Queryable Shared Reference Repository

Atyab Hakeem, Kishan Sathish Babu, Naga Kushal Ageeru, Pranav Kanth Anbarasan
https://github.com/hakeematyab/Queryable-Shared-Reference-Repository

1. Summary

1.1 Problem Description

Research groups must manage an ever-growing volume of scientific literature. While reference managers allow storage and basic retrieval, they lack intelligent, context-aware querying that integrates both paper content and metadata. Large Language Models (LLMs) can enhance search and synthesis but raise privacy concerns and introduce risks of hallucination and inconsistent accuracy. This project aims to address these limitations by developing a on-device, shared, queryable repository of scientific papers that enables natural language queries and minimizes fabricated outputs through careful design and evaluation.

Phase 1 established the data processing pipeline, benchmarked and selected optimal embedding models and chunking strategies, and developed a hallucination detection framework. Phase 2 will build on this foundation by employing advanced data processing, integrating reranking models to improve retrieval quality, implementing generation models for response synthesis, developing agentic workflows for complex queries, evaluating the complete system, analyzing and implementing hallucination mitigation strategies, and, if time permits, deploying a Slack bot interface and user-testing.

1.2 Dataset

The dataset currently consists of 300 curated scientific papers in PDF form, along with supplementary webpages and bibliographic files (e.g., .bib). Both full text (abstracts, methods, results, figures, tables) and metadata (title, authors, journal, year, DOI, citations) are extracted and structured for flexible querying. Early processing includes text cleaning, formatting corrections, and semantic chunking to optimize retrieval. While we currently have 300 papers, the dataset may scale to around 3,000**–**10,000 papers.

1.3 Related work:

Semantic Scholar, developed by the Allen Institute for AI, indexes over 200 million academic papers using machine learning and citation-informed embeddings to provide AI-enhanced search and research recommendations. CORD-19 aggregated 1M+ coronavirus papers through Semantic Scholar with structured full-text parses, facilitating numerous COVID-19 text mining systems. However, both systems' cloud-based architectures pose privacy risks for sensitive research data and lack integrated hallucination detection mechanisms to ensure factual accuracy. Our work addresses these limitations by developing a secure, on-premises RAG system with query-type-specific hallucination detection, specifically designed for research groups managing proprietary scientific literature.

2. Methods

2.1 Architecture Overview

The system architecture comprises three components: document processing, hybrid retrieval, and agentic generation. Documents are processed to extract and clean text, segmented using hybrid chunking, and stored in a vector database (embedded chunks with metadata). The retrieval strategy would employ parallel semantic search (Karpukhin et al., 2020) and lexical search (Robertson & Zaragoza, 2009), each retrieving M=15 candidates. These are reranked using a cross-encoder, returning the top k=5 results, which are fed to the generation model. The generation framework would operate on a Reasoning-Action (ReAct) architecture (Yao et al., 2023), where the generation model acts as a reasoning agent with modular tool access to RAG retrieval. The agent determines appropriate actions—answering directly or invoking RAG with context-enriched queries. This enables multi-hop reasoning where the agent iteratively refines queries using retrieved data, avoiding rigid retrieval-generation loops. The modular design allows dynamic adaptation based on query complexity, from simple factual questions to complex multi-document synthesis–meeting the requirements of the lab. Due to time constraints, we limited the scope of the study to extensively evaluate the first two components.

2.2 Data Processing

Phase 1 explored the extraction of text from the documents using two main approaches: programmatic text extraction (Singer-Vine, 2024) without post-processing, and small language model-based text extraction with built-in processing (Nassar et al., 2025). Simple text extraction yielded significantly better quantitative metrics compared to small language model processing. However, quantitative analysis of the resulting data from the extracted text revealed messy data. Although this wouldn't impact the word-based evaluation metrics, it likely would affect the quality of the outputs with the generation model.

To combat this, Docling (Deep Search Team, 2024) was employed. Docling is an open-source document conversion toolkit that leverages specialized vision models for page layout analysis, reading order detection, and table structure recognition for post-processing of the extracted text. For text extraction, it supports an Optical Character Recognition (OCR) mode, which leverages OCR models (E.g, RapidAI Team, 2021) and a non-OCR mode, which programmatically extracts texts from digital PDFs. Since our dataset consists of digital documents, non-OCR mode not only provides faster processing but also yields better quality outputs.

2.3 Synthetic Data Generation

Synthetic evaluation data was generated using Llama 3.3 70B (Meta, 2024) at half precision for the evaluation of embedding models, chunking strategies, and hallucination mitigation strategies. To evaluate the embedding models and chunking strategies, a dataset of questions, along with the exact excerpt to answer the corresponding questions, was generated. To evaluate the hallucination mitigation strategies, a similar dataset with questions and excerpts was generated. The generated queries fell into three categories: answerable queries where the context contained the answer, unanswerable queries where the context lacked necessary information, and borderline queries where answers could be inferred from the context but required information beyond the explicitly provided scope.

2.4 Chunking & Embedding

Chunking is a process of segmenting large documents into smaller, semantically coherent units for efficient retrieval and processing. Docling provides two main chunking approaches optimized for its processed data: the hierarchical chunker and the hybrid chunker. The hierarchical chunker divides text into smaller segments based on section and subsection hierarchy. It leverages rich information about page layout and section hierarchy, and performs intelligent segmentation. The hybrid chunker builds upon the hierarchical chunker by merging undersized chunks and splitting oversized chunks while respecting the context window of the embedding model.

Embedding models convert text into dense vector representations that capture the semantics of the text. Similarity search can be performed on these embeddings to retrieve similar text. Phase 1 benchmarked domain-specific and general embedding models. However, quantitative analysis revealed fragmented data, prompting us to expand our search to Gemma (Schechter Vera et al., 2025), a model ranked high on the MMTEB leaderboard (Enevoldsen et al., 2025) that has a larger context window.

2.5 Retrieval & Reranking

Retrieval is the process of fetching documents relevant to a given query. Retrieval can be keywords-based through a sparse vector search (Robertson & Zaragoza, 2009) or semantic through dense vector search (Karpukhin et al., 2020). While keyword search offers low latency and excels at exact term matching, it fails to capture semantic relationships. Conversely, semantic search understands contextual meaning but incurs higher computational overhead. As such, we employ a hybrid strategy employing parallel semantic and lexical search. Best Matching 25 (BM25) (Robertson & Zaragoza, 2009) was chosen due to its high performance and ubiquity for lexical search. For semantic search, previously stated embedding models we employed to generate the dense vectors with cosine similarity to obtain relevant documents.

