Demystifying Domain-adaptive Post-training for Financial LLMs

Anonymous ACL submission

Abstract

Domain-adaptive post-training of large language models (LLMs) has emerged as a promising approach for specialized domains such as medicine and finance. However, significant challenges remain in identifying optimal adaptation criteria and training strategies across varying data and model configurations. To address these challenges, we introduce FINDAP, a systematic and fine-grained investigation into domain-adaptive post-training of LLMs for the finance domain. Our approach consists of four key components: *FinCap*, which defines the core capabilities required for the target domain; *FinRec*, an effective training recipe that jointly optimizes continual pre-training and instruction-following, along with a novel preference data distillation method leveraging process signals from a generative reward model; *FinTrain*, a curated set of training datasets supporting FinRec; and *FinEval*, a comprehensive evaluation suite aligned with FinCap. The resulting model, Llama-Fin, achieves state-of-the-art performance across a wide range of financial tasks. Our analysis also highlights how each post-training stage contributes to distinct capabilities, uncovering specific challenges and effective solutions, providing valuable insights for domain adaptation of LLMs.

1 Introduction

While LLMs have demonstrated strong generalization across a variety of tasks, they often struggle to perform well in specialized domains such as finance and law. Consequently, *domain-adaptive post-training* of LLMs has garnered significant attention recently (Colombo et al., 2024a; Xie et al., 2024b). In the earlier days of language models, *continual pre-training* (CPT) was the dominant strategy. This involved further training a pre-trained model on domain-specific plain text and then fine-tuning it for individual tasks (Gururangan et al., 2020; Ke et al., 2023). With LLMs, the post-training focus has shifted to zero- and few-shot task generalization through methods such as *instruction-tunning (IT)* (aka. supervised fine-tuning or SFT) and *preference alignment (PA)*. While prompt engineering of powerful general LLMs with zero- or few-shot examples has emerged as a convenient approach to adapting them to new tasks, to get the most optimal performance on a target domain, recent methods explore fine-tuning model wights to make them domain experts (Chen et al., 2023b; Li et al., 2023; Colombo et al., 2024b).

Building on this trend, this work focuses on adapting LLMs to specific domains *through parameter training*. It complements semi-parametric methods that leverage *external* knowledge, such as retrieval-augmented generation (Lewis et al., 2020; Ke et al., 2024). Our focus is also different from general post-training, as the goal is not to develop another general-purpose LLM but to create specialized, expert-level LLMs tailored to a specific domain. By focusing on a specific domain, we develop models that are not only more compact in size but also deliver significantly more accurate and contextually relevant responses compared to general-purpose LLMs. Their smaller size enhances efficiency, optimizing both computational resource usage and training time.

Despite the potential of domain-specific LLMs, there is still no systematic study on what makes a good domain-specific LLM. In this work, we consider *finance* as the domain of interest and aim to address the following research questions:

> Given a strong general-purpose LLM (*e.g.*, Llama3-8b-inst), how to effectively adapt it to a target domain (*e.g.*, finance) by post-training? What criteria are desirable for successful adaptation? What are effective training recipes with respect to data and model?

Prior studies (Bhatia et al., 2024; Xie et al., 2024a) typically adopt a simplified and informal framework (see §2) in that they evaluate only on a

