FinLoRA: Finetuning Quantized Financial Large Language Models Using Low-Rank Adaptation

Anonymous submission

Abstract

Finetuned large language models (LLMs) have shown remarkable performance in financial tasks, such as sentiment analysis and information retrieval. Due to privacy concerns, finetuning and deploying Financial LLMs (FinLLMs) locally are crucial for institutions. However, finetuning FinLLMs poses challenges including GPU memory constraints and long input sequences. In this paper, we employ quantized low-rank adaptation (QLoRA) to finetune FinLLMs, which leverage low-rank matrix decomposition and quantization techniques to significantly reduce computational requirements while maintaining high model performance. We also employ data and pipeline parallelism to enable local finetuning using cost-effective, widely accessible GPUs. Experiments on financial datasets demonstrate that our method achieves substantial improvements in accuracy, GPU memory usage, and time efficiency, underscoring the potential of low-rank methods for scalable and resource-efficient LLM finetuning.

Introduction

Large language models (LLMs) have demonstrated exceptional capabilities in various applications, such as finance (Liu et al. 2023a, 2024b), healthcare (Wang et al. 2024; Chen et al. 2024), scientific discovery (Lu et al. 2022b; Fu et al. 2024), etc. Finetuning using low-rank structures of these models to domain-specific datasets further enhances their performance and improves their applicability to specialized tasks. In the financial domain, finetuned LLMs demonstrate substantial potential for tasks such as sentiment analysis, named-entity recognition, and knowledge extraction from financial documents.

FinGPT (Liu et al. 2023a, 2024b,a) applied low-rank adaptation techniques for finetuning quantized LLMs in financial contexts, which displayed noticeable improvement over the base model, while having substantial memory reduction and training speedup. XBRL agent (Han et al. 2024) evaluated the potential of LLMs' capabilities on analyzing XBRL reports. The use of Retrieval-Augmented Generation (RAG) and tools-calling techniques on XBRL-related tasks and demonstrated significant improvement in task accuracy.

Due to sensitive data and regulatory constraints, finetuning and inference of LLMs within local environments remain critical requirements for financial institutions. Furthermore, the ability to create personalized and customized LLMs, finetuned for specific tasks, is essential for maximizing the utility of these models in financial applications.

Challenges Existing FinLLMs face the following challenges:

- **Inefficient finetuning**: Financial tasks are often complex and require precise adaptation to large domain-specific data, which can involve extensive parameter updates and prolonged training times.
- **Resource-constrained local devices:** Finetuning and deploying large language models on resource-constrained local devices, such as consumer GPUs, can be challenging due to their memory and computational limitations.

Building upon prior research, we demonstrate that state-of-the-art LLMs can be finetuned for diverse financial tasks locally and cost-effectively using widely accessible GPUs, achieving notable improvements over baseline models. Our main contributions can be summarized as follows:

1. We employ Quantized Low-Rank Adaptation (QLoRA) (Dettmers et al. 2023) to alleviate memory requirements and allow more efficient finetuning. Using the low-rank structure reduces the number of trainable parameters required for finetuning, and quantization compresses the model size, further limiting GPU memory consumption.
2. We employ distributed data parallelism (DDP) and pipeline parallelism to leverage multiple GPUs effectively. DDP distributes training data across GPUs to accelerate finetuning, while pipeline parallelism partitions the model at the layer level to optimize memory usage during inference. Together, these strategies enable more efficient fintuning and inference for FinLLMs.
3. We conduct extensive experiments on diverse financial datasets. Models finetuned with QLoRA exhibit up to a 48% average increase in accuracy compared to baseline models, which validates the effectiveness of low-rank adaptation and quantization in addressing the unique challenges of FinLLMs.

Finetuning LLMs with Quantized Low-rank Adaptation (QLoRA)

Quantized Low-rank Adaptation

Low-rank adaptation (LoRA) (Hu et al., 2021) is a parameter-efficient finetuning method that preserves the pre-

Table 1: The maximum GPU memory used during inference.

