RETRIEVAL-AUGMENTED REASONING FOR CHARTERED ACCOUNTANCY

Jatin Gupta
Dept. of Computer Science and Engineering
Sharda University, Greater Noida, India
jatingupta261001@gmail.com

Akhil Sharma
Dept. of Computer Science and Engineering
Sharda University, Greater Noida, India
sharmaakhil944@gmail.com

Saransh Singhania
Dept. of Computer Science and Engineering
Sharda University, Greater Noida, India
saransh060123@gmail.com

Ali Imam Abidi*
Dept. of Computer Science and Engineering
Sharda University, Greater Noida, India
aliabidi4685@gmail.com

ABSTRACT

The inception of Large Language Models (LLMs) has catalyzed AI adoption in the finance sector, yet their reliability in complex, jurisdiction-specific tasks like Indian Chartered Accountancy (CA) remains limited. The models display difficulty in executing numerical tasks which require multiple steps while also needing advanced knowledge about legal regulations and the method of scaling their operations is not feasible in settings which have limited access to resources. We present CA-ThinkFlow as a parameter-efficient Retrieval-Augmented Generation (RAG) framework which operates with a 14B, 4-bit-quantized reasoning model, 14B-DeepSeek-R1, and a layout-aware Docling extraction system which maintains document structure during extraction. CA-ThinkFlow uses a basic RAG method which automatically adds retrieved information into the prompt, while it depends on the model's built-in Chain-of-Thought (CoT) functions to create context and produce correct answers. The system we developed system operates at performance levels which match large proprietary models when we tested it on the multi-level CA-Ben benchmark, achieving Scholastic Reliability Coefficient (SRC) results which equal 68.75% of GPT-4o and Claude 3.5 Sonnet. The framework shows high efficiency and strength in handling parameters, but essential reasoning abilities fail to process complex regulatory texts which exist in fields such as Taxation.

Keywords Retrieval-Augmented Generation · Financial AI · Large Language Models · Parameter Efficiency · Chartered Accountancy.

1 Introduction

Large Language Models (LLMs) have witnessed significant adoption in financial services, with more than 70% of prominent organizations planning to use them for risk modeling by late 2024 [1, 2]. Despite advancements in efficiency of areas such as compliance automation, LLMs have dependability problems in regulated sectors, including hallucinations in complicated legal activities, particularly those involving Income Tax Act regulations, ICAI standards, and GST computations [3, 4, 5]. Furthermore, their expensive inference, high resource requirements, privacy threats from cloud dependency, and environmental effect from huge training limit their usefulness for resource-constrained environments [6, 7]. Generalist models, such as GPT-4o, obtain just 13-20% accuracy on CA benchmarks due to limited "tail" information and multi-step reasoning. Domain-specific Small Language Models (SLMs) address this by using compact, private, and cost-effective designs that are comparable to LLMs for financial tasks [8, 9, 10]. To achieve a better and comparable performance on the CA benchmark and lower computational overhead due to the high parameter count of such models, we present a RAG pipeline with a reasoning 14B-DeepSeek-R1 model.

*Corresponding author: aliabidi4685@gmail.com

2 Related Works

The recent rise of "System 2" reasoning models has created a new paradigm that shifts reasoning away from pattern matching toward direct logical deduction. OpenAI's o1 and o3 models introduced test-time compute scaling through their ability to generate internal chain-of-thought [11, 12]. DeepSeek-R1, an open-source development, appeared in January 2025, showcasing distilled reasoning abilities that rivaled those of more extensive systems [13, 14]. Subsequently, financial adaptations led to the creation of Fin-R1, a model with 7 billion parameters, and Fino1, which focuses on financial reasoning, achieved through supervised fine-tuning on data tailored to the domain [15, 16, 17].

Financial and legal compliance now require all responses to be supported by original documents, since retrieval-augmented generation (RAG) has become essential in these areas [18]. Fin-RAG uses domain-specific embeddings to analyze financial documents, while compliance designs focus on auditability and traceability [19, 20]. The primary obstacle for financial RAG systems arises from document parsing because standard OCR technology fails to maintain the intricate formats found in tax documents which contain multiple column tables and mathematical symbols. Docling developed into an exact document extractor that keeps document structure intact through its superior TableFormer and DocLayNet models which surpass Dolphin and PaddleOCR as alternative solutions that we have used in our scraping process [21, 22, 23].

