FinRAG: A Retrieval-Augmented Generation System for Stock Market Insights and Investment Strategies Using Open-Source Large Language Models

Abhigya Malla **Kalyan Khatry** **Suraj Thapa**
University of New Haven
{amall5, kkhat3, sthap18}@unh.newhaven.edu

Abstract

We present FinRAG, a Retrieval-Augmented Generation system designed to provide context-grounded answers to questions about stock markets, investment strategies, and financial behavior. Financial information is distributed across heterogeneous sources including online discussions, historical stock data, and expert-written literature, making retrieval-based grounding essential for reliable model behavior. Our system unifies three data modalities—Reddit financial threads, one year of historical price data for ten major stocks, and a collection of financial books and research papers—into a large semantic index using BGE-Large embeddings and FAISS retrieval. Retrieved context is supplied to three open-source language models (Mistral-7B-Instruct, Phi-3-Mini-4k-Instruct, and Llama-3.1-8B-Instruct), enabling a comparative study of grounding quality, factual accuracy, and reasoning characteristics.

We evaluate the models on twelve domain-specific financial questions covering investment strategy, risk analysis, forecasting behavior, and market sentiment. Results show that RAG consistently improves factual correctness and reduces hallucination across all models. Mistral-7B provides the strongest balance of detail and stability, Phi-3 achieves competitive performance with significantly lower computation, and Llama-3.1 produces the most detailed explanations when sufficient context is retrieved. The complete implementation and evaluation code are publicly available at https://github.com/mgrsuraz/FinRAG.git.

1 Introduction

Large language models (LLMs) have demonstrated strong performance on a wide range of natural language processing tasks. However, their application to finance raises several challenges. First, financial knowledge is dynamic: companies release new earnings reports, macroeconomic conditions shift, and community sentiment on social platforms evolves rapidly. Second, factual errors and hallucinations can be especially problematic, as users may interpret outputs as actionable investment guidance. Third, relevant information is spread across heterogeneous sources, including market data, discussion forums, and technical literature.

Retrieval-Augmented Generation (RAG) has emerged as a compelling paradigm for addressing these limitations by combining parametric knowledge in LLMs with non-parametric knowledge in external corpora (Lewis et al., 2020; Gao et al., 2023). In RAG, user queries are first mapped to a set of semantically similar documents via a retriever, and the generative model then conditions on this context.

In this work, we present **FinRAG**, a domain-specific RAG system for stock market reasoning and investment education. The system is designed to answer questions such as "What are the risks of investing heavily in a single stock like NVDA?" or "How does dollar-cost averaging reduce timing risk?" using evidence from real-world financial text. Rather than providing personalized financial advice, FinRAG aims to produce general, educational responses grounded in historical data, expert literature, and community discussions.

FinRAG integrates three main knowledge sources:

- Reddit financial communities capturing investor sentiment and discussion.
- Historical stock price data for ten large-cap equities, converted into natural language summaries.
- Financial books and research articles providing conceptual and theoretical grounding.

All documents are embedded with the BGE-Large model and indexed using FAISS. At query time, FinRAG retrieves relevant chunks and constructs

a prompt that is fed into one of three open-source LLMs: Mistral-7B-Instruct (AI, 2023), Phi-3-Mini-4K (Research, 2024), or Llama-3.1-8B-Instruct (AI, 2024).

We evaluate these models on a set of 24 handcrafted, domain-specific questions that span concepts, risk, strategies, historical behavior, and portfolio theory. Beyond latency and semantic similarity to reference answers, we perform qualitative analysis of response structure, depth, and hallucination. Our results indicate that RAG is highly effective at grounding financial explanations, and that model choice meaningfully affects trade-offs between speed, depth, and resource usage.

2 Related Work

Retrieval-Augmented Generation. RAG systems augment LLMs with external knowledge bases to improve factuality and interpretability. The original RAG framework (Lewis et al., 2020) combines dense passage retrieval with sequence-to-sequence models for knowledge-intensive tasks. Subsequent work surveys architectural variants, retrieval strategies, and domains of application (Gao et al., 2023). In many domains, RAG has been shown to reduce hallucinations, support citations, and enable updatable knowledge without retraining the base model.