	GPU memory (GB)	
Quantization	Llama 3.1-8B	Llama 3.1-70B
16-bit	15.0	131.5
8-bit	8.6	68.5
4-bit	5.6	37.8

trained transformer model weights and introduces a smaller set of trainable weights, which are expressed using low-rank decomposition.

In LoRA, the update weights are assumed to follow the low-rank decompositions $\Delta \boldsymbol{W} = \boldsymbol{B}\boldsymbol{A}$, where $\boldsymbol{A} \in \mathbb{R}^{r\times n}$ and $\boldsymbol{B} \in \mathbb{R}^{n\times r}$ are trainable parameters, $\boldsymbol{W}_0 \in \mathbb{R}^{n\times n}$ is the pretrained weights. Note that n can be large, e.g., 4096 and the rank $r \ll n$, say $r = 4, 8$, or 16. As an example, setting $n = 4096$ and $r = 8$, $\boldsymbol{W}_0$ has about 16 million parameters, while $\boldsymbol{A}$ and $\boldsymbol{B}$ together have 65536 parameters, which is about 0.039% the size of $\boldsymbol{W}_0$.

During the finetuning stage, the forward pass is:

$$\boldsymbol{y} = \boldsymbol{W}_0\boldsymbol{x} + \Delta\boldsymbol{W}\boldsymbol{x} = \boldsymbol{W}_0\boldsymbol{x} + \boldsymbol{B}\boldsymbol{A}\boldsymbol{x},$$

where $\boldsymbol{W}_0$ denotes the pretrained weights.

During the inference stage, $\boldsymbol{A}$ and $\boldsymbol{B}$ will be added back to $\boldsymbol{W}_0$, resulting in matrix $\boldsymbol{W}$ with the size of $\boldsymbol{W}_0$, such that

$$\boldsymbol{W} = \boldsymbol{W}_0 + \boldsymbol{B}\boldsymbol{A},$$

$$\boldsymbol{y} = \boldsymbol{W}\boldsymbol{x}.$$

Therefore, it does not introduce additional costs to the inference process.

Quantized LoRA (QLoRA) (Dettmers et al. 2023) further reduces memory usage by using 8-bit or 4-bit quantization. During finetuning, all weights of the pre-trained model are quantized to 8-bit or 4-bit. Weights will be dynamically dequantized back to 16 bit when performing computation with the input sequence $\boldsymbol{x}$ and the adaptor matrix $\boldsymbol{A}$ and $\boldsymbol{B}$, which remain in 16-bit precision throughout the process.

High-Performance Optimizations on GPUs

Optimizing Finetuning Process

To accelerate the finetuning process and leverage the computational power of multiple GPUs, we employed Distributed Data Parallel (DDP), which distributes the training data across multiple GPUs. DDP launches one process per GPU, where each process gets its own copy of the model and optimizer. Each process then receives different inputs, and the gradients are computed and synchronized across all GPUs to accelerate training. DDP offers significant speedup when multiple GPUs are available (Li et al. 2020).

We also opted to use Brain Floating Point (BF16) during finetuning, BF16 offers range of values the same as FP32 and easy conversion to/from FP32. Studies showed that BF16 can achieve similar results as FP32 while having significant speedup and memory savings (Kalamkar et al. 2019).

We used 0/1 Adam optimizer (Lu et al. 2022a), a modified version of the Adam optimizer that linearizes each Adam step and allows utilizing 1-bit compression for faster convergence speed, while offering reduced data volume and higher training throughput.

Optimizing Inference Process

Inference on larger models like Llama 3.1 70B requires substantial GPU memory usage, particularly when using higher precision like 8-bit or 16-bit. We employ pipeline parallelism, where the model is partitioned at the layer level and distributed across multiple GPUs; each GPU process computes different micro-batches with different parts of the model concurrently (Liu et al. 2024a).

