Financial Advisory System via Fine-Tuning and RAG

Md. Mahir Uddin

ID: 0122430022

United International University

Dhaka, Bangladesh

Necessary Links:

i) Fine-Tuning:

Fine-Tuning Notebook

Pre-Trained Model (Llama 3.2 3B-Instruct) Fine-Tuned Model

Dataset Used for Fine-Tuning

ii) RAG Application:

RAG Application Notebook

Embedding Model (Qwen 3 Embedding 4B)

1.1 Introduction: In recent years, large language models (LLMs) such as ChatGPT have demonstrated remarkable capabilities in understanding and generating natural language across a wide range of topics. However, these generalized models often fall short when it comes to domain-specific tasks that require specialized knowledge, such as performing fundamental analysis of companies or providing long-term investment advice. Investors and financial analysts rely on detailed evaluations of financial statements, market trends, and economic indicators to make informed decisions—tasks that generic LLMs are not fully equipped to handle.

The goal of this project is to bridge this gap by developing a domain-specific LLM tailored for finance, investing, business, and economics. This specialized model is capable of understanding company financials, analyzing business performance, and generating recommendations for long-term investment strategies. To achieve this, a pre-trained Llama 3.2 model with 3 billion parameters was fine-tuned on a curated dataset consisting of 20,000 instruction-based examples covering finance, investing, business statistics, and economic concepts. By focusing the model on domain-specific knowledge, the system can provide insights that general-purpose LLMs cannot reliably produce.

Beyond fine-tuning, this project implements a Retrieval-Augmented Generation (RAG) application to supply the model with the most relevant and up-to-date information. The RAG system retrieves context from multiple sources: preloaded financial and business books,

automatically downloaded annual reports for specific companies, and relevant web data. All retrieved information is processed into embeddings to efficiently extract context, which is then combined with the user's query to generate accurate, context-aware responses.

This approach allows the system to answer complex questions, perform fundamental analyses, and provide long-term investment guidance, all while leveraging both the fine-tuned LLM and dynamic retrieval of authoritative financial information. By integrating fine-tuning with RAG, this project demonstrates how specialized LLMs can be applied in high-stakes domains like finance, where accuracy and domain knowledge are critical.

1.2 Objectives:

- Develop a domain-specific LLM capable of performing fundamental analysis of companies and providing long-term investment insights.
- Integrate the LLM into a Retrieval-Augmented Generation (RAG) system to provide context-aware responses.
- Utilize multiple data sources, including financial books, annual reports, and web data, to ensure accurate and comprehensive analysis.
- Enable automatic extraction of company-specific information from user queries to support targeted financial advice.
- Demonstrate the effectiveness of combining fine-tuning with RAG for specialized financial applications.

2. Methodology:

2.1. Fine-Tuning the LLM

The first phase of the project involves fine-tuning a pre-trained Llama 3.2 model with 3 billion parameters to specialize in finance, investing, business, and economics. The QLoRA (Quantized Low-Rank Adaptation) method was employed to optimize the model efficiently while using limited computational resources.

2.1.1 Dataset Preparation:

For fine-tuning the Llama 3.2 model, three datasets were collected from public sources on Hugging Face and Kaggle:

1. Financial Q&A dataset – consisting of 7,000 rows of data.
2. FinQA dataset – consisting of 883 rows of data.
3. Investopedia Instruction Tuning dataset – a large dataset with 260,000 rows of data.

All datasets were originally provided in CSV format and contained features in instruction–input–output format or similar structures, with some additional metadata. To create a unified and domain-specific fine-tuning dataset, the following preprocessing steps were performed:

- The Investopedia dataset was trimmed to 12,000 rows to balance dataset size with computational efficiency.
- The trimmed Investopedia dataset was merged with the Financial Q&A and FinQA datasets.
- Irrelevant features and columns were removed, keeping only the instruction, input, and output information relevant for fine-tuning.
- The merged dataset was shuffled to ensure randomization and converted into JSONL format.
- Feature names were standardized to match the required format for the fine-tuning pipeline.

2.1.2 Model Selection:

For this project, the Llama 3.2 model with 3 billion parameters was selected as the base for fine-tuning. The model was loaded from the Kaggle models database. Llama 3.2 is a decoder-only transformer model, specifically designed for generating coherent and contextually relevant text.

The model consists of 28 layers, with each layer containing approximately 100 million parameters, giving it the capacity to capture complex patterns in language and domain-specific knowledge. The architecture of the model is presented in the figure below, which illustrates the transformer blocks, self-attention mechanisms, and feed-forward networks employed in each layer.