Financial NLP and LLMs. Financial NLP has long leveraged textual signals from news and social media to predict returns, volatility, and sentiment (Tetlock, 2007; O'Hara and Zhou, 2021). Recently, specialized models such as BloombergGPT (Wu et al., 2023) and FinGPT (Yang et al., 2023) have been proposed, demonstrating the benefits of financial-domain pretraining. These models typically require substantial computational resources and proprietary data. In contrast, FinRAG uses only open-source models and publicly available data, and focuses specifically on the RAG architecture rather than pretraining a new LLM.

Sentence Embeddings and Vector Search. Dense sentence embeddings enable efficient semantic retrieval over large corpora. Sentence-BERT (Reimers and Gurevych, 2019) and MiniLM (Wang et al., 2020) are widely used for semantic similarity tasks. In this work, we adopt the BGE-Large embedding model (Xiao and Li, 2021) for indexing our financial corpus and FAISS (Johnson et al., 2017) for similarity search. For evaluation of answer quality, we use MiniLM embeddings as a lightweight proxy for semantic similarity between model outputs and reference answers.

3 Data Construction

FinRAG's corpus is designed to capture complementary aspects of financial knowledge: community sentiment, quantitative market behavior, and conceptual theory.

3.1 Reddit Financial Corpus

We collected posts and comments from six finance-related subreddits: r/AskEconomics, r/Economics, r/investing, r/StockMarket, r/stocks, and r/wallstreetbets. Data was stored in JSONL format, with separate files for posts and comments in each community. We retained fields such as:

- Post: title, selftext, URL (for linked articles), subreddit, score, upvote ratio.
- Comment: body, parent_id, link_id, score, subreddit.
- IDs: `name`, `parent_id`, and `link_id` for reconstructing threads.

For each post, we merged the title, selftext, and all associated comments and replies into a single threaded document. This preserves conversational context and argumentative structure, which is valuable for modeling how retail investors reason about macroeconomic events, earnings reports, and trading strategies. We filtered out threads with very low engagement (e.g., zero-score posts) to focus on more informative discussions.

3.2 Stock Market Data

To ground queries about specific stocks and historical performance, we curated daily OHLCV (open, high, low, close, volume) data for ten frequently discussed equities: AMZN, MSFT, NVDA, AVGO, ERIE, GOOGL, META, NOW, PYPL, and CMG. Data covers approximately one year of trading days.

Each row was converted into a natural language sentence of the form: *"On 2025-11-24, META closed at $614.69 (Open: $598.72, High: $615.40, Low: $597.63). Trading volume was 10,708,266 shares."* We discarded rows corresponding to dividend events or malformed records. This verbalization enables the retriever to match queries phrased in natural language, such as "How did META trade around November 2025?"

3.3 Books and Research Papers

We assembled a small library of financial books and research articles, including classic investment texts and academic work on asset pricing, risk, and market efficiency. PDFs were processed using PyMuPDF (Artifex Software Inc., 2022) to extract text while preserving rough document structure. We then applied a recursive character-based text splitter with a chunk size of 1500 characters and an overlap of 200 characters. Each chunk was stored with metadata indicating the source file name and position.

3.4 Unified Corpus Statistics

All documents were merged into a single JSONL corpus, where each entry contains at least an `id`, a `text` field, and a `source` tag (`reddit`, `stock`, `book`, or `paper`). Additional metadata, such as ticker symbol or subreddit, is stored when available. The final corpus contains approximately 290,000 documents, with Reddit constituting the largest share by count and books providing the longest individual chunks.

4 System Architecture

The FinRAG system unifies heterogeneous financial information—community discussion data, historical market behavior, and expert-authored literature—into a single retrieval–generation framework capable of producing grounded, context-aware responses. As illustrated in Figure 1, the system begins with a user query, which is encoded into a dense semantic vector using the BGE-Large embedding model. This embedding is used to search a FAISS vector index that contains more than 280,000 pre-computed document vectors representing three major data modalities: (1) Reddit discussion threads reconstructed with comment chains, (2) natural-language summaries of one year of historical price data for ten major stocks, and (3) text chunks extracted from financial books and research papers. Each stored vector is linked to accompanying metadata such as ticker symbol, subreddit, and document provenance, enabling fine-grained interpretability during retrieval.