The finance industry needs LLMs to achieve both high accuracy and high precision before they can become dependable tools for everyday usage. To solve this problem, researchers have developed multiple methods which aim to enhance system reliability [16]. The Indian government shows its commitment to developing financial intelligent systems through ICAI's (Institute of Chartered Accountants of India's) support of domestic AI research which makes this research timely. Our framework builds on these efforts by aligning with the Indian regulatory environment and integrating retrieval methods to ensure reasoning is grounded in credible sources.

3 Methodology

Our system combines the adaptable yet straightforward features of the Retrieval-Augmented Generation (RAG) method which extracts subtle details from embedded financial and legal data that has been collected together with the base model's capacity to perform structured reasoning and mimic natural cognitive thinking.

3.1 Data Preparation

We developed an extensive database of Indian financial sector knowledge. The database contained information from ICAI open-source materials tax regulations and financial textbooks and associated references. The Docling Document Converter was used to extract text from the selected financial documents which were then converted into a structured format.

3.1.1 Text Chunking and Splitting

The Markdown text was segmented into 1000-character chunks with 200-character overlap, matching established financial RAG benchmarks: Fin-RAG uses 1024 chars (15% overlap) [18], ConvFinQA uses 900 chars (200-char overlap) [24], and DocMath uses 1100 chars (20% overlap) [25]. The Markdown text was divided into segments which measured 1000 characters and included an overlap of 200 characters. This division of text materials matched established financial RAG systems, such as Fin-RAG using 1024 chars with 15% overlap, ConvFinQA using 900 chars with 200-char overlap, and DocMath using 1100 chars with 20% overlap. The selected size achieves an optimal solution because it combines two factors, which include embedding model limits and data structure and retrieval precision requirements and contextual continuity needs which occur between chunk boundaries.

3.1.2 Embedding Generation

Each text chunk t_i was encoded using the Qwen-Embedding-0.6B model [26], selected for its multilingual support and high efficiency. Given an input text t_i, the embedding vector v_i is computed as given in Equation 1, where the representation of the first hidden state (analogous to the [CLS] token) is extracted and normalized. Batch encoding was applied for efficient computation, producing a dense embedding matrix $\mathbf{V} \in \mathbb{R}^{n \times d}$, where n denotes the number of chunks and d the embedding dimension.

$$v_i = \text{QwenEmb}\left(\text{Tokenizer}(t_i)\right)_{[0]} \tag{1}$$

3.1.3 Vector Indexing

The embeddings were stored in a FAISS [27] index for L_2-based similarity search S, where $S(Q, t_i)$ denotes the similarity between a query embedding v_Q and document chunk v_i. The FAISS index and associated metadata (chunk mappings) were serialized as `index.faiss` and `index.pkl` respectively, forming the foundation of the vector retrieval layer.

$$S(Q, t_i) = -\|v_Q - v_i\|_2 \tag{2}$$

3.2 System Design

3.2.1 Base Model Selection

We use the 'DeepSeek-R1-Distill-Qwen-14B' model commonly known as '14B-DeepSeek-R1', in a 4-bit (Q4_K_M) quantized format [28]. This choice balances reasoning strength and computational efficiency. The quantization enables deployment on resource-constrained environments while preserving reasoning quality through Chain-of-Thought (CoT) capabilities, crucial for the multi-step problem-solving found in Chartered Accountancy exams.

3.2.2 Prompt and Context Retrieval

To preserve consistency and provide a fair evaluation, we used the exact standardized system prompt described in the original CA-Ben study [8]. The model processes the retrieved knowledge from the vector store which serves as contextual information to be used in the prompt.

3.2.3 Context Integration and Reasoning

The system transfers the most relevant context that it found in the vector store to the base model. The system uses Chain-of-Thought reasoning through reinforcement learning that is built into the internal core of 14B-DeepSeek-R1 to create a response that combines both the user query and the retrieved text.

3.3 Inference and Workflow Logic

The complete inference workflow operates through a unified Retrieval-Augmented Generation system which functions as a RAG loop (Algorithm 1). The system starts its process by receiving a query (Q) which it uses to create embeddings (E_q) and retrieve the top context (C) from the vector store (V). The system directly adds this retrieved context into the standard CA-Ben prompt template. For an apples-to-apples comparison, we set the temperature to 0.75, consistent with the CA-Ben study [8], ensuring moderate randomness that supports learning in complex scenarios. The model evaluates the query together with the context to generate the final answer (A) through controlled generation.