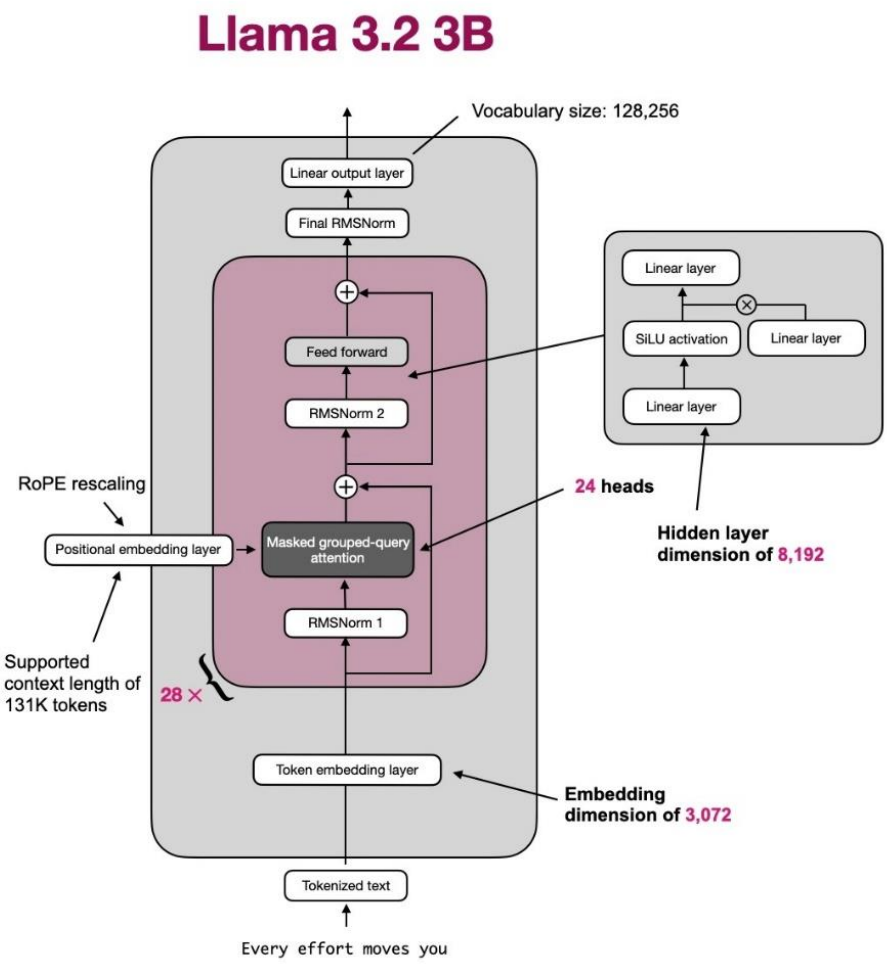

Fig. 1. Llama 3.2 Architecture

This architecture allows the model to effectively learn from domain-specific financial and business data while maintaining high expressive power for generating detailed and accurate responses in the context of fundamental analysis and investment advice.

2.1.3 Fine-Tuning:

To adapt the pre-trained Llama 3.2 (3B parameters) model to the financial domain, the QLoRA (Quantized Low-Rank Adaptation) method was used for fine-tuning. QLoRA is a parameter-efficient fine-tuning (PEFT) technique that allows large language models to be fine-tuned effectively on limited hardware by introducing low-rank adapters while keeping the base model weights frozen.

In this project, a LoRA rank of 4 was used, which provided a good balance between computational efficiency and learning capacity. The model was trained for 5 epochs using the prepared dataset of 20,000 instruction–input–output samples. The training process was implemented primarily using the Transformers library from Hugging Face.

To ensure training stability in a cloud-based environment, the model weights and adapter checkpoints were saved after every epoch, allowing recovery in case of session interruptions or hardware failures.

The fine-tuning configuration achieved the following parameter distribution:

- Trainable parameters: 6,078,464
- Total parameters: 3,218,828,288
- Trainable percentage: 0.1888%

2.1.4 Evaluation

The fine-tuned model's performance was evaluated primarily using the training loss metric. Over the course of 5 epochs, the loss value decreased from 0.53 to 0.2084, indicating steady and effective learning. The training loss curve (shown in the figure below) illustrates a smooth downward trend, reflecting stable convergence without overfitting or sudden fluctuations.