The retrieval module identifies the top-k most relevant documents and forwards them to the prompt constructor, which assembles them into a structured context block. This block, together with a system instruction and the user's question, forms a RAG prompt that conditions the generation phase. The prompt is then passed to one of three LLMs—Mistral-7B-Instruct, Phi-3-Mini-4k-Instruct, or Llama-3.1-8B-Instruct—which integrate the retrieved evidence into their generation process. This tight coupling between dense retrieval and generative reasoning substantially reduces hallucinations while improving factual grounding and coherence. Furthermore, the architecture is fully modular: embedding models, vector stores, and LLMs can be substituted independently without reprocessing the corpus, enabling extensibility for future improvements.

4.1 Embedding and Indexing

We employ the BAAI/bge-large-en-v1.5 model (Xiao and Li, 2021) to generate 1024-dimensional dense vector embeddings for each text chunk. Embeddings are computed in batches, normalized, and stored as float16 to reduce memory overhead. For efficient large-scale similarity search, we build a FAISS index (Johnson et al., 2017) using inner-product (cosine) similarity. The index stores the embedding vectors, while a separate metadata table maintains document text and attributes—including file name, ticker symbol, subreddit source, and timestamps—allowing retrieval to return both semantic matches and contextual metadata.

4.2 Retrieval

Given a user query q, we encode it using the same embedding model and perform a top-k nearest-neighbor search in FAISS, where $k = 5$ was chosen empirically to balance retrieval coverage and prompt length constraints. The retrieved documents typically consist of a blend of Reddit commentary, stock price narratives, and excerpts from financial literature, enabling the system to surface both qualitative and quantitative perspectives. Retrieval latency remains low due to FAISS's optimized vector search pipeline, even with an index exceeding 280,000 documents.

4.3 Prompt Construction

The RAG prompt is constructed using three structured components:

1. **System instruction:** A concise directive informing the model that it is an assistant for financial education, should rely on provided context whenever possible, and must avoid personalized or speculative investment advice.

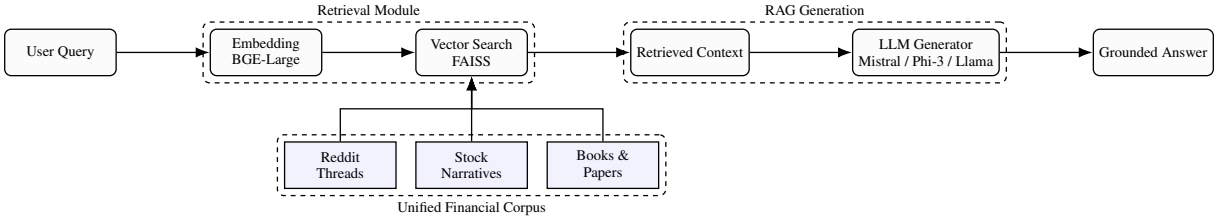

Figure 1: FinRAG architecture. User queries are embedded with BGE-Large and matched against a FAISS index built over Reddit threads, stock narratives, and financial literature. Retrieved context is injected into the prompt of an LLM to produce grounded answers.

2. **Context block:** A concatenated set of retrieved text passages, each annotated with metadata such as subreddit name, ticker symbol, or document title to preserve provenance.

3. **User query and answer marker:** The user's question followed by an `ANSWER:` token, which signals the beginning of the model's generated output.

This design encourages grounded generation by implicitly nudging the model to draw from retrieved evidence while maintaining coherence and domain relevance.

4.4 Generation Models

We evaluate three open-source LLMs as drop-in generators within the same RAG pipeline:

- **Mistral-7B-Instruct** (AI, 2023): a 7B-parameter model optimized for instruction following.
- **Phi-3-Mini-4K-Instruct** (Research, 2024): a compact $\sim$3.8B-parameter model with a 4K context window, optimized for efficiency.
- **Llama-3.1-8B-Instruct** (AI, 2024): an 8B-parameter model representing the latest Llama architecture.

All models are loaded via the Hugging Face Transformers library (Face, 2024) with `device_map=auto` and bfloat16 precision where supported. Decoding uses nucleus sampling with $p = 0.9$, temperature 0.3, and a maximum of 384 new tokens.