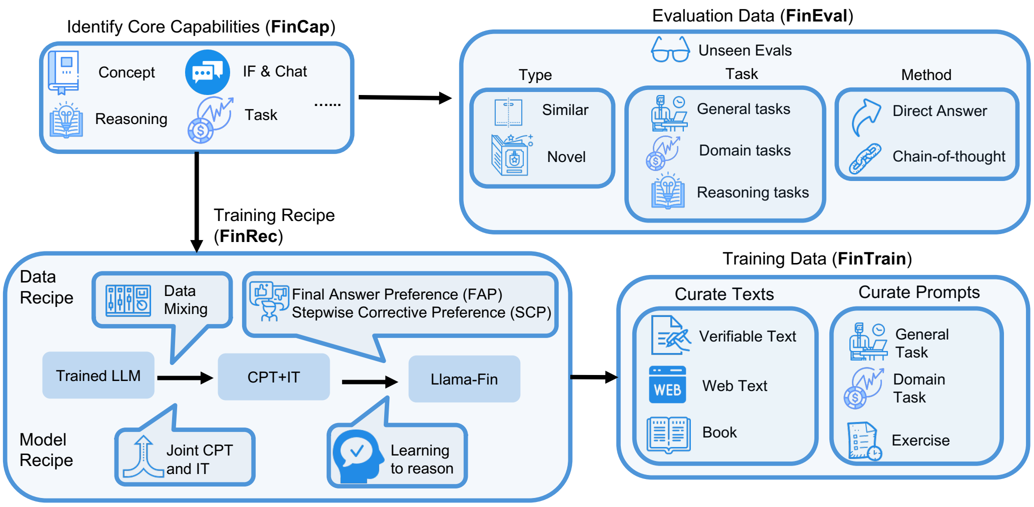

Figure 1: An overview of our finance-specific post-training framework, FinDAP. It comprises four key components: (1) **FinCap**, the core expected capabilities, including concepts, reasoning, instruction-following and tasks; (2) **FinRec**, encompassing both data and model strategies to guide domain-adaptive post-training; (3) **FinTrain**, which curates training texts and prompts based on the data recipe; and (4) **FinEval**, a comprehensive evaluation framework designed to assess performance on unseen tasks, categorized into similar and novel, general and domain-specific, and reasoning tasks, using both direct-answer and chain-of-thought (CoT) evaluation methods.

set of domain-specific end tasks such as sentiment analysis and NER, and they simply follow standard post-training stages (CPT, IT and/or PA) without considering their impact or optimizing their recipe for domain-adaptive post-training. This simplified approach can misalign with our broader expectations for a domain-expert LLM. A domain-expert LLM should not only excel at such end tasks but also achieve broader capabilities, such as follow task instructions effectively and reason in a way that aligns with domain-specific knowledge, while retaining general capabilities.

We argue that domain-adaptive post-training poses unique challenges compared to pre-training or general post-training. There are multiple factors to be considered: **(1)** For a particular target domain, it is essential to establish the **desirable capabilities** that a domain-expert LLM should possess, as these capabilities serve as a guiding framework for the entire adaptation process; **(2)** The training recipe should be tailored specifically to adapt an already trained LLM (e.g., Llama3-inst) through post-training. This differs from training a model from scratch or from a base pre-trained checkpoint, as it requires careful consideration of **catastrophic forgetting** and **knowledge transfer** from the original LLM, which already possesses strong general knowledge and instruction following capabilities. Each of the standard CPT, IT, and PA stages have different impacts and trade-offs with respect to knowledge forgetting and transfer, as do the in-domain, general-domain datasets, and the mixture of them. Moreover, it should also be designed to support the desired capabilities. For example, improving reasoning capability might require more dense supervision than the final answer level correctness score. **(3)** The desired **quantity and quality of training datasets** should be carefully balanced: high-quality general-domain data is required to mitigate forgetting, while diverse data and supervision signals are necessary to learn domain knowledge. **(4)** Finally, the **evaluation methods** should align with the desired capabilities. Different evaluation techniques may be required for certain capabilities; for example, chain-of-thought (CoT) (Wei et al., 2023) reasoning is often necessary to effectively evaluate reasoning tasks.

In this work, we introduce FinDAP (Figure 1), a novel finance-specific framework designed to incorporate all these factors in domain-specific post-training. To our knowledge, none of the prior studies consider all of them to provide a principled guidance on domain-adaptive post-training. FinDAP integrates four key components: (1) **FinCap**, a set of core capabilities required for the domain expert LLM, derived from a systematic review of prior literature and input from domain experts in finance. These include domain concepts, tasks, instruction following and reasoning; (2) **FinRec**, a training recipe that **jointly performs CPT and IT**, and subsequently conducts PA, balancing trade-offs across these stages to mitigate forgetting and improve task generalization. It also proposes to use **mixture of in-domain and general domain data** in the data recipe, alongside a novel preference alignment method for improving reasoning capability that constructs data using the preference signal in reasoning steps, **Stepwise Corrective Preference (SCP)**, and final answer, **Final Answer Preference (FAP)**; (3) **FinTrain**, a curated set of training datasets implementing FinRec, which carefully balances quality and diversity; and (4) **FinEval**, a comprehensive evaluation framework covering a wide range of tasks, including reasoning tasks assessed through CoT.[1]