Table 1 illustrates GPU memory usage achieved through quantization and pipeline parallelism during inference with a batch size of 1. For Llama 3.1-8B, memory usage decreases by 43% with 8-bit quantization and 63% with 4-bit quantization compared to the original 16-bit. Similarly, for Llama 3.1-70B, the memory requirement reduces from 131.5 GB (16-bit, requiring 3 GPUs) to 68.5 GB (8-bit, 2 GPUs) and further to 37.8 GB (4-bit, 1 GPU). These reductions demonstrate the practical benefits of quantization in enabling resource-efficient inference for large-scale models.

Experiments

Experimental Setup

Hardware Configurations. The experiments were conducted on a server equipped with four 16-core AMD EPYC 7313 CPUs, 1 TB of RAM, and four NVIDIA RTX A6000 GPUs, each featuring 48 GB of dedicated GPU memory.

Financial Applications

For classification tasks, our study focuses on three financial language processing tasks: Sentiment Analysis (SA), Named Entity Recognition (NER), and news headline classification.

1. Sentiment Analysis (SA) entails analyzing financial text, such as news articles or tweets, to assign sentiment labels (e.g., positive, negative, or neutral).
2. Named Entity Recognition (NER) is designed to identify and classify critical entities within financial texts, including organizations, locations, and individuals.
3. News headline classification involves categorizing headlines according to predefined criteria or questions, facilitating the automated organization and analysis of financial news.

For Question-Answering (QA) tasks, we focus on eXtensible Business Reporting Language (XBRL) (Saeedi, Richards, and Smith 2007) data extraction. XBRL is a standardized format designed for the exchange of financial information. Although XBRL documents are based on structured XML (eXtensible Markup Language), their inherent complexity presents challenges that can be addressed using the capabilities of LLMs to extract key information, thereby facilitating financial reporting and analysis (Han et al. 2024). In this study, we aim to finetune LLMs to accurately extract both numerical and textual information from XBRL files.

Table 2: Datasets we used for finetuning and evaluation.

Dataset Name	Type	Train/Test Examples
FPB	Classification	1,200 / 3,600
FiQA SA	Classification	961 / 150
TFNS	Classification	9,540 / 2,390
NWGI	Classification	16,200 / 4,050
Headline	Classification	82,200 / 20,500
NER	Classification	13,500 / 3,500
XBRL Tags	QA	375 / 164
XBRL Values	QA	846 / 154

Base Models

We choose **Llama 3.1-8B Instruct** and **Llama 3.1-70B Instruct** (Dubey et al. 2024) models as base models.

Datasets

Sentiment Analysis The following datasets all consist of input texts and sentiment labels such as "neutral", "positive", or "negative".

- **Financial phrasebank (FPB)** (Malo et al. 2013) contains sentences extracted from financial news and reports. These sentences are annotated with sentiment labels. We manually created the train/test split.
- **Financial question-answering Sentiment Analysis (FiQA SA)** (Maia et al. 2018) is another sentiment analysis dataset with the same labels as FPB from microblog headlines and financial news.
- **Twitter financial news sentiment (TFNS)** (Rahman 2022) comprises annotated tweets related to financial news labeled with sentiment categories.
- **News with GPT instruction (NWGI)** (Liu et al. 2023b) comprises samples with seven labels ranging from “strong negative” to “strong positive”. We map the seven labels back to three for simplicity and consistancy with other SA dataset.

Headline classification The Headline dataset (Sinha and Khandait 2020) classifies headlines based on various questions into two classes: "yes" and "no".

Named entity recognition (NER) The NER dataset (Salinas Alvarado, Verspoor, and Baldwin 2015) annotates one entity per sentence, categorized into one of three classes: "location", "person", and "organization"

XBRL The XBRL dataset (Han et al. 2024) comprises questions and answers derived from XBRL filings from 2019 to 2023 for Dow Jones 30 companies. Each example includes a question, a text segment from an XBRL file containing the answer, and the ground truth generated using an XBRL file extraction library. From this dataset, we selected the following two tasks:

- **XBRL tag extraction**: This task involves extracting a specific XBRL tag from a large XBRL raw text segment given a natural language description of the tag.
- **XBRL value extraction**: This task focuses on extracting a numeric value from the raw XBRL text segment given a natural language description of the value.