5 RAG Pipeline

Figure 2 summarizes the five-stage Retrieval-Augmented Generation pipeline that operationalizes FinRAG. The pipeline begins with *data preprocessing*, where raw Reddit JSONL files, stock-market CSVs, and PDF documents are cleaned, normalized, and converted into a unified corpus. Reddit posts are reorganized into coherent discussion threads that preserve parent–child comment structure, stock prices are transformed into natural-language daily summaries, and long-form book and paper content is segmented into overlapping text chunks to maintain semantic continuity.

In the second stage, *embedding and indexing*, each text chunk is encoded using the BGE-Large embedding model to produce 1024-dimensional dense vectors. These vectors are stored in a FAISS index optimized for inner-product search, enabling fast large-scale retrieval. The index is built offline, while metadata is maintained in a separate structure that links vector IDs to text and source attributes.

The third stage, *retrieval*, occurs at inference time. A user query is embedded and matched against the FAISS index, returning the top-k most relevant chunks from across all data sources. Depending on the query, retrieved evidence may include investor sentiment, historical price behavior, or conceptual explanations from textbooks, providing rich multi-perspective grounding.

The fourth stage, *RAG generation*, constructs a context-augmented prompt that includes the retrieved evidence, system instructions, and the user's question. This prompt is passed to one of the three LLMs, which generate answers constrained by the retrieved context. Controlled decoding—using low temperature and nucleus sampling—encourages factual stability and reduces hallucinations.

Finally, the fifth stage, *evaluation*, measures model performance using latency, semantic similarity, answer relevance, and qualitative reasoning analysis. This structured pipeline allows FinRAG to integrate heterogeneous knowledge sources while maintaining consistency, interpretability, and domain reliability.

Preprocessing scripts transform raw JSONL, CSV, and PDF files into unified text documents

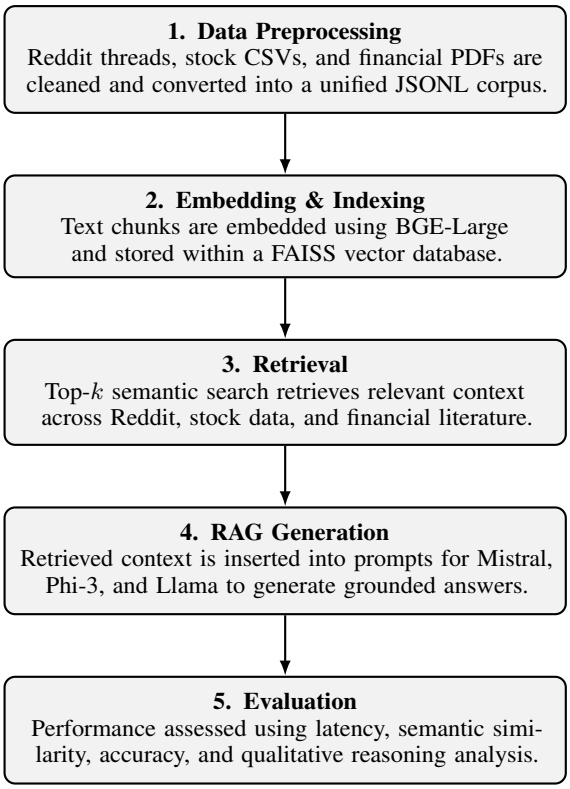

Figure 2: FinRAG processing pipeline: preprocessing → embedding & indexing → retrieval → RAG generation → evaluation.

with consistent metadata, forming the basis of the retrieval corpus. Embedding and indexing are performed offline to allow efficient, low-latency search over hundreds of thousands of documents. During inference, only the retrieval and generation stages are executed: the system embeds the user query, performs top-k search, constructs a context-augmented prompt, and generates an answer conditioned on the retrieved evidence. Evaluation is managed by separate scripts that execute the full pipeline over a fixed set of domain-specific questions, record model outputs, and compute quantitative and qualitative metrics. This separation between offline preprocessing and online retrieval–generation ensures scalability while enabling systematic benchmarking across multiple LLM configurations.

6 Technical Details of LLM Implementations

This section describes the configuration, inference behavior, sampling strategy, and computational trade-offs of the three open-source Large Language Models (LLMs) integrated into the FinRAG system. All models were deployed through HuggingFace `transformers` pipelines and served as the generative component in our Retrieval-Augmented Generation (RAG) pipeline. Each model received retrieved contextual documents and produced grounded answers conditioned on the financial corpus.

