FinRAG-12B: A Production-Validated Recipe for Grounded Generation in Banking

Anonymous ACL submission

Abstract

Large language models are increasingly used across various fields. However, their adoption in regulated banking faces resistance due to demands for high accuracy, regulatory compliance, and traceable, grounded responses. We present a complete recipe for training and deploying grounded banking LLMs. First, we describe a data generation pipeline combining LLM-as-a-Judge filtering, citation annotation, and curriculum learning with only 143M tokens. The resulting 12B model achieves the highest answer quality among all systems tested and outperforms GPT-4.1 on citation grounding, with a modest citation tradeoff versus the untuned base. Second, we train calibrated refusal: 22% unanswerable examples yield a 12% "I don't know" rate, correcting the base model's unsafe 4.3% rate without GPT-4.1's over-refusal at 20.2%. Third, we provide end-to-end methodology from data curation to quantized serving, deployed at 40+ financial institutions with 7.1pp improvement in query resolution ($p < 0.001$). The model responds 3–5$\times$ faster at 20–50$\times$ lower cost than GPT-4.1.

1 Introduction

Large language models (LLMs) have transformed natural language processing across a wide range of domains including customer support, content generation, coding, and others. However, their adoption in regulated banking industry remains challenging due to their innate tendency to hallucinate, propensity toward agreeableness, and misalignment with domain-specific knowledge.

Furthermore, the nature of banking applications relies on ever-changing data, such as interest rates, account balances, and various policies and procedures. This dynamic environment requires LLMs to ground their responses in up-to-date information retrieved from internal databases or external sources.

Consider a customer asking their bank's virtual assistant about early mortgage payoff penalties. The system must retrieve current policies, interpret them, and generate a response that is grounded in those sources within seconds. If the policy is missing from the knowledge base, the model must say so rather than fabricate an answer as misleading information could lead to regulatory violations. Achieving this level of trustworthiness, speed, and accuracy requires careful model fine-tuning beyond off-the-shelf LLMs.

To address these gaps, we developed FinRAG-12B, a 12B-parameter LLM fine-tuned for retrieval-augmented generation (RAG) in banking. We describe a multi-source data curation pipeline combining LLM-as-a-Judge filtering, citation annotation, and two-stage curriculum learning that achieves 73% citation quality with only 143M tokens. Through systematic ablation, we show that adding 22% unanswerable examples in the training data enables calibrated "I don't know" responses, substantially reducing hallucinations while minimizing over-refusal on valid queries. We release the complete recipe for FinRAG12B—from data curation through quantization to serving—which has been validated in production use at 40+ financial institutions. We focus on three key contributions: (1) a high-quality data generation pipeline, (2) calibrated refusal through negative sampling ablation to mitigate hallucinations, and (3) an end-to-end methodology for production RAG systems in banking.

2 Related Work

The field of financial NLP field has evolved rapidly in recent years. BloombergGPT (Wu et al., 2023) trained 50B parameters on 346B tokens of Bloomberg data, achieving strong results on financial benchmarks including sentiment analysis, named entity recognition, and question answering.

However, the model was never publicly released, and the paper reported no production metrics such as latency, inference cost, or hallucination rates: metrics essential for deployment. FinGPT (Yang et al., 2023) democratized financial LLM development by showing that LoRA fine-tuning achieves competitive sentiment analysis results for under $300, but the work focused on classification tasks rather than generative RAG with citation requirements. FinBERT (Araci, 2019) established the paradigm of domain adaptation for financial sentiment analysis before the current generation of decoder-only LLMs, demonstrating that continued pretraining on domain text improves downstream task performance.

RAG has become a common approach for increasing trustworthiness in LLM responses (Lewis et al., 2020). This approach reduces hallucination rates by grounding responses in verified external knowledge sources that are provided as context to the response generation. However, RAG introduces challenges where the model may ignore retrieved context, cite irrelevant passages, or fabricate information entirely. Liu et al. (2024a) demonstrated that LLMs exhibit position bias in long contexts, preferentially attending to information at the beginning and end while "losing" content in the middle. We address this through random context placement during training with discrete trapezoidal distribution generated via a hierarchical uniform mixture.