To allow better instruction following for the base model, we use one-shot prompting by providing an example question and answer.

Implementation Details

Finetuning We employed distinct finetuning strategies based on the nature of the tasks:

- **Classification Tasks:** For classification tasks, including sentiment analysis, headline classification, and named entity recognition, single-task fine-tuning was employed.
- **XBRL Question Answering:** For XBRL question-answering tasks, including tag extraction and value extraction, multi-task fine-tuning was adopted.

All fine-tuning experiments utilized the 0/1 Adam optimizer (Lu et al. 2022a) with learning rate of 1e-4, LoRA alpha of 32, LoRA dropout of 0.1. We use both LoRA rank 4 with 4-bit quantization and rank 8 with 8-bit quantization for Llama 3.1 8B. We adjusted the batch size and number of training epochs based on the model size and task:

- **Classification Tasks:**
  - Llama 3.1 8B: Batch size of 16 with gradient accumulation step of 1; 4 epochs.
  - Llama 3.1 70B: Batch size of 4 with gradient accumulation step of 4; 4 epochs.
- **XBRL Tasks:**
  - Llama 3.1 8B: Batch size of 2 with gradient accumulation step of 2; 1 epoch.

Inference We use 8-bit quantized inference for all evaluations to ensure consistency.

Performance Metrics

We evaluate performance using the following metrics:

Accuracy Accuracy is the ratio of the number of correct answers to the total number of queries. An answer is considered correct if the ground truth answer is included in the generated response.

Weighted F1 score For classification tasks, we report the weighted F1 score, calculated as the weighted average of the F1 scores for each class, with weights proportional to the number of instances in each class

Finetuning and Inference Performance

- **Batch size**: The batch size per GPU during finetuning.
- **GPU memory usage**: The sum of the amount of GPU memory used for all GPUs during training.
- **GPU hours**: The product of total training time and number of GPUs used.
- **Adapter size**: The size of the LoRA adapter file.
- **Inference Speed**: The number of seconds to process an example.

Table 3: Accuracy on classification and XBRL extraction tasks.

Model	Classification Datasets						XBRL Extraction	
	FPB	FIQA	TFNS	NWGI	NER	Headline	Tags	Values
Llama-3.1-8B-Instruct (base)	0.6873	0.4655	0.6997	0.4658	0.4889	0.4534	0.7937	0.5526
Llama-3.1-70B-Instruct (base)	0.7450	0.4727	0.6842	0.7993	0.4628	0.7168	0.8902	0.8766
Llama-3.1-8B-Instruct-4bits-r4	**0.8630**	0.7309	**0.8827**	0.8095	0.9663	**0.8803**	**0.9500**	0.9605
Llama-3.1-8B-Instruct-8bits-r8	0.8284	**0.8036**	0.8405	**0.8396**	0.9805	0.8466	0.9437	**0.9736**
Llama-3.1-70B-Instruct-4bits-r4	-	-	-	-	**0.9888**	-	-	-

Table 4: Weighted F1 score on all classification tasks.

Model	Classification Datasets					
	FPB	FIQA	TFNS	NWGI	NER	Headline
Llama-3.1-8B-Instruct (Base)	0.6768	0.5571	0.6834	0.4117	0.5686	0.5576
Llama-3.1-70B-Instruct (Base)	0.7363	0.5645	0.6864	0.7993	0.4539	0.7294
Llama-3.1-8B-Instruct-4bits-r4	**0.8600**	0.7811	**0.8824**	0.8029	0.9664	**0.8864**
Llama-3.1-8B-Instruct-8bits-r8	0.8302	**0.8177**	0.8436	**0.8492**	0.9806	0.8520
Llama-3.1-70B-Instruct-4bits-r4	-	-	-	-	**0.9887**	-

Table 5: Finetuning and inference performance on one classification task (NER).