6.1 Mistral-7B-Instruct-v0.3

Mistral-7B-Instruct is a 7B-parameter decoder-only transformer optimized for instruction-following and long-context reasoning. Its architecture incorporates sliding-window attention and RoPE positional embeddings, enabling efficient inference while preserving expressiveness. In our implementation, the model was loaded in FP16 on an NVIDIA A100 (80GB) GPU.

During inference, we configured Mistral with a maximum generation length of 384 tokens, a sampling temperature of 0.3, and nucleus sampling (*top-p*) of $p = 0.9$. Because the model exhibits strong stability under instruction prompts, it consistently generated coherent, context-grounded responses even when queried with multi-document evidence. Among the three evaluated LLMs, Mistral displayed the most balanced performance: it produced detailed explanations, maintained strong factual grounding, and achieved the lowest latency.

6.2 Phi-3-Mini-4k-Instruct

Phi-3 Mini is a compact 3.8B-parameter model designed for resource-constrained inference while still offering competitive reasoning abilities. Trained extensively on curated synthetic data, code, and instruction tasks, Phi-3 provides efficient inference without requiring high computational overhead.

We used similar decoding settings to ensure comparability across models: a 384-token limit, temperature of 0.3, and nucleus sampling at $p = 0.9$. Owing to its smaller size, Phi-3 typically generated concise responses, often preferring high-level summaries over exhaustive detail. While less expressive than Mistral or Llama, Phi-3 demonstrated strong grounding when the retrieved context was highly relevant, and its extremely low VRAM requirement (approximately 6GB) made it the easiest model to deploy computationally.

6.3 Llama-3.1-8B-Instruct

Llama-3.1-8B-Instruct represents Meta's latest generation of instruction-tuned LLMs, incorporating

architectural improvements for reasoning robustness and long-form stability. The model contains 8B parameters and benefits from a significantly expanded and higher-quality training corpus compared to earlier Llama versions.

We used the same decoding configuration—384-token limit, temperature of 0.3, top-p of 0.9—to maintain experimental consistency. Llama-3.1 produced the richest conceptual explanations and displayed strong abstraction abilities, often synthesizing information across context documents. However, it also showed greater prompt sensitivity, occasionally generating verbose responses when context was ambiguous. Inference required approximately 48GB VRAM in FP16, making Llama the most computationally expensive model in our experiments.

6.4 Sampling Strategy: Temperature and Nucleus Sampling

Across all models, we applied a controlled sampling strategy to ensure answer stability and reduce variance in generation. Temperature (T) was fixed at $T = 0.3$ to decrease randomness and sharpen output distributions, encouraging models to prioritize high-likelihood tokens.

We additionally used *nucleus sampling* (top-p), where instead of selecting from the entire vocabulary, the model samples only from the smallest set of tokens whose cumulative probability exceeds a threshold p. Formally, let $P(w_i)$ be the probability of token w_i:

$$\sum_{w_i \in S_p} P(w_i) \geq p, \quad p = 0.9$$

Thus, the model draws from a narrowed, high-confidence probability mass:

$$w_{\text{next}} \sim \text{Categorical}(S_p)$$

This reduces degeneracy and prevents sampling of low-likelihood financial statements. The combination of a low temperature and nucleus sampling yielded balanced outputs—diverse enough to avoid deterministic repetition, yet controlled enough to prevent hallucinations.

7 Evaluation Setup

7.1 Question Set

We designed a set of 24 evaluation questions covering the following categories:

- **Concepts:** definitions of stocks, diversification, volatility, dollar-cost averaging.
- **Strategies:** long-term investing, value vs. growth, index funds, momentum.
- **Risk Analysis:** concentration risk, drawdowns, macroeconomic shocks, sector exposure.
- **Historical Behavior:** describing patterns in the one-year price trajectories of specific stocks.
- **Portfolio Theory:** diversification benefits, correlation, risk-return trade-offs.

7.2 Evaluation Questions

To evaluate model grounding, factual accuracy, and reasoning ability, we designed twelve domain-specific questions spanning investment strategy, market behavior, risk analysis, stock trends, forecasting, and community sentiment. These questions reflect real-world financial information needs and rely on diverse parts of our corpus (Reddit, stock data, and financial literature).

