FinRAG: A Retrieval-Based Financial Analyst

If you're looking for an AI model in finance, knows trust funds, 6.5B params, nails BLEU scores...

Krrish Chawla
Department of Computer Science
Stanford University
krrish@stanford.edu

Allen Naliath
Department of Computer Science
Stanford University
analiath@stanford.edu

Abstract

FinRAG aims to enhance retrieval-augmented generation (RAG) methods for financial data analysis, which often struggle with high-level queries using extensive low-level data. To address these limitations, we propose an improved RAPTOR model utilizing a tree-based retrieval approach with metadata-based clustering, better suited for financial data. By clustering documents based on specific metadata attributes such as sector, company, and year, our approach aligns the retrieval process more closely with the inherent structure of financial documents, a significant departure from the naive RAPTOR model and standard retrieval methods which rely on similarity-based approaches. Training on the FinQA dataset, we convert tabular data into natural language using a large language model and create benchmark questions requiring multi-document knowledge integration. Our evaluation demonstrates that metadata-based clustering significantly outperforms traditional methods, exhibiting notable enhancements in LLM-based correctness evaluations, confirming its effectiveness in accurately answering complex financial queries. The implications for financial analysis are profound – this structured approach not only improves retrieval accuracy but also ensures that financial analysts can access and interpret relevant information more effectively, capitalizing on the typical data formats found in financial documents.

1 Key Information

- Stanford CS224N Custom Project
- **TA mentor:** Anna Goldie; **External mentor:** Parth Sarthi
- **Team Contributions:**
  Krrish Chawla: Research, Dataset Creation, RAPTOR Clustering, Benchmarking
  Allen Naliath: Research, Preprocessing, Experiments, Evaluations

2 Introduction

Imagine being a financial analyst. To make informed investment decisions, you would have to study extensive financial documents and burn through the math in the financial statements to understand trends about a company. Further, you would also need general information about the competitors of the company, which takes more in depth analysis of more financial documents and experience in the industry. Now imagine having an assistant do all that for you, with the extra benefit of an extremely quick turn around time per query. This is where an AI language model comes in.

Analyzing large volumes of financial data is a complex task that often requires the expertise of financial analysts. These professionals need to sift through extensive datasets, summarizing and utilizing the information to answer specific questions. While AI models offer potential solutions to simplify this process, they face significant challenges due to the sheer volume and numerical nature

of financial data. Simply feeding all this data into an AI language model doesn't work effectively, as current models struggle to process and understand it properly.

Existing methods like the RAPTOR [1] framework use a tree-based retrieval system that organizes information based on textual similarity. This approach, however, falls short when applied to financial documents. Financial reports from different companies may appear textually similar but contain vastly different numerical data. Thus, clustering based on text alone doesn't capture the essential distinctions needed for accurate financial analysis. This gap in effectiveness highlights the need for a more tailored approach to handle financial data's unique structure.

To tackle this problem, we have developed an enhanced version of the RAPTOR model that uses metadata-based clustering. Instead of relying on text similarity, our model organizes documents by metadata attributes such as company, sector, and year. This method aligns with the inherent structure of financial data, allowing for more precise retrieval and analysis. By clustering information in this manner, our model can better handle complex queries and provide more relevant answers, enhancing the overall efficiency of financial data analysis.

Our preliminary experiments using the FinQA dataset show promising results. While traditional metrics like F1 scores showed minimal improvements, LLM-based correctness evaluations indicated significant enhancements in the accuracy of our model's answers. This suggests that our metadata-focused approach not only improves retrieval quality but also better meets the needs of financial analysts. The implications of this research are substantial, potentially transforming how financial data is analyzed and interpreted, thereby increasing the leverage and effectiveness of financial professionals.

3 Related Work

In the realm of financial data analysis and retrieval-augmented generation (RAG), several key papers have paved the way for the development of more sophisticated models and techniques. We review these foundational works and situate our contribution within the broader context of existing research.

The RAPTOR model[1] serves as a crucial starting point for our research. RAPTOR's innovative approach constructs a tree by recursively embedding, clustering, and summarizing text chunks, thereby enabling holistic understanding across lengthy documents. This method demonstrated significant improvements in tasks requiring complex, multi-step reasoning by integrating information at different levels of abstraction. However, RAPTOR primarily focuses on text-based similarity, which poses challenges when applied to structured financial documents with numerical data and metadata.