Data quantity plays a critical role in LLM training. Phi-3 (Abdin et al., 2024) achieved strong performance with carefully filtered training data, showing that small models trained on high-quality corpora can match larger models trained on noisier data. Zhou et al. (2023) showed that just 1,000 handpicked instruction examples suffice for alignment, challenging assumptions about data scale. QuRating (Wettig et al., 2024) formalized quality assessment through pairwise comparisons, enabling systematic data selection. Xia et al. (2024) extended this to instruction tuning, selecting examples that maximally reduce validation loss. Ye et al. (2024) studied optimal data mixtures, finding that mixture ratios significantly impact downstream performance. These findings motivate our multi-stage approach where, rather than mixing all of the training data together in a single batch, we sequence training to maximize the benefit of high-quality proprietary examples.

For LLM training we leverage LoRA (Hu et al., 2022) and DoRA (Liu et al., 2024b) to learn low-rank weight updates, enabling parameter-efficient adaptation without modifying the full model. DoRA extends LoRA with directional decomposition for improved training stability. This approach reduces memory requirements from storing full gradients. Gunasekar et al. (2023) showed that carefully curated data outperforms larger noisy corpora, a finding we leverage by filtering training data through LLM-based quality scoring.

3 System Architecture

3.1 Base Model Selection

When approaching LLM training, the choice of base model is critical. We require a model that shows strong instruction-following capabilities, yet is efficient enough to meet the latency and cost requirements of production deployment. Gemma 3 12B-IT (Gemma Team et al., 2024) addresses all our requirements showing strong instruction-following, 128K context window, and permissive commercial licensing while maintaining efficiency.

3.2 Training Data Pipeline

"Garbage in, garbage out" is a popular proverb in machine learning that highlights the importance of training data quality. This is especially true when training LLMs for high-stakes domains like banking. While financial institutions have access to large volumes of data, much of it is noisy, outdated, and contains personally identifiable information (PII) that cannot be used for training. For this reason, we developed a multi-stage data curation pipeline that prioritizes quality. Our final corpus contains 143M tokens from a mix of open source and proprietary data processed through a multi-stage quality pipeline.

Source	Samples	License
RAG-v1 (Open)	43,581	Apache 2.0
SEC Reports (Synthetic QA)	16,773	Public
CommonCrawl (Financial)	20,499	CC0
Refusal Calibration (Proprietary)	17,795	Internal
Total	**98,648**	–

Table 1: Training data composition. Stage 1 uses open-source data (RAG-v1, CommonCrawl). Stage 2 adds proprietary banking conversations and synthetic QA from SEC filings.

3.2.1 RAG-v1

We filter 43,581 samples from the glaiveai/RAG-v1 dataset,[1] a synthetic collection of approximately 50,000 samples designed by Glaive AI to fine-tune large language models for retrieval-augmented generation tasks. Each entry includes a question, a list of context documents (1 to 5 chunks), and an answer often containing citations mapping facts to sources. We used JudgeLM (Zhu et al., 2023) to exclude low-quality responses with a score below 5. The final corpus contains 43,581 data points.

3.2.2 SEC Synthetic QA

While LLMs can generate plausible-looking training data, relying on synthetic text carries risks. Zhang et al. (2024) showed that training on model-generated text leads to regurgitative training, where successive generations degrade in diversity and factual grounding. Moreover, synthetic questions often fail to capture the distribution of real user queries, which tend to be shorter, more fragmented, and less interrogative than LLM-generated questions. To mitigate these issues, we designed a pipeline anchored in financial SEC filings rather than purely model-generated content. We extract 16,773 samples focusing on 10-K and 10-Q reports that contain rich financial information.