A key limitation of semantic and keyword search is that documents and queries are vectorized independently. While this reduces latency tremendously, key information that can only be captured by examining documents and queries jointly is lost. Cross-encoders (Reimers & Gurevych, 2019) mitigate this by encoding both jointly, but at the cost of significant latency. To balance effectiveness and efficiency, we employ a coarse hybrid search (semantic and keyword) to retrieve a large pool of candidate documents, then prioritize them using a reranker. Jina Reranker (Günther et al., 2024), GTE Reranker (Zhang et al., 2024), MS-MARCO MiniLM (Reimers & Gurevych, 2019), and EmbeddingGemma (Schechter Vera et al., 2025) were shortlisted based on the metric-resource tradeoff from the MMTEB leaderboard (Enevoldsen et al., 2025).

2.6 Generation Model

Generation models are large language models (LLMs) that produce natural language outputs conditioned on input context. In retrieval-augmented settings, their primary role is to synthesize coherent, accurate responses by grounding generation in retrieved documents rather than relying solely on parametric knowledge, thereby reducing hallucination and improving factual consistency. For our system, we require models capable of multi-turn conversation, answer synthesis from scientific text, moderate reasoning, and tool calling. To enable local, on-premises deployment that ensures data privacy, models must fit within a 20GB VRAM budget, with a total system allocation of 25GB, including all components and KV cache. Based on performance benchmarks from the Open LLM Leaderboard (Beeching et al., 2023) and compatibility with these constraints, we shortlisted Qwen3 8B (Yang et al., 2025), Llama 3.1 8B (Grattafiori et al., 2024), and Qwen3-VL 8B (Bai et al., 2025) as candidate models.

2.7 Hallucination

In the scope of this study, any generated output that is not grounded in a retrieved document is defined as a hallucination. Phase 1 explored hallucination detection, while this study will explore two hallucination mitigation strategies informed by recent research: confidence thresholding (Kalai et al., 2025) and context compression (Chroma Research, 2025).

In confidence thresholding, you instruct the model to answer only if it satisfies certain confidence threshold requirements, and that mistakes would cost more than a simple refusal. The idea is to prime the model to be conservative. To validate the effectiveness of this strategy, a generation model (Qwen Team, 2025) was inferenced to answer the questions generated synthetically. The answers were then mapped to binary classification labels and evaluated on metrics: accuracy, precision, recall, and F1 score. Four prompting styles were used: Baseline prompt simply had the question and the context, Explicit IDK (I don't know) had instructions to output IDK when applicable, Confidence Thresholding had the aforementioned instructions, and Rubric Thresholding had a defined checklist of items with a confidence requirement.

Context Rot (Chroma Research, 2025) is a phenomena where the performance of LLMs drop past a certain context length well within the context window of the model. If we determine this optimal context length, we could either stop the conversation or compress it when we reach the limit. To achieve this, the dataset used in confidence thresholding was augmented to expand the size of the context into several predefined lengths based on the percentage of the maximum context window of the generation

model. To prevent a large number of variables, only answerable and unanswerable questions with Explicit IDK prompt were experimented. The generation model was then inferenced and evaluated as in the previous experiment.

2.8 Evaluation Metrics

To evaluate the quality of chunking, two primary metrics are used: Hit Rate and Mean Reciprocal Rank (MRR). Hit Rate@K measures the quality of the retrieval of documents by computing the proportion of queries for which the retrieved K documents contain the relevant document. MRR evaluates the ranking quality by computing the mean of the reciprocals of the ranks of the relevant document. A document is considered relevant if there is a certain percentage of overlap between the excerpt (ground truth) and the retrieved document.

Hit Rate = $(1/n) \sum_{i=1}^{n} \mathbb{1}(\text{recall}_i > \tau)$ and **MRR** = $(1/n) \sum_{i=1}^{n} (1/\text{rank}_i)$

where $\text{recall}_i = |D_i \cap G_i| / |G_i|$; n = number of queries; D_i = retrieved documents for query i; G_i = ground truth documents for query i; τ = threshold; $\mathbb{1}(\cdot)$ = indicator function; rank_i = rank of the first relevant document; relevant document is defined as $\text{recall}_i > \tau$.

To evaluate the hallucination mitigation strategies, four standard binary classification metrics are used: Accuracy, Precision, Recall, and F1 Score. In the context of our study, any answer not grounded in the context (retrieved documents) is a hallucination. As such, answers to borderline questions are considered hallucinations. Precision measures the rate at which model answers correspond to answerable questions, directly quantifying hallucination avoidance. Recall measures the proportion of answerable questions that received responses, capturing the model's ability to provide information when appropriate. F1 Score balances precision and recall, and Accuracy measures overall correctness. For our use case, precision is the favored metric–to ensure answers are grounded.

Precision = TP / (TP + FP); **Recall** = TP / (TP + FN); **F1 Score** = 2 × (Precision × Recall) / (Precision + Recall)

Accuracy = (TP + TN) / (TP + TN + FP + FN)

where TP = answerable questions answered correctly; FP = unanswerable questions answered (hallucinations); FN = answerable questions not answered (missed opportunities); TN = unanswerable questions correctly abstained.

To evaluate the RAG system quality, two metrics are used: Faithfulness and Answer Relevancy. Faithfulness measures the proportion of generated claims grounded in retrieved context, directly quantifying hallucination avoidance. Answer Relevancy assesses how well responses address the original query via semantic similarity.

Faithfulness = (Supported claims) / (Total claims)

Answer Relevancy = mean(cosine_similarity(original_question, synthetic_questions_from_answer))

3. Results

3.1 Preliminary Results

Phase 1 developed a comprehensive data processing pipeline handling document files with metadata extraction. The pipeline processed 297 documents, averaging 1,782 words with highly technical vocabulary and right-skewed length distribution. We benchmarked six embedding models (SPECTER, MiniLM-L6, MPNet, SciNCL, SciBERT, PubMedBERT) across five chunking strategies (Table A.1) using synthetic data from Llama 3.3 70B. MiniLM with recursive character splitting (512 tokens, 50 overlap) achieved ~60% hit rate and ~55% MRR with the lowest parameter count and fastest query times, retrieving relevant content in top 1-2 results. Weighted normalized scores (42.5% Hit Rate & MRR, 10% parameter count, 5% search time) are presented in Figure A.1. For hallucination detection, we selected Bespoke RoBERTa based on the optimal F1 score per parameter (Figures A.2). Qualitative analysis revealed limitations with numerical claims, lengthy text, and irregular formatting. Additionally, we proposed a three-tiered hallucination reporting system calibrated on experimental results (Figures A.3 and A.4).