Complementing the RAPTOR model, the Tree of Reviews[2] (ToR) introduces a dynamic retrieval framework for multi-hop question answering. Unlike traditional chain-based retrieval methods, ToR dynamically decides on search initiation, acceptance, or rejection based on the relevance of retrieved paragraphs, thereby reducing the impact of irrelevant information and single reasoning errors. This dynamic, tree-based structure enhances the ability to manage and integrate complex reasoning paths, demonstrating state-of-the-art performance on multi-hop QA tasks. This concept of using a tree structure for retrieval relates to our approach to organize financial data more effectively, although our financial documents can't use their paragraph-based approach, which contributes to the significance of metadata-based clustering.

We see an approach to dealing with financial data in "Financial Report Chunking for Effective Retrieval Augmented Generation" [3], which explores a crucial aspect of RAG by proposing an expanded approach to chunk documents based on structural elements rather than mere paragraphs. This approach leverages the inherent structure of financial documents to improve chunking and retrieval accuracy, which is pivotal for handling extensive numerical and tabular data in financial reports. This work demonstrates the importance of considering document structure in retrieval tasks, which aligns closely with our strategy of metadata-based clustering.

An important contribution in the realm of financial language models is FinGPT [4], which aims to democratize access to high-quality financial data. FinGPT provides an open-source alternative that emphasizes a data-centric approach. It features an automatic data curation pipeline and a lightweight adaptation technique, making it accessible for researchers and practitioners. FinGPT supports applications like robo-advising, algorithmic trading, and low-code development, fostering

innovation within the AI4Finance community. This aligns with our work in enhancing financial data retrieval, as it underscores the importance of transparent, collaborative efforts in developing effective financial language models.

The FinQA dataset [5], introduced to facilitate robust numerical reasoning over financial data, underscores the complexity of analyzing large corpora of financial documents. This dataset, comprising QA pairs over financial reports written by experts, highlights the limitations of large pre-trained models in acquiring financial knowledge and performing complex numerical reasoning. Our work builds on the FinQA dataset by enhancing retrieval capabilities, enabling more effective financial analysis and decision-making.

Moreover, the challenges of utilizing long contexts in language models, as discussed in "Lost in the Middle,"[6] reveal significant performance degradation when models must access relevant information from the middle of long contexts. This insight is crucial for our approach as it highlights the necessity of effective retrieval mechanisms to ensure that relevant financial information is readily accessible and utilized correctly by the model.

Finally, the concept of chain-of-thought prompting[7], which elicits reasoning in large language models, demonstrates substantial empirical gains in complex reasoning tasks. By incorporating intermediate reasoning steps, this method improves performance across various domains, including arithmetic and commonsense reasoning. This foundational idea supports our approach by emphasizing the need for structured reasoning in financial data analysis.

In summary, our work builds upon these significant contributions in these papers and many more by addressing the specific challenges of financial data retrieval and analysis. By looking to use metadata-based clustering and enhancing retrieval mechanisms, we provide a more accurate and efficient tool for financial analysts, extending the capabilities of existing RAG models and frameworks.

4 Approach

4.1 Data Preprocessing and Summarization

Our enhanced RAPTOR[1] model addresses the challenges of financial document retrieval by using metadata-based clustering. To begin, we meticulously preprocessed the FinQA dataset to align it with the RAPTOR framework.

4.1.1 Document Preparation

From each document in the FinQA[5] dataset, we extracted the ticker, pretext (text before the table), table (containing tabular information), post-text (text after the table), and document ID. The document ID provided crucial metadata, including the company name, year, and document name. Based on this metadata, we organized the documents into hierarchical clusters:

- Yearly Clustering: Documents were first grouped by year.
- Company Clustering: Within each year, documents were further grouped by company.
- Sector Clustering: Using an LLM, we categorized companies into nine sectors based on their industry, providing sector-level overviews.