To align synthetic data with real-world conditions, we apply five steps: 1) split each filing into passage-length chunks, 2) generate questions at four difficulty levels (easy, medium, hard, expert), 3) rephrase questions via few-shot prompting to match natural query patterns, 4) generate answers from the original question and gold passage, and 5) construct hard negatives by injecting 3–7 distractor passages. Medium-difficulty questions best approximate real user queries and receive the largest sampling weight. Distractor positions follow a discrete trapezoidal distribution from a hierarchical uniform mixture.

This 5-step process is important to align the synthetic questions with real user queries. When comparing to a direct single-shot question and answer generation pipeline, we find that questions are verbose and follow an interrogative style, with a large portion starting with "what" and "how" (avg. 19.6 words, 93% "what" or "how"). Real client questions average 9.9 words, and 52% are fragments or keyword phrases like "min credit score for mortgage." To close this gap, we condition generation on attributes sampled from real question distributions: style (39% fragment, 20% how-do-I, 11% what-is, etc.), word count (log-normal, μ=2.1, σ=0.55), and formality (45% casual). We provide a comparison of standard evaluation measures (Potluru et al., 2024) in Table 2.

Metric	Single	Multi	Real
Avg. length (words)	19.55	**8.85**	9.91
Jaccard w/ real (↑)	0.098	**0.140**	—
Type entropy (↑)	1.418	**1.745**	2.281
Type JS div. (↓)	0.434	**0.041**	—
Coverage, cos. (↑)	0.464	**0.520**	—
Distinct-2 (↑)	0.295	**0.451**	0.721
Fin. term recall (↑)	0.902	**0.951**	—

Table 2: Single-shot vs. multi-step QA generation pipeline evaluated on lexical, question-type, semantic, diversity, and domain metrics. ↑ = higher is better, ↓ = lower is better.

3.2.3 CommonCrawl Financial Subset

Because user data is subject to strict privacy constraints, we adopt a hybrid approach to construct the training set without exposing personally identifiable information. First, we train a random forest classifier to identify banking-relevant content. Next, we extract real user questions that contain no PII and appear more than n times, preventing memorization of rare, potentially sensitive queries. Finally, we cross-reference these questions with the classifier to retrieve relevant passages. The resulting dataset of 20,499 samples is grounded in real-world usage while remaining compliant with privacy requirements.

3.2.4 Banking Refusal Calibration Data

In production RAG systems, the retrieval stage does not always surface passages that contain the answer to a user's query. When the retrieved context lacks the necessary information, a well-calibrated model should abstain rather than fabricate a plausible-sounding response. However, instruction-tuned LLMs exhibit a well-documented sycophancy bias (Sharma et al., 2024) where they tend to comply with the user's implied request for an answer even when no supporting evidence is available, a behavior that amplifies hallucination risks (Lewis et al., 2020). In regulated domains such as banking, a single confident but unsupported claim can expose the institution to compliance violations and erode customer trust.

To explicitly teach the model when not to answer, we construct a dedicated refusal calibration subset

[1] https://huggingface.co/datasets/glaiveai/RAG-v1

drawn from real banking conversations. Each example pairs a user question with a retrieved context that is topically related but does not contain sufficient information to formulate a correct response. The goal is to provide examples where the target output is an "I don't know"-style refusal. By exposing the model to a substantial proportion of negative examples during fine-tuning, we shift its decision boundary toward conservative behavior. The model learns to ground every claim in the retrieved context and to decline when that context is insufficient, rather than defaulting to parametric knowledge that may be outdated or incorrect.