Algorithm 1 Inference Workflow of the RAG Framework

Require: User query Q
Ensure: Final answer A
$E_q \leftarrow \text{Embed}(Q)$
$C \leftarrow V.\text{search}(E_q, k = 1)$
Prompt $\leftarrow \text{Template}(Q, C)$
$A \leftarrow \text{LLM.generate}(\text{Prompt}, \text{temperature} = 0.75)$
return A

The adoption of this streamlined workflow simplifies CA-ThinkFlow's architecture by passing the retrieved text directly to the language model. By relying on a straightforward RAG approach, the system ensures that the model's CoT reasoning is continuously focused on synthesizing the provided context to answer the query accurately.

4 Experimental Setup

The evaluation used zero-shot testing on the CA-Ben [2] benchmark because the models received no training data. The system evaluates performance according to CA-Ben study standards because it simulates an actual cold-start situation [8].

[2]Refer to the repository for benchmark: `https://github.com/thejatingupta7/LLMCA`

Table 1: Performance on Foundation, Intermediate, and Final-level Subjects

Models	Foundation		Intermediate						Final					
	F1	F2	I1	I2	I3	I4	I5	I6	FN1	FN2	FN3	FN4	FN5	FN6
CA-ThinkFlow	77.78	83.00	40.00	33.33	33.33	40.00	66.67	75.00	57.14	73.33	57.14	40.00	26.67	37.50
14B-Deepseek-R1	47.47	70.00	33.33	33.33	20.00	20.00	46.67	56.25	42.86	60.00	50.00	13.33	06.67	25.00
GPT-4o	50.00	58.00	46.66	73.33	20.00	20.00	86.66	75.00	71.43	53.33	78.57	53.33	33.33	41.67
LLAMA 3.3 70B Inst.	59.00	56.00	33.33	60.00	40.00	40.00	73.33	75.00	64.29	33.33	71.43	53.33	06.67	20.83
LLAMA 3.1 405B Inst.	53.00	59.00	40.00	53.33	20.00	40.00	86.66	56.25	64.29	46.67	71.43	13.33	26.67	41.67
MISTRAL Large	41.00	56.00	41.66	53.33	31.25	20.00	73.33	60.00	42.86	41.67	57.14	46.67	13.33	29.17
Claude 3.5 Sonnet	60.00	60.00	33.33	60.00	20.00	46.66	93.33	75.00	78.57	46.67	64.29	53.33	20.00	62.50
Microsoft Phi 4	56.00	62.00	46.66	46.66	33.33	33.33	66.66	68.75	64.29	53.33	57.14	26.67	06.67	41.67

Legend: **F1**: Business Math & Stats; **F2**: Business Econ & BCK; **I1**: Adv. Accounting; **I2**: Corp. Laws; **I3**: Taxation; **I4**: Cost & Mgmt. Acct.; **I5**: Auditing & Ethics; **I6**: Fin. & Strat. Mgmt.; **FN1**: Fin. Reporting; **FN2**: Adv. Fin. Mgmt.; **FN3**: Adv. Auditing; **FN4**: Direct Tax Laws; **FN5**: Indirect Tax Laws; **FN6**: Integrated Business Sol.; **Inst**: Instruct

4.1 Implementation Details

The implementation was completed on a workstation configuration which included two Intel Xeon CPUs and 65 GB RAM and two NVIDIA GeForce GTX 1080 Ti GPUs to meet its computational requirements. The software environment operated on Python version 3.11.9. PyTorch (CUDA 11.8) served as the deep learning and retrieval engine for the project. The data transformation was executed through the use of Transformers and FAISS-CPU dependencies. The complete process of retrieval and reasoning was conducted through the capabilities of LangChain and Ollama. The project used pandas and openpyxl for its data management tasks.

4.2 Evaluation Pipeline and Answer Extraction

To evaluate the models' responses, we utilized the automated Python testing pipeline and regex extraction methodology established in the CA-Ben study [8]. However, because the DeepSeek-R1 reasoning model is highly verbose and produces very long Chain-of-Thought traces, a preprocessing step was required before we could apply the standard extraction procedure.