To summarize the information within these documents, we passed the pretext, table, post-text, ticker, and ID into an LLM, specifically GPT-4.0[8], to generate natural language summaries using the following prompt:

```
Write a summary of the following, including as many key details as possible:
{context}:
```

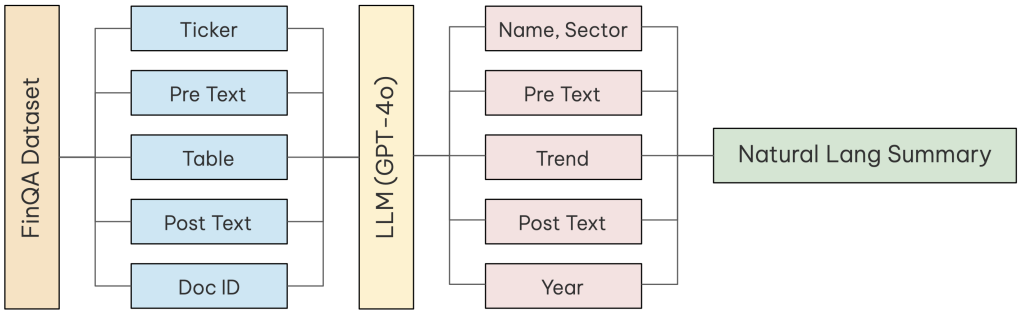

Figure 1: Natural Language Conversion from FinQA dataset.

4.1.2 Hierarchical Summarization

The summarization process involved multiple levels:

1. Document-Level Summarization: Individual documents were summarized.
2. Yearly Summarization: Document-level summaries were aggregated by year and summarized into 300-token summaries.
3. Company-Level Summarization: Yearly summaries were aggregated by company and summarized into 200-token summaries.
4. Sector-Level Summarization: Company summaries were aggregated by sector and summarized into 100-token summaries.
5. Overall Summary: Sector summaries were aggregated to provide an overall economic summary in 1,000 tokens.

This hierarchical structure ensured comprehensive and meaningful summaries at various levels of detail.

4.2 Baselines

To evaluate the effectiveness of our enhanced RAPTOR model, we established two baseline evaluations and compared it against our approach:

Basic RAG: In the basic RAG approach, we performed retrieval-augmented generation without any summarization. Documents were retrieved based on similarity scores, providing a simple yet essential benchmark for understanding the basic retrieval capabilities.

Naive RAPTOR: The Naive RAPTOR model followed the original RAPTOR[1] approach, summarizing documents and clustering them using Gaussian Mixture Models (GMM). However, unlike our enhanced approach, this method did not utilize metadata-based clustering and grouped documents solely based on text-based similarity.

Enhanced RAPTOR with Metadata Clustering: Our enhanced RAPTOR model incorporated several modifications. We adapted RAPTOR to process JSON data and perform metadata-based clustering, grouping documents by sector, company, and year. This method provided a more accurate and efficient way to organize financial documents compared to the traditional text-based clustering approach.

4.3 Modifications and Enhancements

he development of our enhanced RAPTOR model involved significant custom coding efforts. We created a comprehensive data preprocessing pipeline to extract and summarize financial documents, ensuring that each document was accurately labeled with its corresponding metadata. This metadata included crucial details such as the company, year, and sector, which were essential for the clustering process.

One of the key modifications was adapting the RAPTOR framework to process JSON data instead of traditional text-based inputs. This change was necessary to handle the structured information derived from financial documents effectively. Each document was parsed to extract pretext, post-text, numerical data, the ticker symbol, and the document ID. This information was then organized into a JSON format, with each entry containing a summarized document and its metadata.

The integration of GPT-4.0 for summarization tasks was pivotal in enabling the model to manage the hierarchical data structure and generate meaningful summaries at each level. This approach ensured that the summarizations retained critical information while being concise enough to facilitate efficient retrieval. By creating a well-structured JSON dataset and modifying RAPTOR to handle this format, we improved the model's ability to process and analyze financial documents. The combination of these modifications and enhancements has resulted in a robust model capable of handling the complexities of financial document retrieval, thereby providing a powerful tool for financial analysts.

4.4 Metadata-Based Clustering

Metadata-based clustering leverages the inherent structure of financial data by grouping documents according to attributes such as company, year, and sector. This method aligns with the organizational practices of financial analysts, ensuring that related financial documents are grouped together, which enhances retrieval accuracy and relevance. Instead of using Gaussian Mixture Models (GMM) for clustering nodes, we utilized the metadata inherent in the datasets. This approach streamlined the clustering process and minimized the risk of grouping unrelated documents together, thereby maintaining the integrity of the information.

The advantages of metadata-based clustering over traditional GMM clustering are significant. This method is faster and more efficient as clusters are predefined by metadata, eliminating the need for complex similarity calculations. By ensuring that documents from the same company, year, or sector are grouped together, the model preserves the contextual relevance of the information, leading to more accurate and meaningful summaries. Overall, metadata-based clustering enhances the model's performance, making it a more effective tool for financial document analysis and retrieval. This development not only improves retrieval accuracy but also ensures that complex financial queries can be addressed with greater precision and depth.