3.3 Model Training

We adopt a two-stage curriculum inspired by domain-adaptive pretraining (Gururangan et al., 2020), where each stage uses a progressively more specialized data mixture and learning rate (Table 3). In **Stage 1 (Domain Adaptation)**, the model trains on 64,080 open-source samples from RAG-v1 and CommonCrawl financial datasets at a conservative learning rate of 1×10^{-6} with cosine decay, acquiring financial vocabulary and citation conventions from fully reproducible data. **Stage 2 (Task Specialization)** then continues from the Stage 1 checkpoint on 34,568 proprietary banking examples at 5×10^{-6} with linear decay, targeting production behaviors such as refusal calibration, institution-specific formatting, and domain terminology. Staging is critical as training on all data simultaneously yields worse performance (Section 5.3).

Stage	Data Source	Samples	LR
1	RAG-v1 + CommonCrawl	64,080	1×10^{-6}
2	SEC + Proprietary Banking	34,568	5×10^{-6}

Table 3: Curriculum learning stages. Stage 1 uses fully open data for domain adaptation; Stage 2 adds proprietary examples for task specialization.

3.4 Refusal Calibration

Training a model to refuse to answer when the retrieved context is insufficient requires careful control of the negative-example ratio. Too few negative samples and the model defaults to its sycophantic prior (Sharma et al., 2024), generating plausible but unsupported answers (high false-positive rate). Too many and the model refuses queries it could answer correctly (high false-negative rate). We sweep the negative ratio from 10% to 30% in 2-percentage-point increments and find that 22% strikes the best trade-off, minimizing false positives while preserving recall on answerable queries. Above 26%, the model over-refuses, collapsing recall.

3.5 Training Configuration

We fine-tune with LoRA (Hu et al., 2022) applied to all attention and MLP layers (r=64, α=256, dropout = 0.05). Optimization uses AdamW-8bit with a learning rate of 2×10^{-5}, per-device batch size of 4 and gradient accumulation over 4 steps (effective batch size 16), and a maximum sequence length of 65,536 tokens. We hold out 1% of the training data as a validation set and apply early stopping with a patience of 5 evaluation steps where training halts when validation loss ceases to improve and the best checkpoint is restored. Training completes in 1,402 steps across 360 GPU-hours on 8× RTX A6000.

3.6 Quantization

To meet production latency and memory constraints, we quantize the fine-tuned model to W4A16 with group size 128 (4-bit weights, 16-bit activations) using SmoothQuant (Xiao et al., 2023) activation smoothing. This reduces the model footprint from 24 GB to 8.4 GB (2.86× compression), enabling single-GPU deployment. Citation quality degrades only marginally, retaining over 99% of the full-precision performance.

4 Evaluation

4.1 Dataset

We evaluate on proprietary banking data and complement with public financial benchmarks to test generalization.

Proprietary Banking Test Set. We construct 258 banking RAG examples from three financial institutions representing distinct use cases: retail banking (account inquiries, loan information, branch hours) and commercial banking (treasury services, merchant processing). Half the examples contain irrelevant context where "I don't know" is the correct response, simulating real-world scenarios where retrieval returns off-topic documents. Each example contains a user question, 5 retrieved passages (1–2 relevant, 3–4 distractors), human-annotated ground truth with citation labels, and a binary answerable/unanswerable label.

Public Benchmarks. To assess whether these gains transfer beyond proprietary data, we addi-

tionally evaluate on two established financial QA benchmarks: **FinanceBench** (Islam et al., 2023), 150 questions over SEC filings requiring free-form answers grounded in real financial documents; and **FinQA** (Chen et al., 2021), a numerical reasoning benchmark over financial tables (500 sampled examples). FinanceBench tests factual retrieval and citation grounding, while FinQA stresses multi-step numerical reasoning, a capability not explicitly targeted by our training data. We use these benchmarks to augment our internal evaluation and to provide publicly comparable reference scores.

4.2 Metrics

We evaluate along three dimensions: answer quality, refusal calibration, and latency.

JudgeLM (Zhu et al., 2023) scores each response on a 1–10 scale using an LLM judge that assesses correctness, completeness, and coherence given the retrieved sources. Citation Quality (Es et al., 2024) (0–100) is a composite of faithfulness, source relevance, information synthesis, and source usage.