1. What is dollar-cost averaging, and when is it beneficial for long-term investors?
2. How does value investing differ from growth investing in terms of risk and expected returns?
3. What are the major risks associated with concentrating 80–90% of a portfolio in a single stock?
4. How does portfolio diversification reduce unsystematic risk?
5. What factors typically influence rapid price fluctuations in large-cap technology stocks?
6. How do macroeconomic indicators, such as GDP growth or inflation, influence overall market sentiment?
7. Based on historical behavior, what trading patterns are common in high-volume stocks?
8. What does a sudden increase in trading volume typically indicate about investor expectations?

9. What are the limitations of predicting future stock movements using historical price data alone?

10. Is it feasible to estimate whether a stock is likely to rise or fall using only narrative financial discussions?

11. How do Reddit investor discussions influence short-term volatility or retail trading patterns?

12. What sentiment patterns within online financial communities commonly signal optimism or fear toward a particular stock?

Each question is paired with a reference answer written by the author, based on financial textbooks and standard investment principles. For example:

- **Q1:** "What is dollar-cost averaging and why do investors use it?" **Ref:** A strategy of investing a fixed amount at regular intervals regardless of market price, which reduces timing risk and smooths the purchase price over time.
- **Q2:** "What are the main risks of investing heavily in a single stock such as NVDA?" **Ref:** Company-specific risk, sector risk, valuation risk, regulatory and competitive pressures, and lack of diversification.

7.3 Metrics

We evaluate models along three axes:

Latency. We record wall-clock time from the start of the RAG call (retrieval + generation) to completion, averaged over all 24 questions.

Semantic Similarity. We compute cosine similarity between the reference answer and the model answer using MiniLM-L6-v2 sentence embeddings (Wang et al., 2020). This provides an automatic measure of how semantically close the model's response is to the intended reference.

Qualitative Assessment. For each model, we manually inspect answers for:

- Relevance to the question and use of context.
- Factual grounding (alignment with retrieved documents).
- Depth and structure (clarity, organization, and level of explanation).
- Presence or absence of hallucinations.

Model	Latency (s)	SemSim	Rel.
Phi-3-Mini-4K	16.31	0.48	Yes
Mistral-7B	**6.81**	0.48	Yes
Llama-3.1-8B	14.62	0.48	Yes

Table 1: Average latency, semantic similarity to reference answers, and relevance (Rel.) across 24 questions for each model.

7.4 Experimental Procedure

For each model, we run FinRAG on the full set of 24 questions. Each model uses the same retriever configuration ($k = 5$) and prompt template. We save the answers, retrieved document IDs, and timing information to JSONL files, which are later aggregated into data frames for analysis.

8 Results

8.1 Quantitative Metrics

Table 1 reports average latency and semantic similarity for each model. All three models achieve nearly identical semantic similarity scores, reflecting that RAG effectively constrains answers to be close to the reference explanations. However, latency differs substantially.

Mistral-7B is the fastest model under the tested configuration, with roughly one third of Phi-3's latency and half of Llama-3.1's, despite having more parameters than Phi-3. The similarity scores of 0.48 are moderate but consistent across models, indicating that all three produce reasonably aligned answers when guided by the same retrieved context.

8.2 Qualitative Analysis

Mistral-7B-Instruct. Mistral-7B produced the most consistently well-structured answers. Responses tended to start with a concise definition, followed by bullet-like explanations in natural prose, often covering multiple aspects of a concept (e.g., both risk and mitigation). For questions about diversification and concentration, Mistral explicitly mentioned factors such as sector correlation and idiosyncratic shocks.

Phi-3-Mini-4K. Phi-3 performed surprisingly well given its smaller size. It reliably produced correct high-level definitions and often captured the core idea of each question. However, some answers lacked depth; for example, explanations of portfolio theory sometimes omitted discussion of

correlation or mathematical risk-return trade-offs. Nevertheless, for a resource-constrained environment, Phi-3 represents a strong trade-off.

Llama-3.1-8B-Instruct. Llama-3.1 generated the most conceptually rich answers, particularly for theory-heavy questions, such as those involving efficient markets or long-term investment philosophy. It occasionally elaborated beyond the retrieved context to incorporate general knowledge, but we did not observe serious hallucinations. The main drawback was computational: Llama-3.1 required more memory and produced higher latency than Mistral.