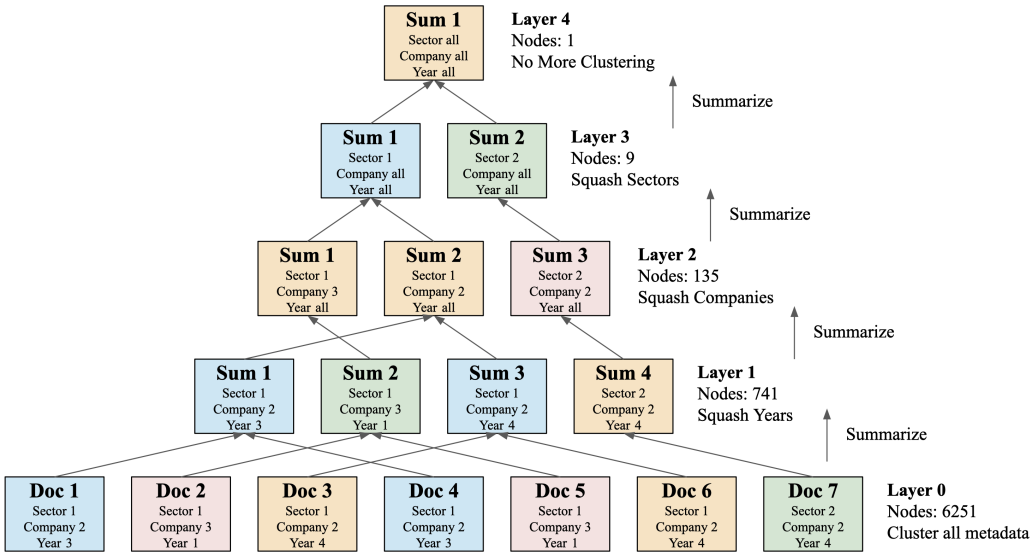

Figure 2: Metadata Clustering visualization to build FinRAG RAPTOR Tree

5 Experiments

We conducted a comprehensive set of experiments to evaluate the performance of our models and validate our hypothesis. The evaluation metric as described below is specifically tailored to the downstream task of financial analysis. The performance of our FinRAG model has been compared against other baseline models, as discussed in the results table, to validate its performance in the financial domain.

5.1 Data

We utilized the processed dataset derived from the FinQA Dataset [5]. The dataset structure and format was according to the clustering algorithm utilized. For Gaussian Mixture Model (GMM) word similarity clustering, we used a regular text dataset consisting of summaries of all documents concatenated together from the FinQA dataset. For metadata-based clustering, a JSON dataset was created, which included not only the summaries but also metadata associated with each summary, specifically the sector, company, and year associated with each document.

5.2 Evaluation Method

Since the objective of FinRAG is to replace or mimic a full human financial analyst, the standard question-answer pairs present in the FinQA dataset were seemingly insufficient. In FinQA, questions are targeted towards specific documents, and evaluation is based on execution accuracy for mathematical calculations, which does not fully incorporate the comprehensive analysis expected of a financial analyst.

5.2.1 New Benchmark - Long Form, Company Specific Questions

To evaluate the model's holistic understanding of the economy, we created a new benchmark set consisting of 30 questions. These questions require the model to retrieve information from multiple documents, demonstrate a comprehensive understanding of financial texts, and perform extensive reasoning. The questions were stored in JSON format, containing question and ground truth 'golden answer' pairs. The questions were designed using a large language model, specifically Anthropic's Claude Opus [9], due to its extensive context window of about 200k tokens. Since the FinQA dataset exceeded the context window of this model as well, and considering the known characteristic of models to lose context from the middle [6], we pruned the dataset by performing a random dropout of documents with a dropout probability of 0.95. This reduction allowed the dataset to fit within Claude Opus's context window.

5.2.2 Initial Evaluation Metric

Initially, we utilized BLEU, ROUGE, and F1 scores to assess the model's comprehensiveness and accuracy in answering the questions. BLEU measures similarity between generated and reference answers using n-gram scores, which is well suited for comparing the generated answers with the gold standard. ROUGE assesses answer quality with three variants: ROUGE-1 (unigram overlaps), ROUGE-2 (bigram overlaps), and ROUGE-L (longest common subsequence). F1 score measures precision and recall. For each question in the test set, the model's answer was compared to the golden answer, and the three scores were computed. These scores were averaged across all questions to output a final score. However, this approach was not enough as it primarily measured the linguistic coherence of the model's response without adequately assessing numerical accuracy and the overall quality of the answer from a financial analyst's perspective.