We report QA F1, which combines precision (fraction of generated answers that are correct) and recall (fraction of answerable queries that receive an answer). F1 captures the tension between hallucination risk (low precision) and over-caution (low recall). We additionally report the refusal rate and the fraction of abstentions that are true negatives.

4.3 Baselines

We compare against Gemma 3 12B-IT (no fine-tuning), GPT-4.1 (API). BloombergGPT weights are unavailable. FinGPT has no citation evaluation.

5 Results

5.1 Main Results

Model	Jud.LM	Cit. Q	QA F1	IDK%
Gemma 3 12B	5.70	**80.2**	**0.964**	4.3
GPT-4.1	5.72	70.8	0.900	20.2
FinRAG-12B	**6.21**	73.1	0.936	**12.0**

Table 4: Main results on 258 banking QA examples (3 institutions). JudgeLM: answer quality (1–10); Cit. Q: citation quality (0–100); QA F1: precision–recall on answerable queries; IDK%: abstention rate (↓ = hallucination risk, ↑ = over-caution).

FinRAG-12B achieves the highest answer quality (JudgeLM 6.21) among all evaluated models, surpassing both GPT-4.1 (+0.49) and base Gemma 3 (+0.51). On citation quality, FinRAG-12B (73.1) outperforms GPT-4.1 (70.8) by 2.3 points. The base model's strong citation score (80.2) validates Gemma 3 as a foundation for grounded generation. The fine-tuning trades a modest citation reduction for production capabilities including perfect citation formatting (100% recall and precision vs. 93.5%/99.0%).

The IDK column reveals refusal calibration where base Gemma refuses on only 4.3% of queries, risking hallucination on unanswerable inputs, while GPT-4.1 over-refuses at 20.2%, degrading recall to 0.831. FinRAG-12B's 12% strikes the balance between the two models.

5.2 Public Benchmark Results

Model	FinanceBench F1	FinQA F1
Gemma 3 12B (base)	0.249	0.025
GPT-4.1 (API)	0.238	0.030
FinRAG-12B	**0.284**	0.028

Table 5: Results on public financial benchmarks. FinRAG-12B achieves the highest FinanceBench F1 with a 97.3% citation rate. FinQA scores are low across all models, as numerical table reasoning was not a training objective.

On FinanceBench, FinRAG-12B outperforms both the base Gemma 3 model and GPT-4.1, achieving the highest F1 (0.284) with a 97.3% citation rate. FinQA scores are uniformly low across all models (0.025–0.030 F1), reflecting that multi-step numerical reasoning over financial tables was not an explicit training objective. This highlights a clear avenue for future work: incorporating table-reasoning data into the curriculum.

5.3 Curriculum Learning Ablation

Strategy	Jud.LM	QA F1	Cit. Q
Combined (all data)	3.28	0.706	51.2
Curriculum (staged)	**5.91**	**0.938**	**74.7**

Table 6: Curriculum vs. combined training on 258 banking QA examples. Mixing all data simultaneously collapses performance.

Training on all data simultaneously collapses: JudgeLM drops to 3.28 and QA F1 to 0.706 (Table 6). The combined model over-refuses at 46.5%, producing an IDK response for nearly half of all queries. The curriculum avoids this by establishing

general RAG capabilities in Stage 1 before specializing in Stage 2, preventing conflicting optimization targets between open-source and proprietary data.

5.4 Latency

Figure 1 compares time to first token (TTFT) and total time to completion (TTC) on a single RTX 6000 Ada GPU in a RAG setting. FinRAG-12B is 3–5× faster than GPT-4.1 on both metrics.

while maintaining the same TTFT.