We apply FINDAP on the instruction-tuned Llama3-8b-instruct (LLaMA, 2024). Our best performing recipe yields **Llama-Fin** that outperforms all considered baselines, including large open models at the 70B scale and proprietary models like GPT-4o, on tasks that are similar (yet unseen) to the training data. Even on novel tasks that were never encountered in training, Llama-Fin remains competitive and consistently outperforms its base model across all identified capabilities. In summary, our key contributions are:

- **Comprehensive guidance** for finance-specific post-training, including identification of capabilities, evaluation, data and model recipe design.
- **Systematic exploration** on each stage of post-training, with an emphasis on the goals, challenges and effective approaches.
- **Novel preference alignment approach** that constructs preference data using on-policy trajectories guided by outcome and process signals.
- **New State-of-the-art financial LLM** (Llama-Fin) at the 8b parameter scale based on the above.

2 Related Work

Finance LLMs Table 1 summarizes popular finance-specific LLMs developed through domain-adaptive post-training. AdaptLLM (Cheng et al., 2024) focuses on CPT and constructs heuristic QA tasks from raw text, but it considers only five financial end tasks. PIXIU (Xie et al., 2023) focus on instruction-following by creating a financial instruction-tuning dataset from diverse open financial tasks and designing a benchmark with nine end tasks for evaluation. FinLLM (Xie et al., 2024a) extends post-training across multiple stages, first performing CPT, then IT, and incorporating multi-modal capabilities via IT. It includes some general-domain data (e.g., FineWeb (Penedo et al., 2024)) but does not explore its impact systematically. Following this line, FinTral (Bhatia et al., 2024) is the only open FinLLM to include PA, where preference labels were given by GPT-4 on the final outcome, considering only coarse-grained signals. It also introduces multi-modality via IT and integrates tool use and retrieval in PA training. Additionally, Palmyra-Fin (Writer, 2024), a recent state-of-the-art FinLLM, reports high performance on finance tasks, particularly CFA exams[2], but its training recipe remains undisclosed.

Comparing to FINDAP, none of these models explicitly identify *desirable capabilities* as we do with *FinCap*, nor do they systematically explore trade-offs between CPT, IT and PA to develop a more effective training recipe. They also do not incorporate fine-grained process signals in PA to improve reasoning, as we do in *FinRec*. Additionally, their evaluations lack the broader range of tasks, methods, and similarities, including reasoning tasks and CoT evaluations, that we adopt in *FinEval*. Finally, unlike Palmyra-Fin, Llama-Fin is fully open-source, ensuring complete transparency in its training recipe, datasets, and evaluation methods, while achieving SoTA in its size category.

PA for reasoning We explore training-time approaches for improving reasoning (Jiao et al., 2024; DeepSeek-AI et al., 2025). These methods first collect trajectories and then train the LLM with the collected trajectories. This helps the model reason more accurately and faster during inference. To collect reasoning trajectories, there are two main approaches. The first is *search-based* (Setlur et al., 2024; Snell et al., 2024), where a trained Reward Model (RM) is used to guide a search method (e.g., Best-of-N, Beam Search) to identify the best reasoning path. The second is *revision-based* (Bai et al., 2022; Du et al., 2023; Madaan et al., 2023; Saunders et al., 2022), which attempts to improve the generation distribution through multi-round interactions, often by leveraging feedback from itself or another strong LLM to refine the input prompt. In practice, revision-based methods have shown mixed results and have not yet been well established as reliable for achieving improvements (Huang et al., 2024a). In contrast, search-based methods have been shown to be more effective. In FINDAP, we propose a novel training-time method that leverages a search-based trajectory collection

[1]We will open-source the data, checkpoint, code, leaderboard for all components upon acceptance.