5.2.3 LLM Quality and Accuracy Evaluation

While quantitative scores are useful for evaluating the model's output in terms of linguistic similarity with the ground truth, they are insufficient in assessing the model's holistic understanding of financial data and the quality of financial analysis. Manual inspection of the results is necessary for a thorough evaluation. Building on the idea that LLMs, when specifically prompted, can serve as effective evaluators, and considering that humans agree with GPT-4's reasoning as much as they agree with other humans' reasoning as shown in the Direct Preference Optimization paper [10], we utilized

an LLM to evaluate the quality of responses. The model utilized was GPT4-o [8], OpenAI's most advanced model at the time of writing. This model was prompted to act as a financial analyst expert and was instructed to judge the quality of a financial analyst's response in terms of quality, factual or numerical correctness, comprehensiveness, coherence, and helpfulness to another financial analyst. The type of judgement would be a score for each answer, which would be one of the following: 0, 0.5 and 1. In the end, all the scores would be totalled up per question to return one final score for the test set of questions.

5.3 Experiment Details

For our evaluation experiments, we built the RAPTOR tree on the summarized FinQA dataset. We performed the following three experiments:

1. **Regular RAG:** RAPTOR Tree built on summarized text. Embedding Model; OpenAI; Summarization Model: GPT4o; QA Model: GPT4o; Clustering method: GMM Word Similarity; Top K Retrieval Nodes: 5; Summarization length: 100 tokens; Number of tree layers: 1; Retrieval method: Collapsed Tree. Note: only leaf layer (layer 0) used for retrieval, which does not involve any summarization, which emulates regular RAG.
2. **Naive RAPTOR:** RAPTOR Tree built on summarized text. Embedding Model; OpenAI; Summarization Model: GPT4o; QA Model: GPT4o; Clustering method: GMM Word Similarity; Top K Retrieval Nodes: 5; Summarization length: 100 tokens; Number of tree layers: 5; Retrieval method: Collapsed Tree. Note: all tree layers used in retrieval.
3. **FinRAG:** RAPTOR Tree built on json data with summarized text and metadata. Embedding Model; OpenAI; Summarization Model: GPT4o; QA Model: GPT4o; Clustering method: Metadata Clustering; Top K Retrieval Nodes: 5; Summarization length: 300 tokens (layer 1), 200 tokens (layer 2), 100 tokens (layer 3), 1000 tokens (layer 4); Number of tree layers: 5; Retrieval method: Collapsed Tree. Note: all tree layers used in retrieval.

5.3.1 Varying Summarization Lengths and Number of Nodes per layer in FinRAG

As noted above, we used varying summarization lengths in FinRAG with metadata clustering. The justification is as below.

- **Layer 0 -> Layer 1: 6251 nodes -> 741 nodes** Summarization length: 300. This transition clusters documents of same sector, company and year, thus, more context was required to get a good understanding of the document and not to loose out on numerical details. 6251 nodes represent 6251 documents in layer 0.
- **Layer 1 -> Layer 2: 741 nodes -> 135 nodes** Summarization length: 200. This transition clusters documents of the same sector, company across years, thus, lesser information was required to capture a general trend of a company over a year. 741 nodes represent summaries for companies over the years in layer 1.
- **Layer 2 -> Layer 3: 135 nodes -> 9 nodes** Summarization length: 100. This transition clusters documents of the same sector, across companies and years, thus, even lesser information was required to capture the trend of a sector over the years. 135 nodes represent summaries of all companies in layer 2.
- **Layer 3 -> Layer 4: 9 nodes -> 1 node** Summarization length: 1000. This transition clusters information from all sectors into one last summary, requiring a large length. 9 nodes represent number of sectors in layer 3, 1 node represents the final summary in layer 4.

This choice of dynamic summarization lengths effectively captured financial information in a document to retain trend information across metadata.