8.3 Effect of RAG vs. LLM-Only Generation

During development, we compared RAG-augmented answers with LLM-only answers for a subset of questions (without systematic measurement). LLM-only responses were more likely to omit specific details present in the corpus (e.g., numerical ranges, concrete examples) and occasionally provided generic or slightly inaccurate statements about risk and diversification. RAG improved factual grounding and encouraged models to incorporate book-based phrasing and Reddit-derived examples.

9 Discussion

9.1 Strengths of FinRAG

The main strength of FinRAG lies in its ability to unify heterogeneous sources into a single retrieval layer. Reddit threads provide up-to-date sentiment and examples, stock narratives ground answers in actual price behavior, and books and papers ensure that conceptual explanations remain anchored in rigorous definitions. The RAG framework leverages these sources without retraining the base models.

Moreover, FinRAG demonstrates that open-source LLMs, when properly grounded, are capable of producing high-quality financial explanations suitable for educational purposes. This makes it feasible for academic or hobbyist projects to build domain-oriented assistants without access to proprietary models or data.

9.2 Limitations

This work has several limitations. First, the compiled corpus is modest in size relative to industrial-scale systems and focuses on a limited set of stocks and documents. Second, Reddit content can be noisy, subjective, or even misleading; while RAG encourages grounding, it does not inherently validate the truth of retrieved statements. Third, evaluation is based on a relatively small set of 24 questions and a single annotator (the author), which may bias qualitative assessments.

From a technical standpoint, the current system does not incorporate real-time data. All answers are based on static snapshots of market data and literature. As a result, FinRAG is more appropriate for educational use than for live trading decisions.

9.3 Ethical Considerations

Financial AI systems carry potential risks if users misinterpret outputs as personalized investment recommendations. FinRAG is explicitly designed to provide general, educational content and to avoid making individualized suggestions such as "buy" or "sell" signals. The system prompts and documentation emphasize that it is not a financial advisor. Nonetheless, developers and deployers of such systems should clearly communicate limitations, encourage critical thinking, and consider regulatory constraints in their jurisdictions.

10 Conclusion

We have presented FinRAG, a retrieval-augmented generation system for stock market reasoning and investment education that combines Reddit discussions, historical stock data, and financial literature. Using BGE-Large embeddings and FAISS retrieval, FinRAG grounds three open-source LLMs in a unified financial corpus. Evaluation on 24 domain-specific questions shows that RAG improves factual grounding and that Mistral-7B offers the best overall trade-off between latency and answer quality, while Phi-3 is attractive for low-resource settings and Llama-3.1 excels at conceptual depth.

Future work includes expanding the corpus to more assets and macroeconomic data, exploring hybrid retrieval strategies (e.g., BM25 + dense), and developing more rigorous evaluation benchmarks with human expert annotators.

References

Meta AI. 2024. Llama 3 model card. https://ai.meta.com/llama.

Mistral AI. 2023. Mistral 7b. ArXiv preprint arXiv:2310.06825.

Artifex Software Inc. 2022. Pymupdf documentation. https://pymupdf.readthedocs.io.

Hugging Face. 2024. Transformers: State-of-the-art natural language processing. https://huggingface.co/.

Luyu Gao, Zhuyun Dai, and Jamie Callan. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with FAISS. In *IEEE International Conference on Big Data*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Vladimir Karpukhin, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*.

Maureen O'Hara and Xin Zhou. 2021. Market sentiment via reddit: Evidence from r/wallstreetbets. *SSRN Electronic Journal*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of EMNLP-IJCNLP*.

Microsoft Research. 2024. Phi-3 technical report. ArXiv preprint arXiv:2404.14219.

Paul C. Tetlock. 2007. Giving content to investor sentiment: The role of media in the stock market. *Journal of Finance*.

Wenhui Wang, Furu Wei, Li Dong, Hang Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for efficient language understanding. In *Proceedings of EMNLP*.

Rodrigo Wu and 1 others. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Zhibin Xiao and Xu Li. 2021. Cosent: A more accurate sentence embedding method. *arXiv preprint arXiv:2104.08743*.

Xiao Yang and 1 others. 2023. Fingpt: Open-source financial large language models. *arXiv preprint arXiv:2306.06031*.