Enhancing Financial Document Retrieval

Hoyoung Lee
Hankuk University of Foreign Studies
Republic of Korea
greenday0021@hufs.ac.kr

Abstract

Retrieval-Augmented Generation (RAG) has become an essential approach for integrating relevant information into large language models, enhancing their ability to address domain-specific questions in fields like finance. This study focuses on the critical task of accurately retrieving information from massive financial documents in response to targeted questions. Financial documents often contain semi-structured and numerical data, requiring retrieval methods that can effectively handle these characteristics for precise information extraction. This empirical study evaluates various retrieval techniques, exploring their ability to address the unique challenges of financial data.

CCS Concepts

• **Information systems** → **Information retrieval**.

Keywords

Financial Document, Retrieval Augmented Generation (RAG)

1 INTRODUCTION

As large language models (LLMs) evolve, their ability to handle extensive context has grown, fueling interest in in-context learning (ICL). For techniques like Retrieval-Augmented Generation (RAG) to effectively support this approach, accurately identifying and presenting relevant information is crucial [10].

In the financial domain, the demand for precise information is especially high, making the retrieval of relevant data essential. Approaches for achieving this include sparse methods like BM25, which rely on term matching, and dense retrieval techniques that leverage embedding models to capture semantic similarity. Each method has strengths and limitations in handling the structured and often numerical nature of financial data, necessitating careful selection and optimization to ensure that only the most relevant information is provided within the context [9].

Embedding models transform text into vector representations, with various models now available and the MTEB benchmark commonly used to compare their performance [7]. However, even state-of-the-art models often experience significant performance drops in specialized domains, such as finance [11]. Therefore, selecting an embedding model that performs well in a domain-specific context is essential for maintaining retrieval accuracy.

Given resource constraints, the selection of an embedding model is guided by performance using the Finance RAG Challenge evaluation dataset [2]. Models showing strong results on this dataset were chosen to represent LLM-based, domain-specific, and proprietary options, allowing for a comprehensive comparison in the experiments.

No single model performed best across all tasks, with each model instead showing strengths in specific areas. Even domain-specific models exhibited substantial variability in performance depending on the task. Moreover, hybrid search and table converting methods proved effective in boosting retrieval performance.

2 METHODOLOGY

The retrieval process starts by identifying relevant documents, followed by a hybrid search that combines sparse and dense models to enhance relevance. An optimized Golden Set is then created and reranked for maximum performance.

2.1 Retrieval Methods

The retrieval process employs both sparse and dense models to capture a broad range of relevant information. For the sparse retrieval, the BM25 model is utilized, known for its effective term-matching capabilities. The dense retrieval approach leverages multiple embedding models to enhance semantic understanding. Specifically, the LLM-based model [1] is used [6, 13], while the domain-specific model [2] provides optimized performance for financial contexts [5]. Additionally, the proprietary model [3] serves as a commercially available option in the retrieval process, contributing to a comprehensive comparison across model types.

2.2 Hybrid Search

Hybrid search combines lexical and dense retrieval methods to leverage the strengths of both approaches. This setup explores the impact of varying α values, which adjust the weighting between sparse and dense retrieval components. The relevance score S_h, is calculated as:

$$S_h = \alpha \cdot S_s + (1 - \alpha) \cdot S_d \tag{1}$$

where S represents the normalized evaluation score after embedding by each model, with S_s as the sparse retrieval score and S_d as the dense retrieval score. Although [1] indicates that TMM is the most effective normalization method, it did not yield significant effects in terms of adjusting weights in this setup. Therefore, min-max normalization was applied to ensure consistent scaling. The weighting parameter α was adjusted in increments of 0.01, ranging from 0 to 1, for each task. The optimal α value was selected based on the highest evaluation score achieved across tasks.

2.3 Reranking

The Golden Set, created through optimal retrieval, undergoes a reranking process to further refine relevance. Reranking reorders the retrieved documents based on their contextual relevance to the

[1] dunzhang/stella-en_1.5B_v5
[2] rbhatia46/financial-rag-matryoshka
[3] Voyage-3, Voyage-finance-2

Method	NDCG@10
Voyage-3 (original)	0.13546
Pseudo Document + Query	0.1096
Query Decomposition	0.11795
Table Converting	0.14623

Table 1: Performance on the MultiHiertt

query, ensuring that the most pertinent information is prioritized. Various reranking models are employed, including an LM-based reranker[4], an LLM-based reranker[5], and a proprietary reranker[6]. Each reranker model contributes unique strengths to the reranking process, enhancing the quality of the final retrieved results.

3 EXPERIMENTS

3.1 Retriever Enhancement Strategy

Among the various tasks in the Finance RAG Challenge, this section focuses on the challenging MultiHiertt task, a multi-hop QA task involving tabular data. Effective retrieval strategies are explored to address its complex, structured, and multi-layered information requirements. The following methods were applied:

Pseudo Document Creation. The HyDE generates pseudo documents [3, 12], but these often rely on limited context, which may affect performance. To produce more relevant generated content, GraphRAG is used [8]. However, due to the high computational cost of GraphRAG, a lightweight version called LightRAG[4] is employed to generate text responses that serve as pseudo documents, enriching the retrieval process with contextually relevant content.

Query Decomposition. Given the multi-hop nature of the task, complex queries were broken down into multiple sub-queries. This decomposition allows each sub-query to address specific aspects of the original query.

Table Converting. Since the task involves tabular data, all information from tables was transformed into natural language using an LLM. This method reformulates tabular data in a way that enhances understanding and retrieval, making it easier for models to process and extract relevant information.