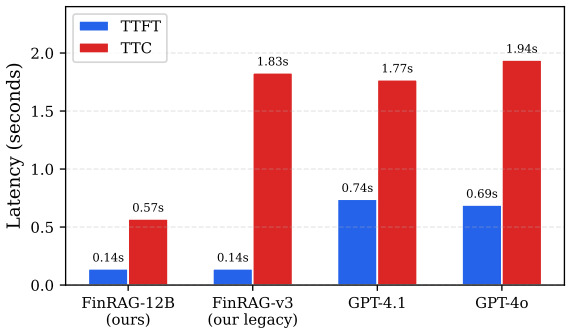

Figure 1: Inference latency comparison. FinRAG-12B achieves 0.14s TTFT and 0.57s TTC, outperforming both proprietary APIs and the prior production model.

5.5 Deployment

FinRAG-12B is available to 40+ financial institutions. Uptime: 99.7% (30-day rolling). P95 latency: under 20 seconds. One deployment serves 10+ million retail banking customers.

5.6 Production Impact

While the preceding sections evaluate on curated benchmarks, production deployment reveals whether improvements translate to real business outcomes. We analyze 3,297 randomly sampled user queries collected over seven months (May–December 2025) at a large US credit union serving 10+ million retail banking customers, comparing our legacy production model against FinRAG-12B to measure business impact. Both models serve the same RAG pipeline and knowledge base. We filter text-input queries (excluding UI-driven interactions such as source-document clicks) to isolate LLM performance.

Resolution rate, Table 7, defines the fraction of queries the model answers successfully improves by 7.1 percentage points (χ^2=24.4, p<0.001, Cramér's V=0.09), meaning 7 additional queries per 100 are resolved without human escalation. The

	Old	FinRAG	Δ
N (queries)	1,044	2,253	—
Resolution	77.4%	**84.5%**	+7.1***
Unresolved	20.7%	**13.7%**	−7.0***
Satisfaction	59.5%	**62.9%**	+3.4
resolved	65.0%	textbf66.7%	+1.7
unresolved	23.7%	textbf25.9%	+2.1

Table 7: Production metrics (3,297 queries, 7 months). Deltas in pp. ***p<0.001 (χ^2). Satisfaction: p=0.26, 95% CI [−2.5, +9.3] pp.

unresolved rate drops by the same margin, confirming that the model converts previously unanswerable queries into successful responses rather than shifting failures between categories.

Overall user satisfaction (binary thumbs-up/thumbs-down) increases from 59.5% to 62.9%, though this difference does not reach statistical significance (χ^2=1.3, p=0.26; 95% bootstrap CI [−2.5, +9.3] pp). Decomposing by outcome reveals why: per-query satisfaction is nearly identical for both resolved queries (65.0% vs. 66.7%) and unresolved queries (23.7% vs. 25.9%). The observed overall gain is driven by a compositional shift where more queries fall into the higher-satisfaction resolved category rather than by improving individual response quality. This finding suggests that resolution rate, not per-response polish, is the primary lever for user satisfaction in production RAG systems.

6 Conclusion

We present FinRAG-12B, a citation-grounded banking RAG model. First, we show that a data pipeline with LLM-as-a-Judge filtering and two-stage curriculum learning produces the highest answer quality (JudgeLM 6.21) outperforming GPT-4.1 on citation quality by 2.3 points. Second, mixing 22% unanswerable examples into training yields calibrated refusal behavior: FinRAG-12B abstains on 12% of queries, between the under-cautious base model (4.3%) and the over-cautious GPT-4.1 (20.2%). Third, the complete recipe transfers to production. FinRAG-12B improves query resolution by 7.1 percentage points (p<0.001) while running 3–5× faster and 20–50× more cheaply than commercial APIs. These results suggest that data quality and training methodology matter more than scale for grounded generation in regulated domains.

Limitations

Our evaluation covers banking RAG from three financial institutions (258 examples). Results may not transfer to trading, insurance, investment advisory, or other financial domains. The evaluation set skews toward common retail banking queries; rare edge cases remain untested.