The evaluation process was streamlined as follows:

1. **Reasoning Text Removal:** Reasoning models emit their intermediate thinking results within the `<think>...</think>` tokens which they use to deliver their ultimate response. The system used the regular expression patterns from Equation 3 to remove all text between the specified tags. The `DOTALL` modifier acted as a wildcard, allowing the matching of text that spanned multiple lines.

$$\text{pattern} = \texttt{r'<think>.*?</think>'}\ [\texttt{DOTALL}] \tag{3}$$

2. **Standardized Extraction and Validation:** After stripping the reasoning traces, we applied the standard CA-Ben regex pattern to reliably capture the final answer choice (A, B, C, or D). The extracted options were then compared against the ground-truth labels.

Overall accuracy was then calculated using the exact mathematical formulation detailed in the original CA-Ben methodology.

5 Results and Analysis

A detailed and thorough summary of the performance of each LLM over the 14 different domains of the CA-Ben benchmark is given in this section. The fine-grained accuracy scores in Table 1 depict the capacity of various models and are the main data source for the level-wise and model-specific analysis that follows. A comprehensive visual and statistical discussion of these results is demonstrated in the next subsections:

5.1 Exam Level-wise Breakdown

Table 2 displays the accuracies of various models evaluated across the three CA-Ben levels. As per observations, CA-ThinkFlow acquires the highest accuracy score at the Foundation level (80.39%), and also outperforms almost

all the given state-of-the-art models at the Final level (except GPT-4o and Claude-3.5-Sonnet). It also outperforms Mistral Large and the base 14B-Deepseek-R1 model, validating the impact of the RAG mechanism implemented. This validates the strong reasoning capability and robustness of CA-ThinkFlow across all difficulty levels in comparison to significantly larger and computationally inefficient models like Claude 3.5 Sonnet, GPT-4o, LLaMA-3.1-405B-Instruct, and others, especially at the Foundation and Final levels.

Table 2: Cumulative accuracy (%) of LLMs across CA-Ben levels.

Model	Foundation	Intermediate	Final
CA-ThinkFlow	**80.39**	48.05	48.63
14B-Deepseek-R1	58.73	34.93	32.97
GPT-4o	54.00	53.61	**55.28**
LLaMA-3.3-70B-Instruct	57.50	53.61	41.65
LLaMA-3.1-405B-Instruct	56.00	49.37	44.01
Mistral-Large	48.50	46.59	38.47
Claude-3.5-Sonnet	60.00	**54.72**	54.23
Microsoft-Phi-4	59.00	49.23	41.63

5.2 Subject-wise Breakdown

Figure 1 represents the accuracy of each model that was evaluated for each subject across the 14 domains of the CA-Ben benchmark. The bar chart with grouped bars directly and distinctly compares model performance in every subject area of Chartered Accountancy, thus throwing light on the different patterns, such as strengths and weaknesses across topics belonging to the foundational, intermediate, and final levels for all the models.

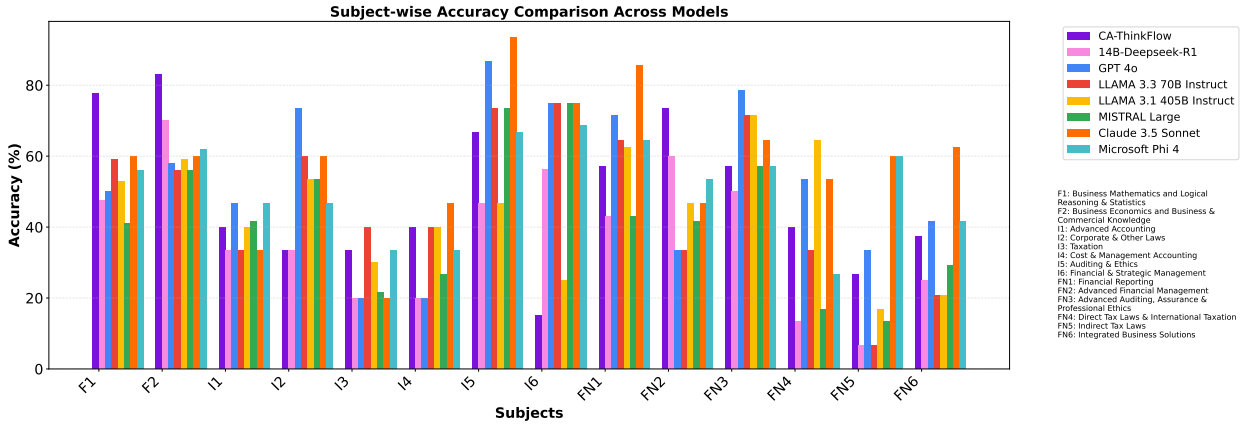

Figure 1: Subject-wise accuracy comparison of all evaluated models across 14 CA-Ben domains (F1–FN6). Each bar represents the model's accuracy for a specific subject, color-coded by model.