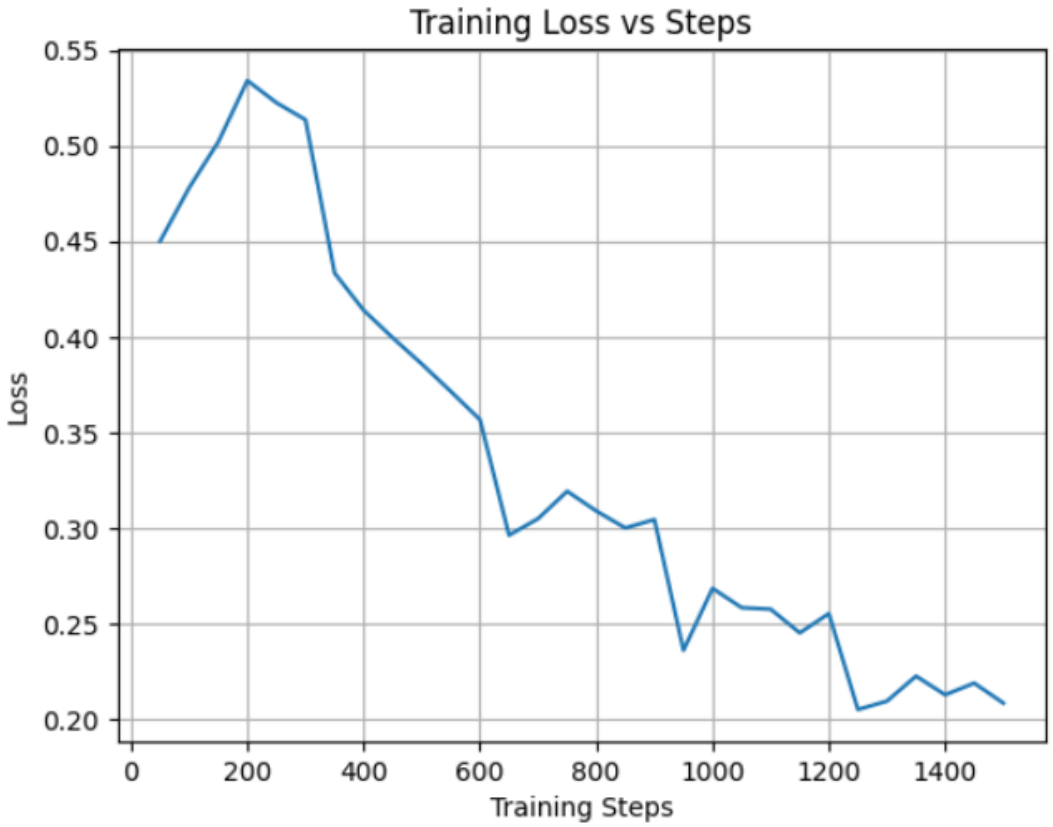

Fig. 2. Fine-Tuning Training Loss Curve

2.2 Retrieval-Augmented Generation (RAG) Application

The second phase integrates the fine-tuned model into a RAG system. RAG allows the LLM to access external information dynamically, enhancing its ability to provide context-aware, accurate responses for financial and investment queries.

2.2.1 Embedding Model Selection: The Qwen 3 Embedding (4B) model was selected for generating document embeddings, as it offers an optimal balance between performance and computational efficiency. It produces 2520-dimensional vectors by default and sits between the smaller 0.6B and larger 8B variants. The model was sourced from Kaggle and added to the input directory for seamless integration into the RAG pipeline.

2.2.2 Data Source: To ensure diverse, high-quality, and contextually relevant information retrieval, three different data acquisition approaches were employed for building the knowledge base in the RAG system. Each approach targets a distinct type of information — from static reference materials to dynamically fetched web data — enabling the model to provide both foundational insights and up-to-date company-specific information.

2.2.2.1. Preloaded PDF Books

A curated collection of books related to long-term investing, finance, economics, marketing, accounting, business strategy, business statistics, entrepreneurship, and consumer behavior was preloaded into the system. These include popular titles such as *The Intelligent Investor* and *One Up on Wall Street*, which serve as reliable sources of fundamental investment knowledge. The documents were processed using the PyPDFLoader class to extract text content. Each book was then split into manageable chunks to preserve semantic coherence before being converted into embeddings for efficient retrieval during query processing.

2.2.2.2. Web Search Results

For real-time and broader contextual information, the system integrates web search functionality using the DuckDuckGoSearch library. When a user submits a query, relevant web content is dynamically fetched and summarized to complement the static knowledge base. This allows the RAG model to provide updated and diverse perspectives on investment topics, latest market events, and company price sensitive news, ensuring the responses remain current and well-rounded.