3.2 Embedding Models & Chunking Strategies

The expanded evaluation of embedding models across multiple chunking strategies, incorporating Docling's advanced text extraction pipeline, demonstrated approximately 11% and 13% improvements in hit rate (Figure 1) and MRR (Figure 2), respectively, relative to the Phase 1 baseline using conventional digital text extraction. The hybrid chunking strategy consistently performed best across all models, with Gemma configured with the larger context window achieving the highest metrics compared to the smaller context window configuration. Additionally, qualitative evaluation of the chunks revealed a cleaner and well-segmented text.

3.3 Retrieval & Reranking

Excluding Gemma, rerankers generally improved quality metrics over baseline semantic search (Figures 3–4). Jina Reranker achieved the highest overall performance (+6.5% Hit Rate, +10.2% MRR over baseline), while MS-MARCO offered the best efficiency tradeoff (+6.7% Hit Rate, +8.4% MRR over baseline) at 8x fewer parameters. However, both models have limited context windows, a constraint that may not surface with current synthetic evaluation data and document scale but could impact performance in the future. GTE Reranker (+5.1% Hit Rate, +10% MRR over baseline) was selected as the final choice, offering comparable performance with a larger context window. BM25 lexical search alone performed exceptionally well (Figures 5–6), and reranking further boosted the results (85.8% Hit Rate, 87.4% MRR), closely followed by the hybrid strategy with reranking (85.1% Hit Rate, 86.4% MRR).

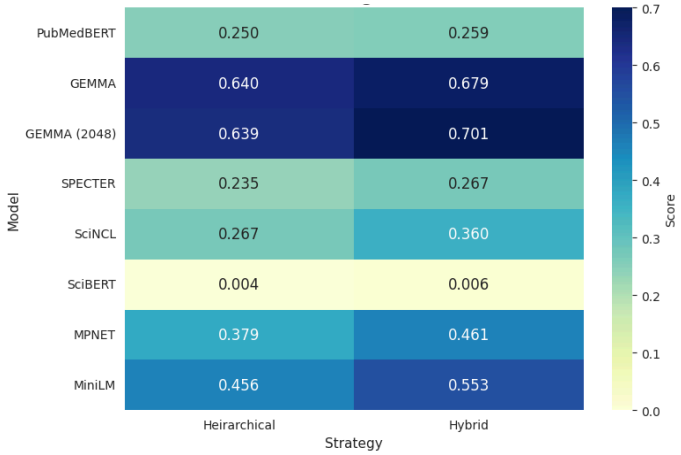

Figure 1: Heat map of Hit Rate across embedding models and chunking strategies. Gemma model with a larger configured context window coupled with a hybrid chunking strategy, outperforms other configurations.

Figure 2: Heat map of MRR across embedding models and chunking strategies. Gemma model with a larger configured context window coupled with a hybrid chunking strategy, outperforms other configurations.

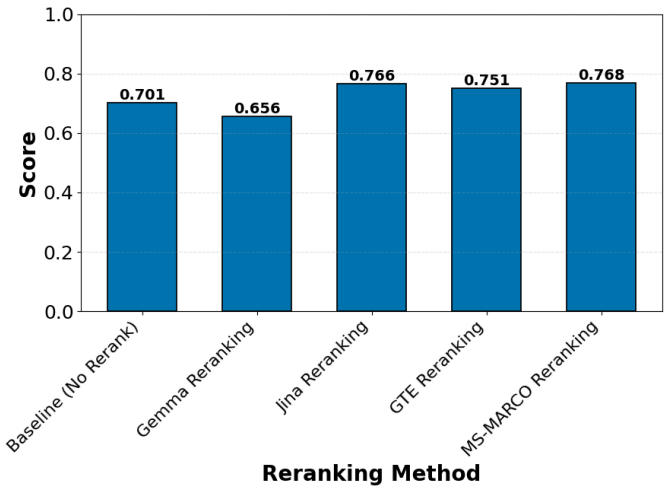

Figure 3: Comparison of Hit Rate across reranking models. MS-MARCO reranker has the highest score.

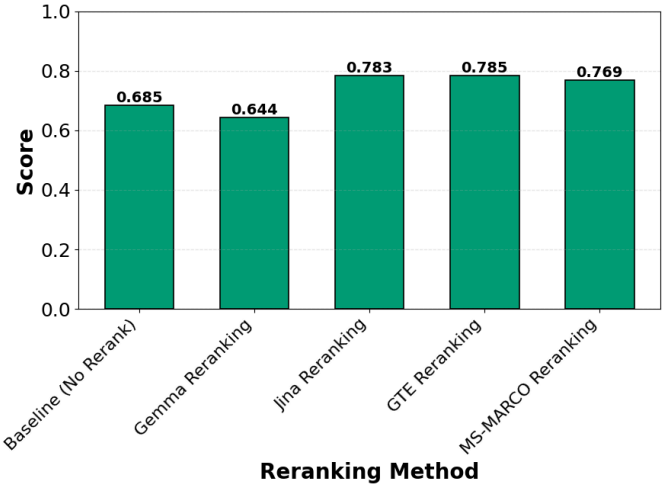

Figure 4: Comparison of MRR across reranking models. GTE reranker has the highest scores.

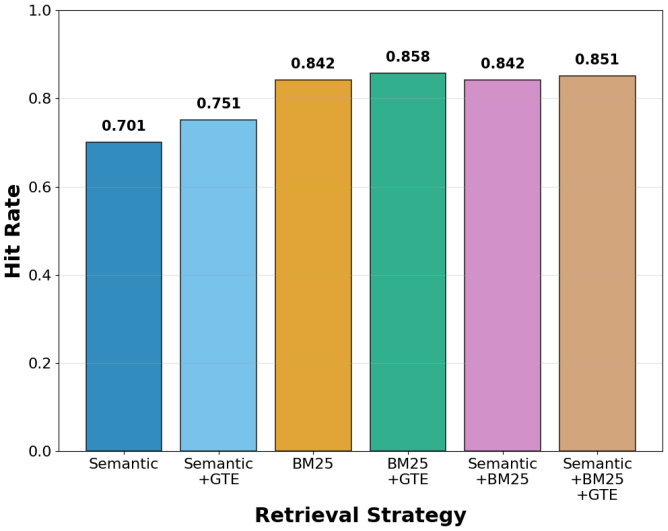

Figure 5: Comparison of Hit Rate across retrieval strategies. BM25 lexical search with GTE reranking achieves the highest scores.