Model	Finetuning				Inference
	Batch size	GPU memory (GB)	GPU hours	Adapter size (MB)	Time per example (s)
Llama-3.1-8B-Instruct-4bits-r4	16 × 4	83.6	0.77 × 4	4.5	0.1
Llama-3.1-8B-Instruct-8bits-r8	16 × 4	96.7	0.90 × 4	9.0	0.1
Llama-3.1-70B-Instruct-4bits-r4	4 × 4	184.3	3.50 × 4	21.3	0.9

Table 6: Finetuning and inference performance on XBRL.

Model	Finetuning				Inference
	Batch size	GPU memory (GB)	GPU hours	Adapter size (MB)	Time per Example (s)
Llama-3.1-8B-Instruct-4bits-r4	2 × 4	139.2	0.44 × 4	4.5	1.9
Llama-3.1-8B-Instruct-8bits-r8	2 × 4	152.2	0.48 × 4	9.0	1.9

Results and Analysis

Tables 3 and 4 summarize the accuracy and weighted F1 scores under different finetuning configurations. Table 5 and 6 displays resource usage and inference performance for NER and XBRL QA. The finetuned Llama 3.1 8B demonstrates noticeable improvements in accuracy compared to its base model and even surpasses the results of the Llama 3.1 70B base model.

Notably, even with lower quantization (4-bit) and rank 4, the finetuned Llama 3.1 8B model achieves comparable performance to its 8-bit, rank 8 counterpart, while requiring less memory. Furthermore, the fine-tuned 70B model demonstrates practical usability with 4-bit quantization, showcasing the feasibility of deploying larger LLMs for complex financial tasks in resource-constrained environments.

While we utilized four GPUs to expedite finetuning, it is important to note that all finetuning are achievable with one GPU with 48GB memory, albeit with longer training times.

Conclusion and Future work

This study demonstrates the effectiveness of Quantized LoRA (QLoRA) for finetuning large language models (LLMs) on a range of financial tasks, including sentiment analysis, named entity recognition, news headline analysis, and XBRL data extraction. We finetuned both Llama 3.1 8B and 70B models, achieving up to 48% improvements in accuracy compared to the base models on average across all tasks. Notably, these performance gains can be achieved with only four GPUs and less than 20 hours of training times per task, making local finetuning and deployment of customized models a feasible option for financial institutions.

In future work, we plan to explore multi-task finetuning in classification tasks and expand our investigation of XBRL-related tasks, including formula calculations. This will enable FinLLMs to perform more complex analysis and reasoning tasks, further increasing their utility in the financial domain.

References

Chen, T.; Hao, N.; Van Rechem, C.; Chen, J.; and Fu, T. 2024. Uncertainty quantification and interpretability for clinical trial approval prediction. *Health Data Science*, 4: 0126.

Dettmers, T.; Pagnoni, A.; Holtzman, A.; and Zettlemoyer, L. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. arXiv:2305.14314.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; and et al. 2024. The Llama 3 Herd of Models. arXiv:2407.21783.

Fu, Y.; Lu, Y.; Wang, Y.; Zhang, B.; Zhang, Z.; Yu, G.; Liu, C.; Clarke, R.; Herrington, D. M.; and Wang, Y. 2024. DDN3. 0: Determining significant rewiring of biological network structure with differential dependency networks. *Bioinformatics*, btae376.

Han, S.; Kang, H.; Jin, B.; Liu, X.-Y.; and Yang, S. Y. 2024. XBRL Agent: Leveraging Large Language Models for Financial Report Analysis. In *Proceedings of the 5th ACM International Conference on AI in Finance*, ICAIF '24, 856–864. New York, NY, USA: Association for Computing Machinery. ISBN 9798400710810.