[2]https://www.cfainstitute.org/programs/cfa-program/exam

Finance LLM	Capabilities	Recipe: Model Recipe	Recipe: Data Recipe	Evaluation
AdaptLLM	Concept	CPT	**CPT**: Financial text + heuristic QAs constructed from the text	Financial tasks + Direct answer
PIXIU	Task	IT	**IT**: Financial tasks	Financial tasks + Direct answer
FinLLM	Concept, Task	CPT → IT	**CPT**: Financial text + Fineweb; **IT**: Filtered Financial tasks	Financial tasks + Direct answer
FinTral	Concept, Task	CPT → IT → PA	**CPT**: Financial text; **IT**: Financial tasks; **PA**: Outcome signal only	Financial tasks + Direct answer
Palmyra-Fin	SoTA public checkpoint, but recipe is not disclosed			
Llama-Fin	Concept, IF/Chat, Task, Reasoning	CPT+IT → PA	**CPT**: Financial + General text. **IT**: Financial + General tasks **PA**: A novel PA that leverages outcome and process signals	General + Financial tasks; Similar + Novel tasks Knowledge Recall + Reasoning tasks Direct answer + CoT

Table 1: Comparison between Llama-Fin with other finance LLMs.

approach, incorporating both outcome and process rewards from a Generative RM (GenRM).

3 FinDAP Framework

In FinDAP, we first identify *four* desired capabilities for a finance-expert LLM (**FinCap**, §3.1). We then develop the training recipe **FinRec**, which includes both the *model recipe* that performs CPT and IT jointly followed by PA, and the *data recipe*, which examines the impact of in-domain, general-domain, and mixed-domain data while introducing a novel data construction approach for PA (§3.2). We then introduce **FinTrain** (§3.3), a set of carefully curated training datasets designed to mitigate forgetting while effectively learning domain-specific knowledge. Finally, we propose an evaluation framework **FinEval** (§3.4), which considers a diverse set of tasks, ranging from familiar to novel and from general to domain-specific, while also evaluating both direct-answer and CoT methods.

3.1 Core Capabilities (FinCap)

We began by conducting a comprehensive survey of existing work and consulting two financial domain experts: a banking industry advisor and a financial industry product manager. From this, we identified four key fundamental capabilities essential for a finance LLM: understanding domain-specific concepts to process financial language accurately, performing domain-specific tasks to solve real-world problems, reasoning effectively to analyze complex financial data, and following instructions to interact naturally in practical applications. These capabilities are deeply interconnected: reasoning depends on conceptual knowledge, while instruction-following ensures effective communication.

• **Domain specific concepts.** A domain typically includes its own specific concepts. For example, 'bond' in finance refers to a loan agreement between an investor and a borrower. Adapting the LLM to domain-specific concepts is crucial, as these concepts form the fundamental building blocks of domain knowledge. However, this adaptation should not come at the cost of losing knowledge about general concepts, which are essential for both domain-specific and general tasks.

• **Domain specific tasks.** While many NLP tasks, such as NER or sentiment analysis, are shared across different domains, a domain typically has its own tasks. For example, stock movement detection is primarily found in finance. Adapting LLMs to these domain-specific tasks is important, as it demonstrates how they can leverage domain-specific concepts to solve tailored tasks effectively.

• **Reasoning.** For complex tasks, reasoning with concepts is a highly desired capability in LLMs. For example, in finance, the LLM is often required to analyze a company's financial report, involving extensive reasoning, particularly mathematical reasoning, to compute key financial concepts such as market rate or earnings per share.

• **Instruction-Following (IF) and chat.** This is a core capability for both general and domain-specific LLMs, as tasks are often presented in the form of instruction following or conversation.

3.2 FinDAP Training Recipe (FinRec)

As shown in Figure 1, FinRec consists of two recipes: the *model recipe*, which focuses on the training stages and losses, and the *data recipe*, which focuses on constructing training data.

