAL INFORMATION PROCESSING SYSTEMS
🌟 NORA'25

Workshop on KNOwledge GRaphs & Agentic Systems Interplay

Agentic RAG framework for evidence retrieval, using iterative query refinement across notes.

Retrieval-Augmented Generation (RAG) has emerged as a leading approach for evidence-based retrieval, combining dense retrieval with generation. In medicine, this paradigm was adapted domain-specific models like BioBERT to handle specialized terminology, yet traditional RAG pipelines are often static, retrieving once without adapting their reasoning. A more advanced paradigm, Agentic RAG, extends this by embedding autonomous decision-making and iterative reflection into the retrieval loop.

- dual domain-specific encoders
- self-critique loops
- benchmarks on established biomedical QA datasets
- Patients-PMC benchmark to assess generalization for clinical discovery
- corrective feedback or query routing to achieve more adaptive reasoning
- adaptive retrieval for clinical decision support

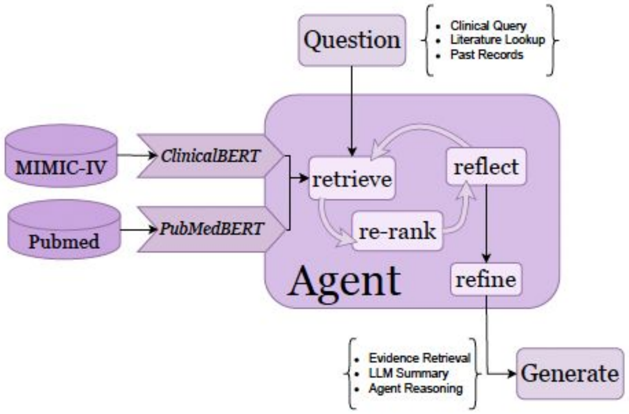


Figure 1: Hybrid biomedical RAG with iterative self-critique. Evidence from PubMed (literature) and MIMIC-IV (clinical notes) is retrieved via domain-specific encoders and re-ranked. An agent cycles between reflect and refine, yielding a final, evidence-grounded response.

Methodology

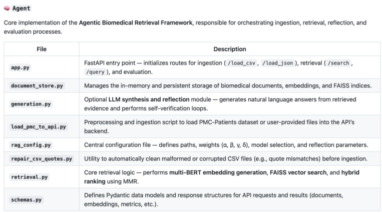
Embedding clinical notes

Our system employs an agentic RAG framework that iteratively refines search queries and integrates evidence from biomedical literature (PubMed) and clinical notes (MIMIC-IV). The core is a dual encoder as shown in figure 1. We encode queries and documents using two specialized models: PubMedBERT for literature and ClinicalBERT for clinical notes, enabling parallel searches. Retrieved documents are then merged and refined using a cross-encoder reranker.

Results

Testing and Comparing to Baselines

- We evaluate our agentic retrieval system on the PMC-Patients benchmark; covering Patient-to-Article Retrieval (PAR) and Patient-to-Patient Retrieval (PPR); and the reasoning-free setting of PubMedQA.
- As shown in Table 1, our framework achieves competitive results across all tasks. While the model also performs competitively on the more challenging PPR task, the PAR scores highlight the system's strength in precise evidence matching.
- On PubMedQA, our framework attains an accuracy of 82.09%, outperforming key baselines like BioBERT (80.80% shown in table 2.



🧠 Agent

Core implementation of the **Agentic Biomedical Retrieval Framework**, responsible for orchestrating ingestion, retrieval, reflection, and evaluation processes.

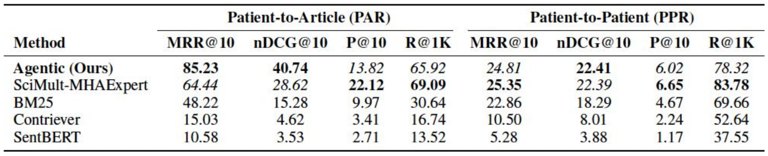
File	Description
`app.py`	FastAPI entry point — initializes routes for ingestion (`/load_csv` , `/load_json`), retrieval (`/search` , `/query`), and evaluation.
`document_store.py`	Manages the in-memory and persistent storage of biomedical documents, embeddings, and FAISS indices.
`generation.py`	Optional **LLM synthesis and reflection** module — generates natural language answers from retrieved evidence and performs self-verification loops.
`load_pmc_to_api.py`	Preprocessing and ingestion script to load PMC-Patients dataset or user-provided files into the API's backend.
`rag_config.py`	Central configuration file — defines paths, weights (α, β, γ, δ), model selection, and reflection parameters.
`repair_csv_quotes.py`	Utility to automatically clean malformed or corrupted CSV files (e.g., quote mismatches) before ingestion.
`retrieval.py`	Core retrieval logic — performs **multi-BERT embedding generation**, **FAISS vector search**, and **hybrid ranking** using MMR.
`schemas.py`	Defines Pydantic data models and response structures for API requests and results (documents, embeddings, metrics, etc.).

Conclusion

Developing more reliable tools for evidence based medicine

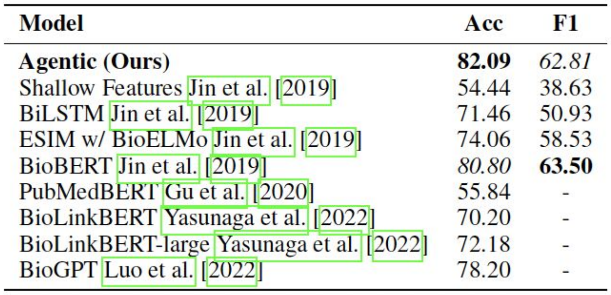
We demonstrated the effectiveness of an agentic RAG framework for complex biomedical retrieval. Our system achieved competitive performance on the PMC-Patients and PubMedQA benchmarks, highlighting the advantages of agentic strategies over static pipelines.

Table 1: Results for Patient-to-Article Retrieval (PAR) and Patient-to-Patient Retrieval (PPR) on the PMC-Patients dataset. Best results are in **bold**, second best are in *italics*.



	Patient-to-Article (PAR)				Patient-to-Patient (PPR)			
Method	**MRR@10**	**nDCG@10**	**P@10**	**R@1K**	**MRR@10**	**nDCG@10**	**P@10**	**R@1K**
Agentic (Ours)	**85.23**	**40.74**	*13.82*	*65.92*	*24.81*	**22.41**	*6.02*	*78.32*
SciMult-MHAExpert	*64.44*	*28.62*	**22.12**	**69.09**	**25.35**	*22.39*	**6.65**	**83.78**
BM25	48.22	15.28	9.97	30.64	22.86	18.29	4.67	69.66
Contriever	15.03	4.62	3.41	16.74	10.50	8.01	2.24	52.64
SentBERT	10.58	3.53	2.71	13.52	5.28	3.88	1.17	37.55

Table 2: Comparison of reasoning-free baselines on the PubMedQA dataset.



Model	**Acc**	**F1**
Agentic (Ours)	**82.09**	*62.81*
Shallow Features Jin et al. [2019]	54.44	38.63
BiLSTM Jin et al. [2019]	71.46	50.93
ESIM w/ BioELMo Jin et al. [2019]	74.06	58.53
BioBERT Jin et al. [2019]	*80.80*	**63.50**
PubMedBERT Gu et al. [2020]	55.84	-
BioLinkBERT Yasunaga et al. [2022]	70.20	-
BioLinkBERT-large Yasunaga et al. [2022]	72.18	-
BioGPT Luo et al. [2022]	78.20	-