As shown in Table 1, the only method that significantly improves performance over the original approach is Table Converting, indicating its effectiveness in handling tabular data within the MultiHiertt task.

3.2 Performance Across Challenge Tasks

Table 2 demonstrates that no single model consistently outperforms across all tasks. However, hybrid search generally leads to performance improvements in most cases. While Table Converting contributes to additional gains, its impact is moderate rather than substantial. Reranking boosts performance in many cases, significantly increasing scores. However, it does not always lead to improvements. For tasks like FinQABench and FinanceBench, where high retrieval scores were already achieved, reranking actually resulted in decreased performance, as shown in Table 3. Interestingly, reranking on the Golden Set with Table Converting applied yields lower performance than reranking on the set without Table Converting. This suggests that the additional processing involved in Table Converting may not always align with the reranking models' criteria for relevance.

[4]Alibaba-NLP/gte-multilingual-reranker-base
[5]BAAI/bge-reranker-v2.5-gemma2-lightweight
[6]Voyage-reranker2

4 RESULT

Table 4 presents the public scores for each task, compiled into an Answer Set, representing the highest performance achieved after reranking and evaluated on the entire test dataset rather than just the validation set. Notably, the Table Converting method did not produce a meaningful improvement and, in fact, resulted in a slightly lower score. This suggests that both the embedding model and reranking model already possess adequate capabilities for handling tabular data, making the conversion to natural language unnecessary. Additionally, optimizing the alpha value in Hybrid Search using a subset of the evaluation data led to slight overfitting, causing a minor decrease in the overall public score.

Method	Public Score
Converting	0.59522
w/o Converting	0.59701

Table 4: Public Score Comparison for Answer Set with and without Table Converting

References

[1] Sebastian Bruch, Siyu Gai, and Amir Ingber. An analysis of fusion functions for hybrid retrieval. *ACM Transactions on Information Systems*, 42(1):1–35, 2023.

[2] Chanyeol Choi, Jy-Yong Sohn, Yongjae Lee, Subeen Pang, Jaeseon Ha, Hoyeon Ryoo, Yongjin Kim, Hojun Choi, and Jihoon Kwon. Acm-icaif '24 financerag challenge. https://kaggle.com/competitions/icaif-24-finance-rag-challenge, 2024. Kaggle.

[3] Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. *arXiv preprint arXiv:2212.10496*, 2022.

[4] Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. Lightrag: Simple and fast retrieval-augmented generation. *arXiv preprint arXiv:2410.05779*, 2024.

[5] Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. Matryoshka representation learning, 2024.

[6] Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023.

[7] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022.

[8] Bhaskarjit Sarmah, Benika Hall, Rohan Rao, Sunil Patel, Stefano Pasquali, and Dhagash Mehta. Hybridrag: Integrating knowledge graphs and vector retrieval augmented generation for efficient information extraction. *arXiv preprint arXiv:2408.04948*, 2024.

[9] Kuldeep Singh, Simerjot Kaur, and Charese Smiley. Finqapt: Empowering financial decisions with end-to-end llm-driven question answering pipeline. *arXiv preprint arXiv:2410.13959*, 2024.

[10] Pragya Srivastava, Manuj Malik, Vivek Gupta, Tanuja Ganu, and Dan Roth. Evaluating llms' mathematical reasoning in financial document question answering. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 3853–3878, 2024.

[11] Yixuan Tang and Yi Yang. Do we need domain-specific embedding models? an empirical investigation. *arXiv preprint arXiv:2409.18511*, 2024.

[12] Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian, et al. Searching for best practices in retrieval-augmented generation. *arXiv preprint arXiv:2407.01219*, 2024.

[13] Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, et al. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. *arXiv preprint arXiv:2407.19669*, 2024.

Model	FinDER	FinQABench	FinanceBench	TATQA	FinQA	ConvFinQA	MultiHiertt
BM25	0.120	0.824	0.181	0.417	0.458	0.422	0.107
Stella_en_1.5B_v5	0.583	0.913	0.894	0.354	0.299	0.301	0.098
Hybrid	**0.589**	0.935	0.894	0.488	0.467	0.445	0.126
Financial-rag-matryoshka	0.421	0.947	0.904	0.318	0.288	0.428	0.098
Hybrid	0.435	**0.950**	**0.912**	0.452	0.478	0.551	0.136
Voyage-2-finance	0.317	0.881	0.848	0.497	0.628	0.620	0.140
Hybrid	0.334	0.890	0.856	0.550	0.664	0.662	0.164
Table Converting + Hybrid	-	-	-	0.489	0.671	0.685	**0.172**
Voyage-3	0.342	0.874	0.796	0.522	0.622	0.649	0.135
Hybrid	0.356	0.910	0.800	0.570	0.670	0.705	0.161
Table Converting + Hybrid	-	-	-	**0.574**	**0.675**	**0.705**	0.168

Table 2: NDCG@10 for All Tasks Using the Evaluation Dataset. Note that Table Converting was applied only to tabular retrieval tasks.

Model	FinDER	FinQABench	FinanceBench	TATQA	FinQA	ConvFinQA	MultiHiertt
Reranker-v2	0.492	0.862	0.769	0.704	**0.769**	**0.786**	0.241
Reranker-v2-Convert	0.492	**0.862**	0.769	**0.706**	0.769	0.780	**0.243**
GTE-reranker	0.512	0.812	0.881	0.564	0.682	0.717	0.202
bge-reranker-v2.5-gemma2	**0.658**	0.838	**0.926**	0.647	0.531	0.575	0.186

Table 3: NDCG@10 for Reranking Results on the Golden Set Composed of Top-Scoring Retrievals