3.2.1 Model Recipe

Previous studies often de facto treat domain-adaptive post-training as a sequential process involving, or partially involving, CPT, IT, and PA. However, our experiments with LLaMA3-8B-Inst show key trade-offs among these stages (App. B). While CPT is effective at introducing domain concepts, it often leads to *significant forgetting* of general concepts and instruction-following capabilities. In contrast, IT strengthens instruction-following capabilities and introduces domain-specific tasks with minimal forgetting. IT alone however struggles with *task generalization*. PA is effective for learning reasoning but depends heavily on high-quality preference data, which can be difficult to synthesize. To address these limitations, we propose a *joint CPT+IT approach*, resulting in CPT+IT checkpoint. Subsequently, PA is performed with a novel trajectory collection method that provides fine-grained supervision signals.

Joint continual pre-training and instruction-tuning (CPT + IT). In this stage, the goal is to learn domain-specific knowledge while maintaining general capabilities, such as instruction-following. It is well known that CPT can adapt the LLM to learn domain-specific concepts while IT can help learn the domain-specific and instruction-following tasks (Ke et al., 2022; Wei et al., 2022). Typically CPT involves next-token prediction *without masking* any context tokens, and IT involves next-token prediction with *instructions masked out*; thus training them sequentially from an instruction-tuned LLM naturally leads to forgetting general capabilities, including instruction-following. Intuitively, if the loss function incorporates both CPT and IT, forgetting can be largely mitigated (Scialom et al., 2022).[3] To achieve this, we mix CPT and IT data, effectively performing joint optimization, as the only difference between the two is whether the instruction is masked. This approach also facilitates knowledge transfer, as CPT helps the model learn domain knowledge, which can be leveraged by IT training. More importantly, since concepts learned from CPT are often inherently more generalizable due to the shared nature of concepts across tasks, jointly training CPT and IT can improve generalization without require exposure to a diverse range of tasks , which is often impractical in certain domains, particularly long-tail ones. Since CPT datasets are typically much larger than IT datasets, we downsample CPT data to match the size of IT data, allowing for effective joint training.

Improving reasoning with preference alignment. CPT+IT improves capabilities such as in general and domain-specific concepts, tasks and IF/Chat. However, we find that the resulting model lacks in its reasoning capability, especially when it comes to complex reasoning like solving problems in CFA exams, where it is important to make each reasoning step correct. We use PA for this, which trains the model to assign higher probability mass to better generations, and has been shown to be effective in enhancing LLM reasoning capabilities (Lambert et al., 2024; Jiao et al., 2024). Specifically, we employ Direct Preference Optimization or DPO (Rafailov et al., 2023), which allows the model

[3]This is akin to 'replay' method in continual learning.

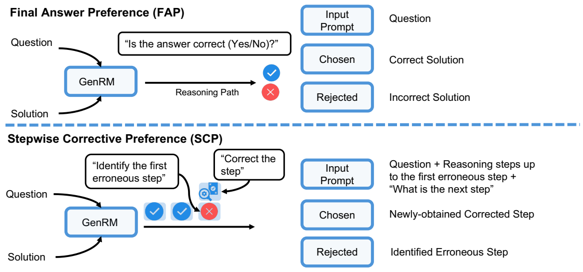

Figure 2: An overview of the proposed final answer preference (FAP) and stepwise corrective preference (SCP). In FAP, we collect trajectories from the GenRM by evaluating the entire solution. In SCP, we collect trajectories from the GenRM, by identifying and correcting the first erroneous step.

to learn from both positive and negative examples, providing a richer learning signal compared to SFT. We synthetically generate such data from the *on-policy* model, i.e., the jointly trained CPT+IT checkpoint, as it has shown the strongest performance in preliminary experiments (Appx. B.3). We propose a novel trajectory collection method that provides fine-grained step-level supervision signals (§3.2.2)

3.2.2 Data Recipe

While data quality and diversity are standard concerns in LLM training, we focus on two under-explored challenges: (a) the impact of in-domain, general-domain, and mixed-domain datasets on model performance at different training stages; (b) the generation of fine-grained supervision signals in PA to improve reasoning.