5.3 Model-Specific Comparative Analysis

CA-ThinkFlow demonstrated unique strengths across different domains in the CA benchmark, reflecting its varied capabilities in greater reasoning and numerical skills in the Foundation Level and comparable scores in Final Level. The radar chart in Figure 2 clearly visualizes an overall comparison of models across each individual exam. While some models display balanced performance across most domains, CA-ThinkFlow reveals a sharper contrast, excelling in certain areas like Business Economics and Business & Commercial Knowledge, Business Mathematics and Logical Reasoning & Statistics, Advanced Financial Management, and others with comparable performance compared to other SOTA models.

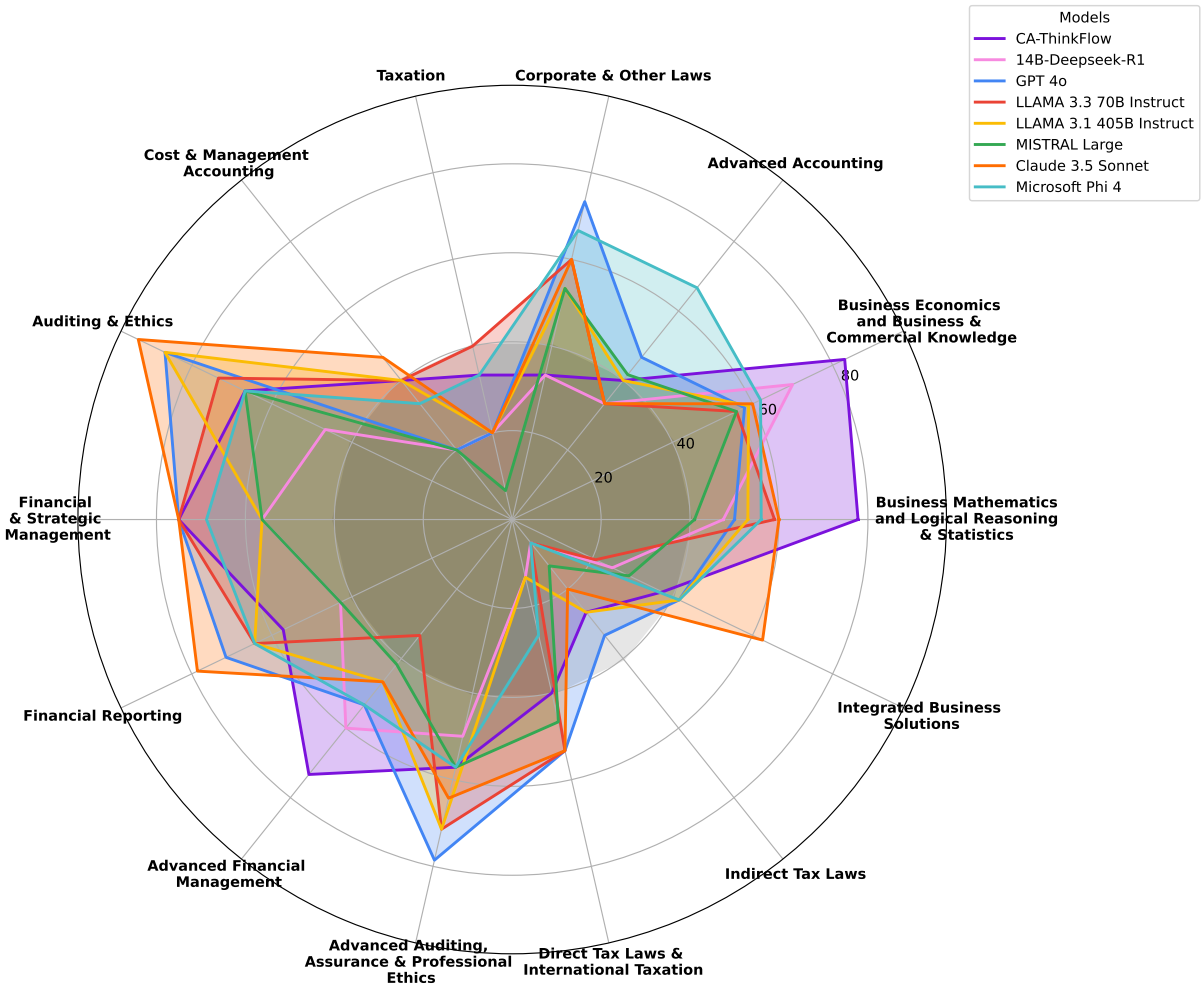

Figure 2: An overall comparison of models across each individual exam.