Kalamkar, D. D.; Mudigere, D.; Mellempudi, N.; Das, D.; Banerjee, K.; Avancha, S.; Vooturi, D. T.; Jammalamadaka, N.; Huang, J.; Yuen, H.; Yang, J.; Park, J.; Heinecke, A.; Georganas, E.; Srinivasan, S. M.; Kundu, A.; Smelyanskiy, M.; Kaul, B.; and Dubey, P. K. 2019. A Study of BFLOAT16 for Deep Learning Training. *ArXiv*, abs/1905.12322.

Li, S.; Zhao, Y.; Varma, R.; Salpekar, O.; Noordhuis, P.; Li, T.; Paszke, A.; Smith, J.; Vaughan, B.; Damania, P.; and Chintala, S. 2020. PyTorch distributed: experiences on accelerating data parallel training. *Proc. VLDB Endow.*, 13(12): 3005–3018.

Liu, X.-Y.; Wang, G.; Yang, H.; and Zha, . D. 2023a. Data-centric FinGPT: Democratizing Internet-scale data for financial large language models. In *Workshop on Instruction Tuning and Instruction Following, NeurIPS*.

Liu, X.-Y.; Wang, G.; Yang, H.; and Zha, D. 2023b. Data-Centric FinGPT: Democratizing Internet-scale Data for Financial Large Language Models. In *Workshop on Instruction Tuning and Instruction Following, NeurIPS*.

Liu, X.-Y.; Zhang, J.; Wang, G.; Tong, W.; and Walid, A. 2024a. Efficient Pretraining and Finetuning of Quantized LLMs with Low-Rank Structure . In *2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS)*, 300–311. Los Alamitos, CA, USA: IEEE Computer Society.

Liu, X.-Y.; Zhu, R.; Zha, D.; Gao, J.; Zhong, S.; White, M.; and Qiu, M. 2024b. Differentially Private Low-Rank Adaptation of Large Language Model Using Federated Learning. *ACM Transactions on Management Information Systems*.

Lu, Y.; Li, C.; Zhang, M.; Sa, C. D.; and He, Y. 2022a. Maximizing Communication Efficiency for Large-scale Training via 0/1 Adam. arXiv:2202.06009.

Lu, Y.; Wu, C.-T.; Parker, S. J.; Cheng, Z.; Saylor, G.; Van Eyk, J. E.; Yu, G.; Clarke, R.; Herrington, D. M.; and Wang, Y. 2022b. COT: an efficient and accurate method for detecting marker genes among many subtypes. *Bioinformatics Advances*, 2(1): vbac037.

Maia, M.; Handschuh, S.; Freitas, A.; Davis, B.; McDermott, R.; Zarrouk, M.; and Balahur, A. 2018. WWW'18 Open Challenge: Financial Opinion Mining and Question Answering. 1941–1942.

Malo, P.; Sinha, A.; Takala, P.; Korhonen, P.; and Wallenius, J. 2013. Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts. arXiv:1307.5336.

Rahman, M. A. 2022. Twitter financial news sentiment. http://precog.iiitd.edu.in/people/anupama.

Saeedi, A.; Richards, J.; and Smith, B. 2007. An Introduction to XBRL. In *British Accounting Association's Annual Conference*.

Salinas Alvarado, J. C.; Verspoor, K.; and Baldwin, T. 2015. Domain Adaption of Named Entity Recognition to Support Credit Risk Assessment. In Hachey, B.; and Webster, K., eds., *Proceedings of the Australasian Language Technology Association Workshop 2015*, 84–90. Parramatta, Australia.

Sinha, A.; and Khandait, T. 2020. Impact of News on the Commodity Market: Dataset and Results. arXiv:2009.04202.

Wang, Y.; Xu, Y.; Ma, Z.; Xu, H.; Du, B.; Gao, H.; and Wu, J. 2024. TWIN-GPT: Digital Twins for Clinical Trials via Large Language Model. *arXiv preprint arXiv:2404.01273*.