Mixture of in-domain and general-domain data. Most existing finance LLMs rely exclusively on in-domain data in post-training with the exception of FinLLM, which uses general domain data in CPT (see Table 1). Intuitively, this exclusive reliance on in-domain data can lead to forgetting of general knowledge in the original pre-trained LLM. To understand how the forgetting happens across different stages, we conduct ablations by constructing three versions of data for **each training stage**: in-domain, general-domain, and a mixture of both. Experiments (App. B) show that the impact of forgetting **decreases progressively** from CPT to IT to PA, with CPT experiencing the most severe forgetting and PA the least. Guided by these findings, we adopt a mix of in- and general-domain data for CPT+IT training to maximize both specialization and retention of essential general knowledge.

Preference data construction for reasoning. Existing training methods to improve reasoning primarily rely on outcome-based rewards, which provide sparse supervision and do not guide intermediate reasoning steps. At the same time, stepwise reward models can be computationally expensive if applied at every step. To strike a balance, we employ a Generative Reward Model (GenRM) and design Final Answer Preference (FAP) to efficiently collect preference signals at the *final answer level (outcome reward)*, while also collecting Stepwise Corrective Preference (SCP) at the reasoning step level (process reward) by asking the GenRM to identify and correct the first erroneous step. By combining these two complementary strategies, our PA provides stronger supervision signals for reasoning improvements while maintaining efficiency, making it particularly suitable for domains like finance, where both accuracy and efficiency are critical. Figure 2 illustrates the proposed method.

• **Final Answer Preference (FAP).** Given a prompt and a model generated solution, we prompt the GenRM to give a holistic judgment for the entire solution using a single "Yes" or "No" token. We then use the correct solutions as chosen samples and the incorrect solutions as rejected samples.

• **Stepwise Corrective Preference (SCP).** Since reasoning could be complex (e.g., CFA exams) and *process rewards* have been shown to be more effective in such cases (Lightman et al., 2024), we further leverage the GenRM to provide step-level signals. Instead of requesting rewards at each step, which has been shown to be unnecessary in (Lightman et al., 2024; Luo et al., 2024), we prompt the GenRM to identify the *first* erroneous step and ask it to provide a correction for that step. Using this correction, we construct a preference data sample. The input prompt is formed by concatenating the original question, the candidate reasoning steps up to the first error, and a follow-up question framed as "What is the next step?". The *chosen* response of this preference sample is the newly-obtained corrected step, while the original first erroneous step is deemed as *rejected* response. This approach produces trajectories that focus on predicting the correct next step given a reasoning prefix, unlike FAP, which requires a prediction of the entire reasoning trajectory (see App. H for prompt details).

3.3 FinDAP Training Data (FinTrain)

In FinTrain, we carefully balance the trade-off between quality and quantity of the training data at each stage. Specifically, in CPT, we leverage available general-domain supervised data, like NaturalInstrution (Mishra et al., 2022). Since such data has been carefully curated and cleaned for labeling, they maintain good quality. For quantity and diversity, which is essential for learning new domain knowledge during CPT, we collect large-scale, diverse data from relevant sources, including 70 financial websites and books covering 12 financial topics, like CFA exam preparation materials. We further use a strong LLM to filter out low-quality tasks based on an additive scale prompt (Yuan et al., 2024). This results in approximately 6B tokens.

For IT, to promote diversity, we conduct a broad survey and source general, financial, instruction-following, and reasoning datasets from public datasets. We also include large open QA datasets like FinQA (Chen et al., 2021). To ensure quality, we prioritize datasets that shown to perform well in the literature, like UltraChat (Ding et al., 2023). We also incorporate exercises or demonstrations from books that often contain human-written CoT. The final IT dataset consists of ∼3M prompts.

For PA, we use CFA preparation materials as a representative source for in-domain reasoning as they cover diverse financial scenarios, emphasize complex reasoning, and, most importantly, are derived from real-world exams. We construct preference data with FAP and SCP introduced in §3.2.2. The final PA dataset consists of about 32K prompts. Additional details are given in App. D.