5.4 Results

5.4.1 Quantitative Conventional Results

The F1, ROUGE and BLEU Results are as described in Table 1

Experiment	BLEU	ROUGE-1	ROUGE-2	ROUGE-L	F1 Score
RAPTOR Leaf Layer	0.0694	0.3323	0.1564	0.2144	0.4008
Naive RAPTOR	0.0679	0.3390	0.1602	0.2202	0.4005
FinRAG RAPTOR	0.0638	0.3075	0.1447	0.1995	0.3806

Table 1: Quantitative Results for all three models

5.4.2 Qualitative LLM Eval Results

GPT-4o's evaluation of each model's output responses are as described in Table 2. Since an LLM's evaluation score is not deterministic, we conducted this evaluation twice to further corroborate the trend in the results.

Experiment	GPT-4o Eval Score 1	GPT-4o Eval Score 2
RAPTOR Leaf Layer	10.5	10
Naive RAPTOR	13.5	13
FinRAG RAPTOR	21	20.5

Table 2: Quantitative Reuslts for all three models

A visualization of the first of the two results from the LLM eval is as in Figure 3

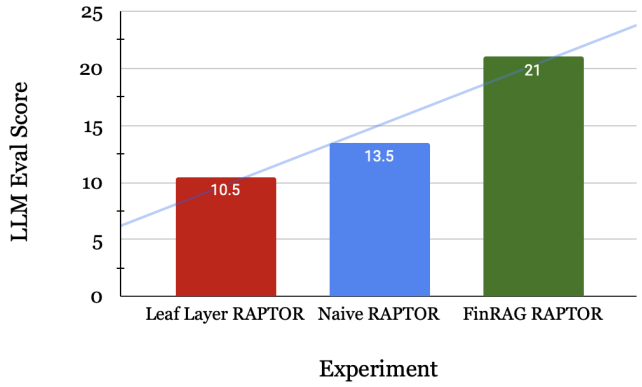

Figure 3: Experiment vs LLM Evaluation Total Score

6 Analysis

The results show the challenges of applying the naive RAPTOR model to financial retrieval tasks. The insignificant difference between the naive RAPTOR approach and simple leaf layer retrieval (regular RAG) suggests that the GMM based word similarity clustering method used by RAPTOR does not effectively capture the inherent structure and metadata relationships present in financial documents.

However, the metadata-based clustering method introduced in our FinRAG model shows promising results. The significant improvement in LLM-based evaluation scores indicates that organizing financial data based on metadata like sector, company, and year allows for more accurate retrieval and integration of important information required to answer complex financial queries.

6.1 Quantitative Analysis

Quantitatively, The conventional metrics like BLEU, ROUGE, and F1 scores did not show any improvements. Naive RAPTOR improved the results very slightly over regular RAG, however

FinRAG further pulled scores slightly down. The change in scores is negligible enough to consider them the same. This may be due to these metrics measuring linguistic correctness in the sentence compared to the ground truth, while not measuring financial analysis quality very precisely.

The LLM evaluation shows the importance of assessing the quality and correctness of responses from a domain-specific perspective. Regular RAG performed the worst at a score of around 10, Naive RAPTOR at a score of around 13, and FinRAG at a score of around 21. For both experiments, there is almost a two time boost in performance from Naive RAPTOR to FinRAG.

These results show the drawbacks of naive word-based clustering techniques when applied to structured data formats present in the financial domain. By utilizing the metadata-based organization of financial documents, our FinRAG model demonstrates the potential for enhancing retrieval accuracy, improving the overall quality of financial analysis and decision-making processes.

6.2 Qualitative Analysis

Qualitatively, FinRAG responses were far superior to other responses. There was much lesser hallucination. This corroborates our argument about LLM evaluation being better than score based evaluation. For example, if the LLM hallucinates a number, from a BLEU or F1 perspective, that will pass. However, an LLM will penalize the response for numerical inaccuracy and lack of information relevant to a financial decision maker. Overall, the responses of FinRAG were seemingly more helpful to a financial analyst than the other baseline models.

7 Conclusion

In this research, we overcome the obstacles of retrieving and integrating information from extensive financial data to enable complex analytical and decision making tasks. Our FinRAG model develops on top of RAPTOR tree based retrieval method, enables RAPTOR to work with JSON data instead of plain text, and also introduces metadata based clustering to improve the tree structure for financial data. By organizing data based on inherent sections like sector, company, and year, FinRAG conforms to the retrieval process with the unique structure and relationships within the financial domain. We also demonstrated the usability of financial data with a model like this by converting tabular and numerical data into natural language to preserve the general trend and emotion of the data. To test our model's usefullness to a financial analyst, we created a new becnhmark set of 30 questions which require the model to retrieve information from multiple documents and analyse over trends of a company in years, or trends of multiple companies in a sector to output an answer compared to the traditional QA tasks performed on singular documents. Our LLM-based evaluation demonstrated FinRAG's promising potential for improving retrieval accuracy on our benchmark financial analysis questions compared to baseline methods.