5.4 Scholastic Reliability Coefficient (SRC) Analysis

We evaluate the Scholastic Reliability Coefficient (SRC) to assess model consistency across increasing difficulty levels in CA-Ben. Table 3 shows that CA-ThinkFlow scored an SRC of 68.75% which matches the performance of top state-of-the-art systems including Claude 3.5 Sonnet and GPT-4o while exceeding other systems. The results show that CA-ThinkFlow operates as a parameter-efficient quantized system which delivers high reliability and consistent performance throughout its foundation and final-level assessments because it possesses strong domain adaptability and reasoning capabilities.

Table 3: Model rankings by Scholastic Reliability Coefficient (SRC).

Model Name	Foundation Passes	Intermediate Passes	Finals Passes	Weighted Score	SRC (%)
CA-ThinkFlow	2/2	4/6	4/6	22 / 32	**68.75**
14B-Deepseek-R1	2/2	2/6	3/6	15 / 32	46.87
GPT-4o	2/2	4/6	4/6	22 / 32	**68.75**
LLaMA-3.3-70B-Instruct	2/2	4/6	3/6	19 / 32	59.38
LLaMA-3.1-405B-Instruct	2/2	4/6	3/6	19 / 32	59.38
Mistral-Large	2/2	3/6	4/6	20 / 32	62.50
Claude 3.5 Sonnet	2/2	4/6	4/6	22 / 32	**68.75**
Microsoft-Phi-4	2/2	4/6	3/6	19 / 32	59.38

5.5 Systemic Bottlenecks

The original CA-Ben study reported a 100% failure rate, where no models were able to surpass a 40% score in a subject. The two subjects of Taxation (I3) and Indirect Tax Laws (FN5) showed these systemic bottlenecks. Our system did not clear the Systemic Bottlenecks because the core reasoning abilities of Large Language Models show fundamental knowledge gaps that exist across all domains.

The results demonstrate that CA-ThinkFlow delivers better parameter efficiency and balanced reasoning performance across foundation and final levels while exceeding the performance of full-precision systems that require more resources.

6 Conclusion

The research studies how contextual accuracy and deep understanding work together in the assessment process for chartered accountancy tests. We introduced CA-ThinkFlow as a simplified Retrieval-Augmented Generation (RAG) framework which uses the built-in Chain-of-Thought reasoning of the 4-bit quantized 14B-DeepSeek-R1 model. The system provides high-quality contextual information to the language model which enables it to use its internal reasoning methods for query assessment and synthesis. The multi-level CA-Ben benchmarking of the framework shows that CA-ThinkFlow delivers outstanding parameter efficiency results. The system delivers noteworthy results for all three exam levels while matching the performance of much larger commercial models which operate at full precision and no quantization.

7 Future Work

The general performance of CA-ThinkFlow proves to be strong, but its performance needs critical improvement because it cannot overcome essential obstacles which exist in complex fields like Taxation (I3) and Indirect Tax Laws (FN5). Future work should focus on creating a more advanced retrieval pipeline which will enable us to handle both multi-chunk synthesis and dynamic context scaling for our multi-faceted tax query system. It should also investigate domain-specific supervised fine-tuning methods which will enable the base reasoning model to develop deeper regulatory knowledge through examination of Indian financial statutes before the retrieval process. Finally, expanding the system to ingest real-time, evolving tax regulations and incorporating multilingual support will further bridge the gap toward deployable, expert-level financial AI.

A Data Availability

This study uses official ICAI Study Material as context documents, processed through our pipeline for reproducibility.

- Foundation Course: `https://www.icai.org/post/foundation-course`
- Intermediate Course: `https://www.icai.org/post/intermediate-course`
- Final Course: `https://www.icai.org/post/final-course`

For testing data, we used the CA-Ben benchmark under a zero-shot setup, simulating a cold-start scenario [8].

- Benchmark repository: `https://github.com/thejatingupta7/LLMCA`

References

[1] Turing Institute. The impact of large language models in finance: Towards trustworthy adoption. *Alan Turing Institute Report*, 2024. [web:51].