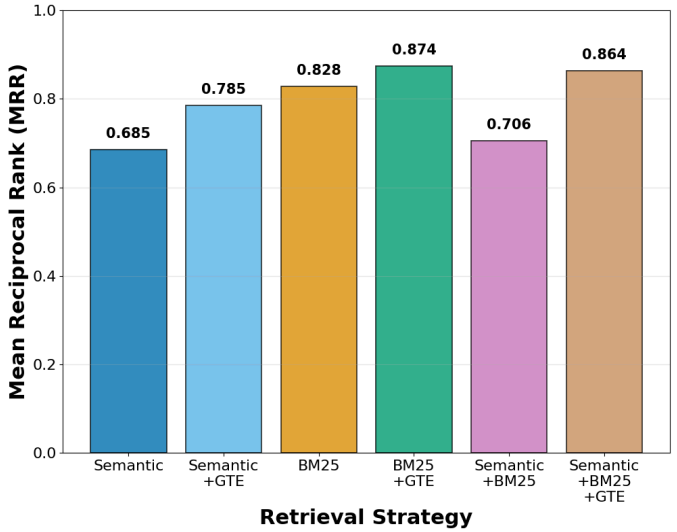

Figure 6: Comparison of MRR across retrieval strategies. BM25 lexical search with GTE reranking achieves the highest scores.

3.4 Generation Model

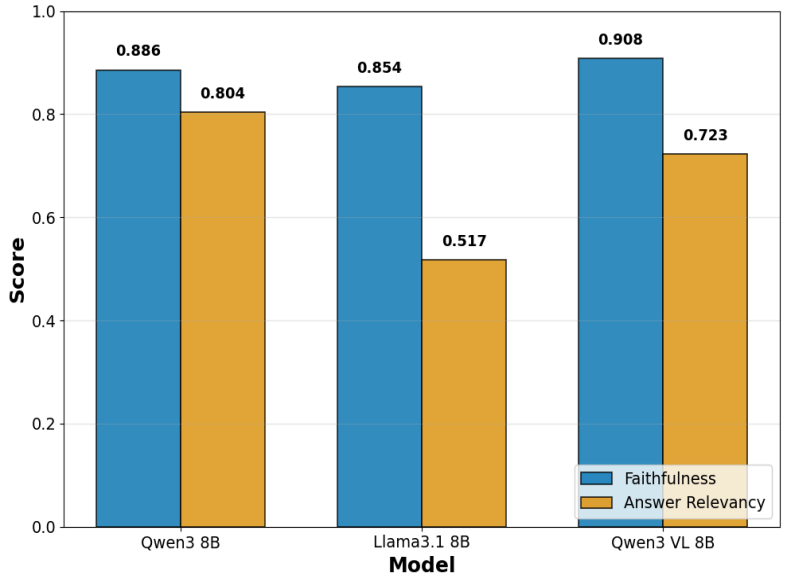

Figure 7: Comparison of Faithfulness and Answer Relevancy RAG metrics across generation models. Qwen3 8B achieves the highest average scores.

Qwen3 obtained the highest average scores (Figure 7)–it not only had a lower tendency to hallucinate, but the answers were also concise and relevant. Although Qwen3 VL achieved the highest Faithfulness, it has lower Answer Relevancy. Llama, although not too far behind in Faithfulness, has significantly lower Answer Relevance, implying that although the responses did not hallucinate much, they incorporated a large amount of unnecessary details.

3.5 Hallucination Mitigation Strategies

Comparing answerable and unanswerable queries (Figure 8), unsurprisingly, the prompt without any instructions (Baseline) elicited an answer every time from the generation model, even with unanswerable queries. Explicit IDK was more conservative in answering questions, answering about only half of the answerable questions; however, nearly all questions that were answered had backed evidence (i.e., answerable queries). Confidence thresholding, similar to Explicit IDK, achieves full precision; however, it becomes overly conservative, answering only about 20% of the answerable questions. Confidence Rubric mitigates this issue somewhat, improving the recall. As such, it would seem that a simple explicit denial instruction (Explicit IDK) would be most appropriate. Interestingly, in the case of borderline queries (Figure 9), while Explicit IDK suffers a huge drop in precision (~29%), Confidence Rubric better handles the ambiguity and suffers only a minor drop (~6%).

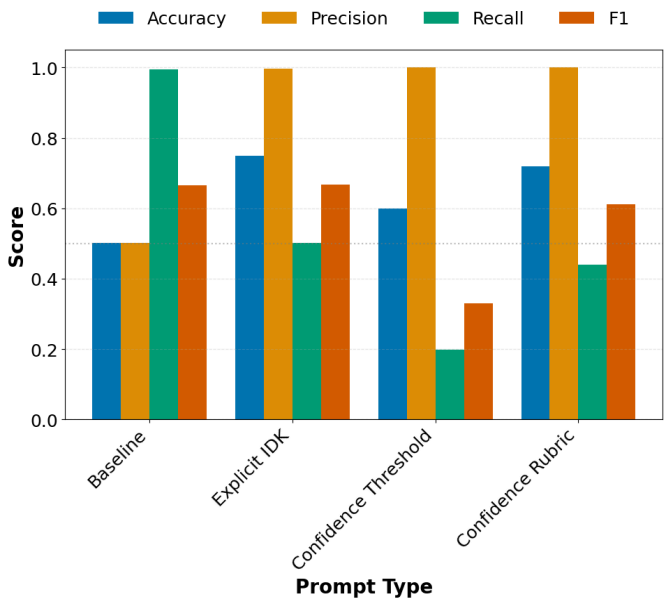

Figure 8: Comparison of binary classification metrics across prompting strategies with answerable and unanswerable queries on Qwen3 8B. Explicit IDK provides the best precision-recall tradeoff.