While standard metrics like BLEU and ROUGE did not reflect substantial gains, our findings show FinRAG's main achievement - proving to be a either a replacement or an aiding tool for a financial analyst. This also shows the importance of developing domain-specific evaluation frameworks beyond linguistic similarity for tasks involving structured or numerical data. In the future, further research could explore advanced metadata-driven clustering, dynamic summarization strategies, integration of domain specialized reasoning capabilities, and extending this metadata-focused approach to other structured data domains. Also, dataset creation and benchmark creation which better aligns with analysts' goals is key to improving the quality of this work.

8 Ethical Statement

Financial Retrieval, though useful in its way to replace or aid analysts, comes with its own set of ethical considerations. FinRAG, our design of the RAPTOR model, relies heavily on the summaries generated from the FinQA dataset. The act of converting the dataset into natural language may have significant problems. For example, the natural language output by the model may not capture the exact numerical values or trends since LLMs cannot do math they have not seen before. Incorrect summaries could lead to incorrect financial analyses, resulting in wrong investment decisions or misinformation in financial planning, which can be catastrophic. To mitigate this risk, while preparing the natural language dataset from a financial dataset, there should be multiple layers of validation,

which could be an LLM evaluation layer again, which adds an additional check to verify if the conversion into natural language conforms to the tabular numeric data. As the Chain of Thought paper suggests [7], multiple self evaluation prompts can help greatly improve the accuracy of an LLM. We could not implement this strategy in this paper due to limited compute, however if this strategy is implemented, the factual precision of the model should greatly be improved.

Second, financial retrieval can come with its biases and opinions. Since the underlying QA model is a language model like GPT-4, its biases could always get into the responses, bringing biases outside of the context from retrieving the nodes by RAPTOR. Biases inherently present in GPT-4 may have its influence on the response of the model even when prompted with the retrieved context. If the training corpus of the underlying model has significant biases in it, it would pass it on to its response. For example, if the training corpus of the model has a lot of information about a company performing poorly, the model would think that this company does not perform very well probabilistically, and will neutralize the the good performance of that company for a specific period in the model's opinion. This can again lead to misinformation while outputting financial analyses. To mitigate this risk, the most straightforward way is to prompt the model to strictly stick to the input context, however that cannot be fully trusted as well. Thus, as an enhanced strategy, the model responses can be validated by looking at out of corpus context, mitigating the model outputting its own biased sentences. However, even that does not fully mitigate this risk because while it may prevent the model from explicitly outputting out of context opinions, it may not stop the model from making inferences internally while generating text. To better alleviate this risk, ensuring a diverse training data is necessary, though it is very hard to quantify a very diverse and well informed dataset for financial tasks.

Since financial analyses can lead to great impacts on society, these risks are substantial and need to be studied carefully to mitigate misinformation and misrepresentation in financial decision making. Apart from these risks, there could be a multitude of other risks, like preventing hallucination, weighting penalizing (assigning an evaluation score with different tolerances for different aspects of the output, like numerical accuracy, linguistic accuracy, financial analysis quality).

References

[1] Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D. Manning. Raptor: Recursive abstractive processing for tree-organized retrieval. In *International Conference on Learning Representations (ICLR)*, 2024.

[2] Li Jiapeng, Liu Runze, Li Yabo, Zhou Tong, Li Mingling, and Chen Xiang. Tree of reviews: A tree-based dynamic iterative retrieval framework for multi-hop question answering. *arXiv preprint arXiv:2404.14464*, 2024.

[3] Antonio Jimeno Yepes, Yao You, Jan Milczek, Sebastian Laverde, and Leah Li. Financial report chunking for effective retrieval augmented generation. *arXiv preprint arXiv:2402.05131*, 2024.

[4] Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. Fingpt: Open-source financial large language models. *arXiv preprint arXiv:2306.06031*, 2023.

[5] Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan R Routledge, et al. Finqa: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3697–3711, 2021.

[6] Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

[7] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[8] OpenAI. Gpt-4o: Openai's new language model, 2024.

[9] AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 2024.

[10] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.