2.2.2.3. Web PDFs (Company Annual Reports)

In certain cases, answering user queries requires specific financial or corporate information found in company annual reports. To handle this, a language model (LLM) first determines whether the user query necessitates accessing an annual report. If required, the LLM extracts the company name and relevant year, which are then used to search and download the corresponding annual report directly from the web. Once retrieved, the PDF is parsed, split, and embedded into the vector store, making its contents immediately available for retrieval and reference. This process ensures

that the system can deliver accurate, company-specific insights grounded in authentic financial documents.

A parallel processing chain is maintained for efficiency and modularity. Each of the three data sources independently generates its own set of split text documents, which are then combined into a unified corpus. This aggregated dataset is subsequently converted into embeddings and stored in the vector database for retrieval.

2.2.3 Retrieval: In the retrieval phase, the system identifies the most relevant information by computing the similarity scores between the embedding vector of the user's query and the stored document embeddings in the vector database. The documents with the highest similarity scores are retrieved as the most contextually relevant pieces of information. These retrieved chunks are then passed to the language model during the generation phase to ensure that responses are accurate, evidence-based, and contextually grounded in the retrieved knowledge.

2.2.4 Augmentation: In the augmentation phase, the retrieved contextual information is combined with the user's query to form a single enriched prompt that is passed to the language model. This step ensures that the model's response is not only based on its pretrained knowledge but also grounded in the most relevant, up-to-date, and factual information retrieved from external sources. A prompt template is used to structure this combination, where the retrieved context is inserted before the user's query in a clearly defined format. This allows the model to understand both the background context and the specific question being asked, leading to more coherent, accurate, and informative answers in the final output.

2.2.5 Generation: In the generation phase, the fine-tuned Llama 3.2 model receives the augmented prompt—which combines the user query with the retrieved contextual information—and generates a response.

By leveraging both its domain-specific knowledge from fine-tuning and the real-time, contextually relevant information from the RAG pipeline, the model produces answers that are accurate, detailed, and tailored to the user's query. This approach enables the system to provide fundamental analysis, investment recommendations, and insights that are grounded in authoritative sources, while maintaining natural and coherent language in the generated output.

3. Limitations:

- Hardware Constraints: Fine-tuning large models like Llama 3.2 is resource-intensive, and limited GPU memory restricted the model size and batch sizes used.
- GPU Scarcity: Cloud-based GPUs with session time limits required frequent checkpointing, which slowed down experimentation and training.
- Dataset Size: The domain-specific dataset was relatively small (20,000 examples), limiting the diversity and coverage of financial scenarios the model could learn.

- Suboptimal Model Choice: While Llama 3.2 3B was sufficient for experimentation, larger models could capture more complex financial patterns and reasoning.
- Retrieval Limitations: The RAG pipeline relies on embeddings and similarity search, which may not always capture nuanced context or interpret complex multi-step financial queries perfectly.
- Absence of Multimodal Inputs: The current system only processes textual data and cannot directly analyze tables, charts, or images from financial documents.
- No Web Interface: The application is not currently hosted on a website, limiting accessibility for end users.

4. Future Work

- Larger Models and Datasets: Fine-tuning larger LLMs (e.g., Llama 3.5 or 7B+) with more comprehensive datasets could improve accuracy and reasoning in complex financial tasks.
- Expanded Data Sources: Incorporating structured financial data, news feeds, and alternative data sources could enhance model insights.
- Improved RAG Techniques: Advanced retrieval strategies, including hybrid search (vector + keyword) and relevance re-ranking, could improve context precision.
- Multi-Modal Inputs: Future versions will integrate tables, charts, and images directly into the model input to allow deeper analysis of financial statements.
- Web Application: Hosting the system on a user-friendly website will make it accessible to investors and analysts.
- User Interaction Enhancements: Developing a user-friendly interface with query clarification, step-by-step analysis, and visualizations could make the system more practical for real-world users.

5. Conclusion: This project developed a domain-specific LLM capable of performing fundamental analysis and providing long-term investment insights by fine-tuning Llama 3.2 (3B) using the QLoRA method on a curated financial dataset. Integrating a RAG system allowed the model to retrieve relevant information from preloaded books, web content, and company annual reports, enabling it to generate accurate, evidence-based, and context-aware responses. Despite limitations such as hardware constraints, a small dataset, absence of multimodal inputs, and lack of a web interface, the project demonstrates the effectiveness of combining fine-tuning with retrieval-based augmentation for specialized financial tasks. Future work, including larger models, multi-modal inputs, improved retrieval strategies, and a hosted web application, will further enhance the system's capability, accuracy, and accessibility.