3.4 FinDAP Evaluation (FinEval)

Our evaluation framework **FinEval** is designed to systematically assess model performance across unseen tasks. Unlike prior studies that rely on a narrow set of domain-specific tasks, FinEval categorizes tasks by similarity (similar vs. novel tasks), domain specificity (general vs. domain-specific vs. reasoning tasks), and evaluation methods (direct-answer vs. chain-of-thought). By structuring evaluations along these dimensions, FinEval consists of **35** tasks and can serve as a comprehensive benchmark for the expected capabilities going beyond simple task-based evaluation. We took extra care to ensure that FinEval does not duplicate any samples from FinTrain: the 10-gram contamination rate is only **0.003%**, indicating minimal overlap (see App. A). We provide details about the evaluation tasks and methods in App. E.

4 Experiments

We apply our method to the instruction-tuned Llama3-8b-inst, resulting in **Llama-Fin** (GPT-4o is used as GenRM). A summary of the

Task	Benchmark	Llama-Fin 8B	Llama3 Instruct 8B	Llama3.1 Instruct 8B	Palmyra Fin 70B	Phi 3.5-mini Instruct 3.8B	Mistral Nemo instruct 12B	GPT4o
Sentiment Ana.	FPB (Acc)	**91.13**✓	73.09	71.55	67.11	78.04	78.25	82.16
Sentiment Ana.	FiQA SA (Acc)	**95.32**✓	77.87	70.64	71.91	69.36	55.74	68.51
Monetary Policy	FOMC (Acc)	**64.31**✓	56.65	54.64	63.10	58.47	57.86	67.94
Named Entity	NER (Rouge1)	**76.69**✓	45.03	51.22	54.29	39.37	49.84	43.02
Abs Summ.	EDTSUM (Rouge1)	**53.78**✓	11.50	12.53	21.77	19.97	12.32	18.15

Table 2: Results on **similar (unseen)** tasks. Llama-Fin is highlighted in blue while the closed model is highlighted in gray. The best performing model for 8b on each benchmark is **bolded**. The overall best performance across all models is underlined. ✓ indicates that Llama-Fin outperforms the base Llama3-8b-inst.

hyper-parameters and computational resource requirements is given in Table G.1. For evaluation, we compare Llama-Fin with a wide range of baselines models, including its base model, Llama3-8b-inst, and the 8B peer, Llama3.1-8b-inst. We also include comparisons with models of other sizes, such as Phi-3.5-mini-instruct (Abdin et al., 2024) (3.8B), and Mistral-Nemo-inst (Jiang et al., 2023) (12B), as well as the *closed model* GPT-4o (OpenAI, 2023). Furthermore, we evaluated against the latest SoTA finance-specific LLM, Palmyra-Fin (70B) (Writer, 2024). Note that there are other financial LLMs available, such as FinMa (Xie et al., 2023) and FinLLaVA (Xie et al., 2024a). However, they are either not publicly available (FinLLaVA) or based on less advanced LLMs (e.g., LLaMA2). In preliminary experiments, these models performed considerably worse than our model (see App. F). Therefore, we have only included the SoTA financial LLM in our comparisons.

4.1 Main Results

Similar (unseen) tasks. To validate our approach, we first evaluate Llama-Fin on tasks that are similar (yet unseen) to the tasks used for training (e.g., test task EDTSUM (abstractive summarization) is similar to the training task TradeTheEvent (abstractive summarization)). From Table 2, we observe that Llama-Fin outperforms all other baselines in its size category by 10% - 25% absolute gain. It also surpasses significantly larger models, such as the finance-specific Palmyra-Fin (70B). Notably, Llama-Fin also exceeds the performance of GPT-4o. These results are not very surprising since the test tasks are not entirely novel, but it demonstrates the effectiveness of our data and model recipe for domain-adaptive post-training.

Novel tasks. We now evaluate the generalization of Llama-Fin on the completely novel tasks that are also aligned to the expected capabilities (FinCap). Table 3 presents the results. Below, we summarize the key takeaways from the comparison:

• **Llama-Fin preserves general concepts (rows 2-5).** We observe that Llama-Fin performs better or remains competitive with its base model in general knowledge recall tasks, indicating that it effectively preserves general capabilities and mitigating forgetting. It performs slightly worse than the base model in finance knowledge recall (MMLU-Finance), despite our finding that the CPT benefits IT (see ablations in Appendix B.3). We hypothesize that CPT helps learn concepts that are helpful but differ from those emphasized in MMLU-Finance.