The proprietary Stage 2 data cannot be released. However, Stage 1 training uses entirely open-source data (RAG-v1, CommonCrawl), and we provide all hyperparameters. Researchers can replicate our curriculum methodology with their own proprietary datasets.

We compare against API model (GPT-4.1) with different deployment constraints. Direct comparison to other self-hosted 12B models (Llama 3.1 8B, Mistral 7B) would strengthen the evaluation.

Our evaluation counts explicit "I don't know" and variations of responses but may miss hedged language ("I wish I could help, but...") or partial uncertainty expressions.

Ethics Statement

Our model operates in regulated financial services. All proprietary training data was anonymized to remove personally identifiable information before use. The model refuses queries outside its knowledge domain rather than hallucinating, which banking compliance requires. LLMs can perpetuate training data biases; we monitor response quality across customer demographics.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, and 1 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borber, Curtis Langlotz, and William Yang Wang. 2021. Finqa: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7199–7210.

Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. RAGAs: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 150–158, St. Julians, Malta. Association for Computational Linguistics.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, and 1 others. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Suriya Gunasekar, Yi Zhang, Jyoti Anber, Girish Menber, Xin Chen, Adam Santoro, and 1 others. 2023. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. 2023. Financebench: A new benchmark for financial question answering. *CoRR*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.

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

Shih-Yang Liu, Chien-Yi Wang, Hongxia Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024b. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2402.09353*.

Vamsi K Potluru, Daniel Borrajo, Andrea Coletta, Niccolò Dalmasso, Yousef El-Laham, Elizabeth Fons, Mohsen Ghassemi, Sriram Gopalakrishnan, Vikesh Gosai, Eleonora Kreacic, and 1 others. 2024. Synthetic data applications in finance. *CoRR*.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R

Johnston, and 1 others. 2024. Towards understanding sycophancy in language models. *arXiv preprint arXiv:2310.13548*.

Alexander Wettig, Tianyu Gao, Zexuan Zhong, and Danqi Chen. 2024. Qurating: Selecting high-quality data for training language models. *arXiv preprint arXiv:2402.09739*.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arber, Mike Lewis, and Danqi Chen. 2024. Less: Selecting influential data for targeted instruction tuning. In *International Conference on Machine Learning*.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. 2023. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, pages 38087–38099.

Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. 2023. Fingpt: Open-source financial large language models. In *IJCAI 2023 Workshop on Financial Technology and Natural Language Processing*.

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Yunhua Zhou, Jun Zhan, and Xipeng Qiu. 2024. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. *arXiv preprint arXiv:2403.16952*.

Jinghui Zhang, Dandan Qiao, Mochen Yang, and Qiang Wei. 2024. Regurgitative training: The value of real data in training large language models. *Available at SSRN 4870843*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment. In *Advances in Neural Information Processing Systems*, volume 36.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2023. Judgelm: Fine-tuned large language models are scalable judges. *arXiv preprint arXiv:2310.17631*.

A Training Hyperparameters

Parameter	Value
Base Model	Gemma 3 12B-IT
LoRA Rank (r)	64
LoRA Alpha (α)	256
LoRA Dropout	0.05
Target Modules	q,k,v,o,gate,up,down
Learning Rate	2×10^{-5}
Optimizer	AdamW-8bit
Batch Size	4
Gradient Accumulation	4
Effective Batch Size	16
Max Sequence Length	65,536
Training Steps	1,402
Final Loss	1.871

Table 8: Training hyperparameters.

B Quantization Details

W4A16-G128 quantization configuration:

- **Weight Quantization**: 4-bit, group size 128
- **Activation Precision**: 16-bit (FP16)
- **Preprocessing**: SmoothQuant, migration factor 0.5
- **Calibration**: 512 samples from training distribution

Size reduction: 24GB $\rightarrow$ 8.4GB (2.86$\times$ compression). Citation quality degrades marginally, retaining over 99% of full-precision performance.