[2] Institute of International Finance and Ernst & Young. 2024 iif-ey annual survey report on ai/ml use in financial services. Technical report, IIF, 2024. [web:60].

[3] Hans B. Christensen, Elizabeth Floyd, and Mark Maffett. Large language models and generative ai in finance. SSRN Electronic Journal, 2023. [web:49].

[4] Hongwei Mo and Shumiao Ouyang. (generative) ai in financial economics. *Journal of Chinese Economic and Business Studies*, 23(4):509–587, October 2025.

[5] Huaxia Li and Miklos A. Vasarhelyi. Applying large language models in accounting: A comparative analysis of different methodologies and off-the-shelf examples, November 2023.

[6] Jacob Morrison, Clara Na, Jared Fernandez, Tim Dettmers, Emma Strubell, and Jesse Dodge. Holistically evaluating the environmental impact of creating language models. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

[7] Ahmad Faiz, Sotaro Kaneda, Ruhan Wang, Rita Chukwunyere Osi, Prateek Sharma, Fan Chen, and Lei Jiang. LLMCarbon: Modeling the end-to-end carbon footprint of large language models. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

[8] J. Gupta, A. Sharma, S. Singhania, M. Adnan, S. Deo, A. I. Abidi, and K. Gupta. Large language models acing chartered accountancy, 2025.

[9] Fali Wang, Zhiwei Zhang, Xianren Zhang, Zongyu Wu, Tzu Hao Mo, Qiuhao Lu, Wanjing Wang, Rui Li, Junjie Xu, Xianfeng Tang, Qi He, Yao Ma, Ming Huang, and Suhang Wang. A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with llms, and trustworthiness. *ACM Transactions on Intelligent Systems and Technology*, 16(6), November 2025.

[10] Karmvir Singh Phogat, Sai Akhil Puranam, Sridhar Dasaratha, Chetan Harsha, and Shashishekar Ramakrishna. Fine-tuning smaller language models for question answering over financial documents. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10528–10548, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[11] OpenAI. o1 system card. Technical report, OpenAI, 2024. OpenAI Technical Report.

[12] OpenAI. Openai o3 and o4-mini system card. Technical report, OpenAI, 2025.

[13] Zartis Team. Deepseek-r1: The open-source ai challenger rewriting the rules of enterprise ai. `https://www.zartis.com/deepseek-r1-the-open-source-ai-challenger-rewriting-the-rules-of-enterprise-ai/`, 2025.

[14] DataCamp. Fine-tuning deepseek r1 (reasoning model). `https://www.datacamp.com/tutorial/fine-tuning-deepseek-r1-reasoning-model`, 2025.

[15] Shanghai AI Laboratory. Fin-r1: A large language model for financial reasoning, 2025.

[16] Yifan Zhou, Peng Li, et al. Fincot: Grounding chain-of-thought in expert financial blueprints. In *Proceedings of the 3rd Workshop on FinNLP*, 2025.

[17] The FinAI Team. Fino1: A financial reasoning model. `https://github.com/The-FinAI/Fino1`, 2025. GitHub Repository.

[18] A. Nawal and S. Kumar. Fin-rag: A rag system for financial documents, 2024.

[19] J. Muñiz Sánchez. Rag-based system for document information retrieval in financial compliance. `https://www.linkedin.com/pulse/rag-based-system-document-information-retrieval-muniz-sanchez-dk1bf`, 2024.

[20] Auxilio Bits. Rag architecture for financial compliance knowledge retrieval. `https://www.auxiliobits.com/blog/rag-architecture-for-domain-specific-knowledge-retrieval-in-financial-compliance/`, 2025.

[21] Docling Team. Docling: The document alchemist, 2024.

[22] Michele Besso et al. Docling technical report. Technical report, IBM Research, 8 2024.

[23] IBM Docling Team. docling-project/docling-models, 2025.

[24] Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. ConvFinQA: Exploring the chain of numerical reasoning in conversational finance question answering. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6279–6292, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[25] Yilun Zhao, Yitao Long, Hongjun Liu, Ryo Kamoi, Linyong Nan, Lyuhao Chen, Yixin Liu, Xiangru Tang, Rui Zhang, and Arman Cohan. DocMath-eval: Evaluating math reasoning capabilities of LLMs in understanding long and specialized documents. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16103–16120, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[26] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

[27] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. *IEEE Transactions on Big Data*, pages 1–17, 2025.

[28] DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025.