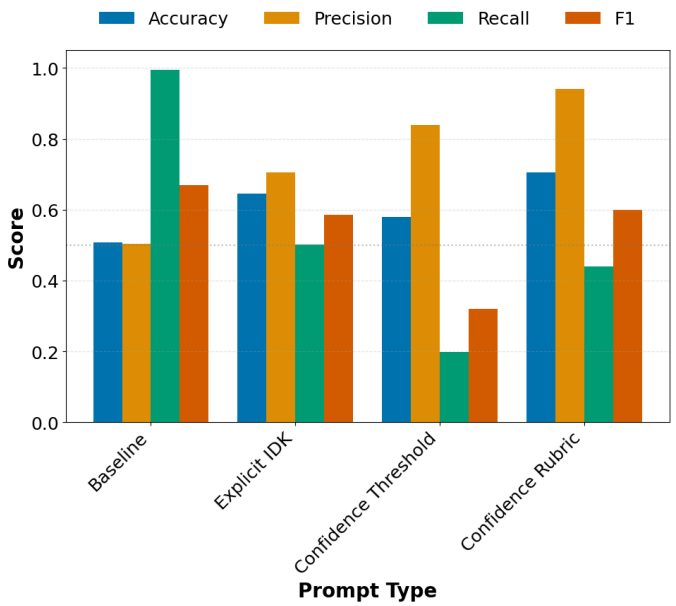

Figure 9: Comparison of binary classification metrics across prompting strategies with answerable and borderline queries on Qwen3 8B. Confidence Rubric suffers the least drop in precision and offers the best precision-recall tradeoff.

Contrary to expectations, as the context length increased, the Qwen3 8B became less hallucination-prone (100% precision at 95% context length) as seen in Figure 1. However, this was because the model became extremely conservative at higher context lengths, answering fewer answerable questions (10% recall at 95% precision). Further analysis of the data revealed 'Lost in the Middle' phenomenon (Liu et al., 2024), where the model is unable to find the answer when it's located in the midst of other text (Figure 2). Comparatively, when the answer is located at the top, the drop in recall is less substantial (Figure 2).

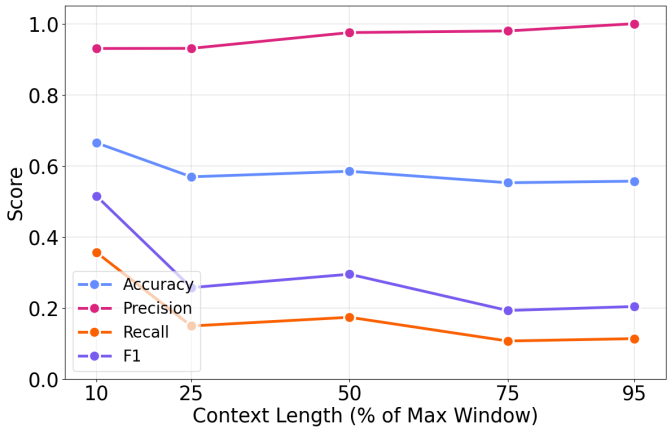

Figure 10: Comparison of binary classification metrics across increasing context length represented as a percentage of the maximum context window on Qwen3 8B. Hallucination rate (Precision) drops with fewer responses (Recall) as the utilized context length increases.

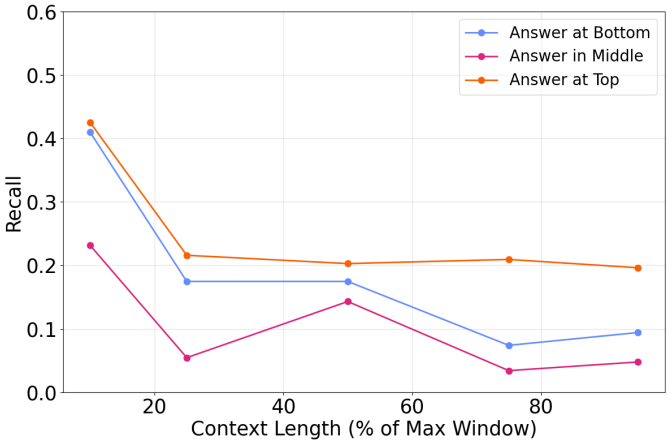

Figure 11: Comparison of Recall (Response for answerable questions) across different locations of answer with increasing context length represented as a percentage of the maximum context window on Qwen3 8B. 'Lost in the Middle' phenomenon observed where the model can't locate answers when they are located in the middle of a large amount of text.

4. Discussion

The substantial improvement in quantitative metrics achieved through Docling-based data processing suggests that not only was text extraction more accurate, but document structural layout was also preserved. This structural preservation enabled optimized chunking methods to group semantically related segments more consistently, while the embedding model's larger context window prevented fragmentation of these segments. BM25's exceptional performance suggests its application without semantic search would enable fast and efficient retrievals. However, while keyword-based retrieval has historically been the dominant approach, the strong metrics here are likely inflated by limitations in the synthetic evaluation data: generated queries tend to share identical keywords with source documents, artificially favoring BM25. In deployment, user queries are unlikely to exhibit such lexical overlap, making the hybrid retrieval with reranking more robust and suitable for real-world use.

The Explicit IDK approach performed well when questions and context were clearly distinguishable, achieving better precision-recall tradeoffs. However, performance degraded substantially with ambiguous queries. The Confidence Rubric, while making the model more conservative overall, demonstrated superior handling of ambiguous queries. This improved robustness likely stems from explicit rubric instructions that delineate which question-context combinations warrant responses. The Baseline prompt, lacking specific instructions, operated according to its fundamental objective of next-token prediction, resulting in increasingly liberal response generation. Conversely, the Confidence Threshold approach overly constrained model responses, as the penalty structure for incorrect answers induced excessive conservatism. The impact of context length on hallucination revealed an interesting finding: rather than increased hallucination, performance degradation primarily stems from the "Lost in the Middle" phenomenon, where models struggle to retrieve information from central portions of long contexts, resulting in fewer responses rather than hallucinated ones. To maintain performance, either conversations should be limited to ~10% of the context window, or aggressive context management (e.g., summarization) should be employed. Additionally, critical information should be front-loaded to improve information retrieval.

In conclusion, although the primary deliverables were met (Table A.1), implementation of the architecture, application layer, and deployment remains as future work. Additionally, user testing is critical to establish accurate evaluation baselines as the reported metrics, while derived from provided documents, rely on synthetically generated evaluation data that may not reflect real-world usage patterns, likely resulting in performance drift at deployment.

5. Statement of Contributions

Atyab Hakeem formulated the solution architecture, conducted advanced data processing, and designed the evaluation methodology. He expanded initial surveys to comprehensively benchmark embedding models, chunking strategies, retrieval, reranking, generation models, and hallucination mitigation techniques. Additionally, he designed and executed experiments to validate hallucination mitigation strategies and developed the corresponding synthetic dataset generation pipeline for evaluation.

Kishan Sathish Babu contributed to the agent architecture design, and the experimentation with the data ingestion pipelines and choosing the embedding model for the vector store.

Naga Kushal worked on experimenting with the re-rankers to figure out which approaches performed better, ran the tests, and generated the final results.

References

1. Lewis, P., et al. (2020). *Retrieval-augmented generation for knowledge-intensive NLP tasks.* NeurIPS.
2. Thakur, N., et al. (2021). *BEIR: A heterogenous benchmark for zero-shot evaluation of information retrieval models.*
3. Gao, L., et al. (2022). *Rethink Training of BERT Rerankers in Multi-Stage Retrieval Pipeline.* ACL.
4. RAGAS (2023). *Faithfulness evaluation for RAG systems.* https://github.com/explodinggradients/ragas
5. Singer-Vine, J. (2024). pdfplumber (Version 0.11.4) [Computer software]. https://github.com/jsvine/pdfplumber
6. Nassar, A., Marafioti, A., Omenetti, M., Lysak, M., Livathinos, N., Auer, C., Morin, L., Teixeira de Lima, R., Kim, Y., Gurbuz, A. S., Dolfi, M., Farré, M., & Staar, P. W. J. (2025). SmolDocling: An ultra-compact vision-language model for end-to-end multi-modal document conversion. arXiv:2503.11576. https://doi.org/10.48550/arXiv.2503.11576
7. Nassar, A., Marafioti, A., Omenetti, M., Lysak, M., Livathinos, N., Auer, C., Morin, L., Teixeira de Lima, R., Kim, Y., Gurbuz, A. S., Dolfi, M., Farré, M., & Staar, P. W. J. (2025). SmolDocling: An ultra-compact vision-language model for end-to-end multi-modal document conversion. arXiv:2503.11576. https://doi.org/10.48550/arXiv.2503.11576
8. RapidOCR. (2021). RapidAI. https://github.com/RapidAI/RapidOCR
9. Wang, L. L., Lo, K., Chandrasekhar, Y., et al. (2020). CORD-19: The Covid-19 Open Research Dataset. *arXiv preprint arXiv:2004.10706*.
10. Schechter Vera, H., Dua, S., et al. (2025). EmbeddingGemma: Powerful and Lightweight Text Representations [arXiv:2509.20354]. Google DeepMind. https://arxiv.org/abs/2509.20354
11. Enevoldsen, K., Chung, I., Kerboua, I., et al. (2025). MMTEB: Massive Multilingual Text Embedding Benchmark [arXiv:2502.13595]. https://doi.org/10.48550/arXiv.2502.13595
12. Robertson, S., & Zaragoza, H. (2009). The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends® in Information Retrieval*, 3(4), 333-389. https://doi.org/10.1561/1500000019
13. Reimers, N., & Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. EMNLP 2019.
14. Günther, M., Ong, J., Mohr, I., Abdessalem, A., Abel, T., Akram, M. K., ... & Xiao, H. (2024). Jina Embeddings 2: 8192-Token General-Purpose Text Embeddings for Long Documents. arXiv preprint arXiv:2310.19923.
15. Zhang, X., Zhang, Y., Long, D., Xie, W., Dai, Z., Tang, J., ... & Zhang, M. (2024). mGTE: Generalized Long-Context Text Representation and Reranking Models for Multilingual Text Retrieval. In Proceedings of EMNLP 2024: Industry Track, 1393–1412.
16. Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of EMNLP 2019.
17. Schechter Vera, H., Dua, S., Zhang, B., Salz, D., Mullins, R., et al. (2025). EmbeddingGemma: Powerful and Lightweight Text Representations. arXiv preprint arXiv:2509.20354.
18. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W. (2020). Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781. https://aclanthology.org/2020.emnlp-main.550/
19. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., ... & Qiu, Z. (2025). Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*.
20. Grattafiori, A., Dubey, A., Jauhri, A., ... & Zettlemoyer, L. (2024). The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*.
21. Bai, S., Cai, Y., Chen, R., ... & Zhou, J. (2025). Qwen3-VL Technical Report. *arXiv preprint arXiv:2511.21631*.
22. Beeching, E., Fourrier, C., Habib, N., Han, S., Lambert, N., Rajani, N., Sanseviero, O., Tunstall, L., & Wolf, T. (2023). *Open LLM Leaderboard*. Hugging Face. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard
23. Kalai, A. T., Nachum, O., Vempala, S. S., & Zhang, E. (2025). Why Language Models Hallucinate. arXiv preprint arXiv:2509.04664.
24. Chroma Research. (2025). Context Rot: How Increasing Input Tokens Impacts LLM Performance. Retrieved from https://research.trychroma.com/context-rot
25. Qwen Team. (2025). Qwen3 Technical Report [arXiv:2505.09388]. https://doi.org/10.48550/arXiv.2505.09388
26. Lo, K., Wang, L. L., Neumann, M., Kinney, R., & Weld, D. S. (2020). S2ORC: The Semantic Scholar Open Research Corpus. *Proceedings of ACL 2020*.
27. Roberts, K., et al. (2020). TREC-COVID: Rationale and Structure of an Information Retrieval Shared Task for COVID-19. *Journal of the American Medical Informatics Association*, 27(9), 1431-1436.
28. Voorhees, E., et al. (2020). TREC-COVID: Constructing a pandemic information retrieval test collection. *SIGIR Forum*, 54(1).
29. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). ReAct: Synergizing Reasoning and Acting in Language Models. In Proceedings of the International Conference on Learning Representations (ICLR). https://arxiv.org/abs/2210.03629
30. Meta AI. (2024). Llama 3.3: Multilingual large language model (Version 70B) [Computer software]. https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_3
31. NVIDIA. (2024). Llama 3.3 70B Instruct FP8 [Quantized model]. Hugging Face. https://huggingface.co/nvidia/Llama-3.3-70B-Instruct-FP8
32. Smith, B., & Troynikov, A. (2024). Evaluating chunking strategies for retrieval (Technical Report). Chroma. https://research.trychroma.com/evaluating-chunking
33. Beltagy, I., Lo, K., & Cohan, A. (2019). SciBERT: A pretrained language model for scientific text. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 3615-3620). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1371
34. Cohan, A., Feldman, S., Beltagy, I., Downey, D., & Weld, D. S. (2020). SPECTER: Document-level representation learning using citation-informed transformers. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (pp. 2270-2282). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.207
35. Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., & Poon, H. (2020). Domain-specific language model pretraining for biomedical natural language processing. arXiv preprint arXiv:2007.15779.
36. Ostendorff, M., Rethmeier, N., Augenstein, I., Gipp, B., & Rehm, G. (2022). Neighborhood contrastive learning for scientific document representations with citation embeddings. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (pp. 11670-11688). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.emnlp-main.802

37. Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers. arXiv preprint arXiv:2002.10957
38. Song, K., Tan, X., Qin, T., Lu, J., & Liu, T. Y. (2020). MPNet: Masked and Permuted Pre-training for Language Understanding. In Proceedings of NeurIPS 2020
39. Subburaj, A., Dua, K., Shah, V., Baumann, P., Poon, H., & Xu, F. (2025). Finding the best chunking strategy for accurate AI responses. NVIDIA Technical Blog. https://developer.nvidia.com/blog/finding-the-best-chunking-strategy-for-accurate-ai-responses/
40. Tang, L., Laban, P., & Durrett, G. (2024). MiniCheck: Efficient fact-checking of LLMs on grounding documents. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (pp. 8818-8847). Association for Computational Linguistics.
41. Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., & Liang, P. (2024). Lost in the Middle: How Language Models Use Long Contexts. Transactions of the Association for Computational Linguistics, 12, 157–173. https://doi.org/10.1162/tacl_a_00638
42. Bao, F., Li, M., Luo, R., & Mendelevitch, O. (2024). HHEM-2.1-Open: Hughes Hallucination Evaluation Model. Hugging Face. https://doi.org/10.57967/hf/3240

Appendix:

Strategy	**Type**	**Chunk Size**	**Chunk Overlap**
S1	Recursive	256	0
S2	Recursive	512	0
S3	Recursive	256	25
S4	Recursive	512	50
S5	Semantic	**Breakpoint** : Percentile	**Threshold**: 90

Table A.1: Chunking strategy definition (Phase 1)

Q) What is matrix suppression? Is it related to ion competition/suppression?	**Q) What is matrix suppression? Is it related to ion competition/suppression?gen 1 alpha 2 (COL1A2) a good marker of cartilage?**	**Q) If I have a highly skewed distribution of subject-level variances and I want to use LIMMA for hypothesis testing. Does this violate any assumptions of LIMMA?**
Answer: Matrix suppression refers to interference caused by competing or suppressed ions in a sample, leading to inaccuracies and noise. It is related as it competes with specific molecules for charge during detection.	Answer: collagen 1 alpha-1 (COL1A2) is mostly detected in subchondral bone for type I and upper cartilage for type II. Post-translational modifications suggest regulatory elements beyond the protein, implying potential link to structural issues but not definitive proof as a cartilage marker.	Answer: Yes, using LIMMA with a highly skewed distribution of variances may not strictly adhere to the normality assumption required by the method, potentially leading to violations of these assumptions or affecting its performance.

Table A.2: Sample Outputs Produced by the Model for Lab Member Queries.

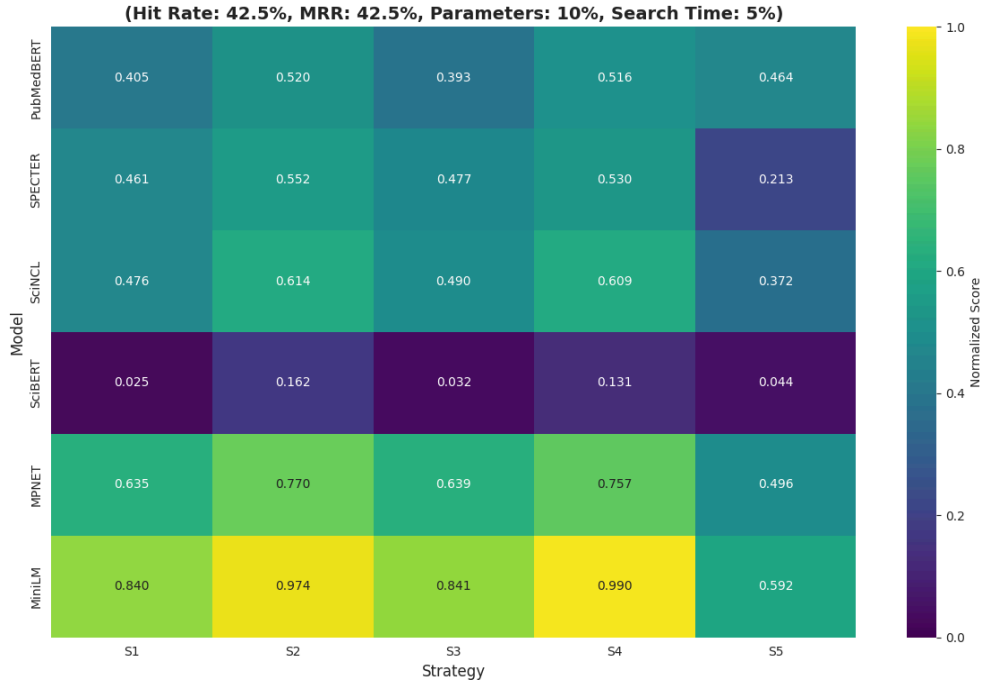

Figure A.1: Heat map of combined scores across embedding models and chunking strategies with parsed text data. MiniLM with strategy 4 provides the best scores

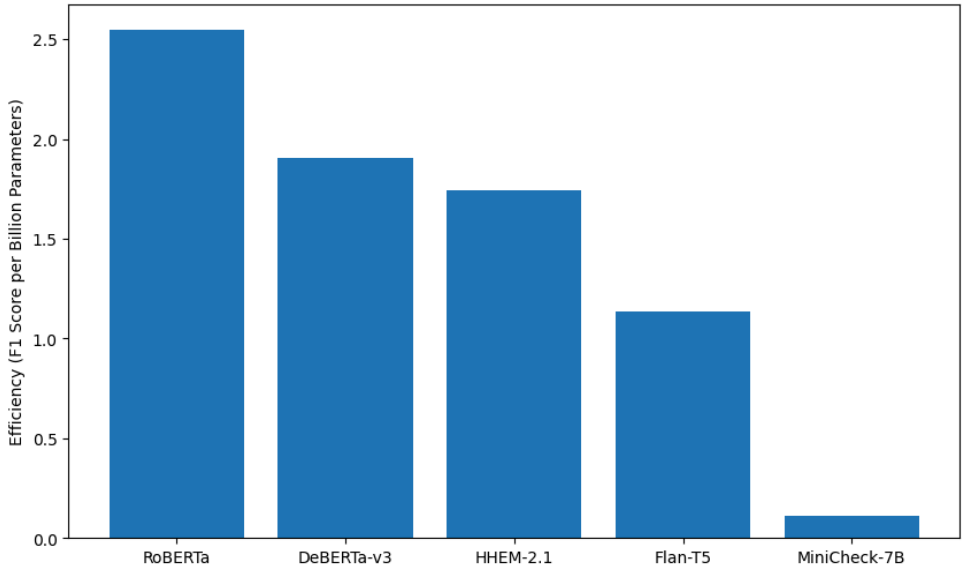

Figure A.2: F1 Score per billion parameters comparison across hallucination detection models. Bespoke RoBERTa had the highest efficiency.

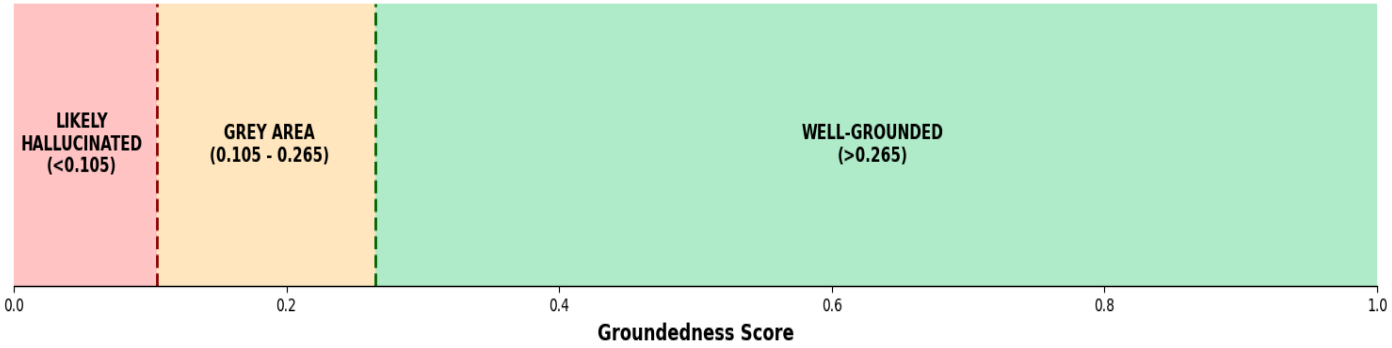

Figure A.3: Hallucination detection and reporting decision framework

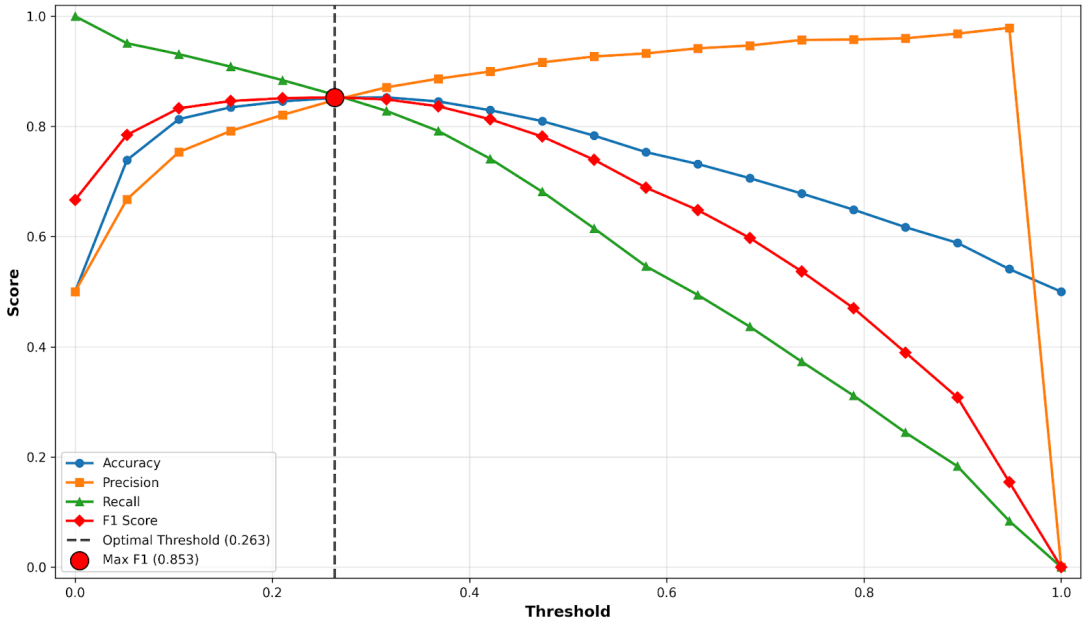

Figure A.4: Confidence threshold gridsearch across accuracy, precision, recall, and F1 score on RoBERTa model. Optimal F1 score achieved at 0.85 confidence threshold.

Objective	Component	Target	Status	Result
Queryable Repository	Parsing, Chunking, Embedding, Retrieval	Hit Rate@10 ≥75% MRR@10 ≥65%	✓	Hit Rate@5 = 85.1% MRR@5 = 86.4%
	Chat Model	Faithfulness ≥85% Relevancy ≥80%	✓	Faithfulness = 88.6% Relevancy = 80.04%
Private	GPU Memory	≤25GB VRAM	✓	~18GB VRAM
	Latency	Simply Query: <10s Complex Query: <60s	⚠	-
	External API	None	✓	Fully private
	Deployment	Integrate with Slack	⚠	-
Groundedness	Hallucination Detection	F1 ≥80%	✓	F1 = 85.3%
	Hallucination Mitigation	Precision ≥85%	✓	Precision = 93%

Table A.3: Final project scorecard