• **Llama-Fin is effective in the majority of tasks (rows 6-22).** It outperforms the base model in 13 out of 17 tasks, demonstrating that our approach can lead to models that generalize well to novel, unseen tasks requiring the same capabilities.

• **Llama-Fin preserves IF/Chat capabilities (row 23).** Llama-Fin achieves a competitive MT-Bench score compared to the base model, indicating that it effectively maintains the IF capability.

• **Llama-Fin excels in reasoning tasks (rows 24-31).** For reasoning capability, Llama-Fin significantly outperforms the base models across all considered benchmarks in a large amount (up to 20% in CFA-Challenge), indicating substantial improvements in reasoning capability.

4.2 Further Analysis and Ablations

As discussed in §3.2, we performed a number of data and model ablations in pursuit of designing the best training recipe (including parameter-efficient finetuning methods like LoRA) for Llama-Fin. Those ablations are detailed in Appendix B. In this section, we present the impact of our PA strategy in the overall post-training process.

Table 4 presents the effectiveness of PA on similar tasks. We see that PA leads to improved performance in 3 out of 5 tasks, while not causing any significant forgetting on the other two. This is expected as PA primarily targets the reasoning tasks whereas these tasks do not need much reasoning.

In Table 5, we show the same ablation for the novel tasks. In **Concept (rows 2-5)** and **IF/chat (row 23)** capabilities, removing PA often leads to worse results, indicating its effectiveness. In **Task (rows 6-22)**, we see a mixed performance with and without PA. This is again not surprising as PA specifically focuses on reasoning tasks. Interestingly, we observe that for certain tasks (e.g.,

Capability	Domain	Task	Benchmark	Llama-Fin 8B	Llama3 Instruct 8B	Llama3.1 Instruct 8B	Palmyra Fin 70B	Phi 3.5-mini Instruct 3.8B	Mistral Nemo instruct 12B	GPT4o
Concept	General	Knowledge Recall	MMLU (CoT, Acc)	47.42	**48.14**	47.42	54.93	45.07	49.64	63.88
			AI2-ARC (CoT, Acc)	89.43✓	89.29	**89.80**	89.01	87.25	88.19	97.85
			Nq-open (CoT, Acc)	19.20✓	18.47	**22.52**	19.25	6.20	17.01	27.92
	Finance	Knowledge Recall	MMLU-Finance (Acc)	64.20	65.71	**66.74**	75.15	68.17	61.88	86.52
Task	Finance	Extractive Summ.	Flare-ECTSUM (Rouge1)	34.10	**35.92**	35.77	33.24	35.52	37.86	35.90
		ESG Issue	MLESG (Acc)	**40.67**✓	36.33	36.00	39.67	38.33	32.67	45.67
		Rumor Detection	MA (Acc)	84.00✓	82.60	**84.20**	62.60	75.40	85.20	73.80
		Stock Movement	SM-Bigdata (CoT, Acc)	54.14	**55.3**	46.06	48.70	53.26	53.53	49.18
			SM-ACL (CoT, Acc)	**51.99**✓	50.51	45.30	51.21	49.84	50.75	50.97
			SM-CIKM (CoT, Acc)	54.94	**55.56**	48.03	52.92	50.03	53.28	49.78
		Fraud Detection	CRA-CCF (CoT, Mcc)	0.83✓	-0.32	**2.73**	3.12	1.20	3.94	6.16
			CRA-CCFraud (CoT, Acc)	**34.03**✓	14.78	17.3	33.03	45.33	32.94	49.57
		Credit Scoring	Flare-German (CoT, Acc)	**64.00**✓	33.50	15.00	12.00	49.50	32.50	17.00
			Flare-Astralian (CoT, Acc)	44.60	**66.91**	11.51	12.95	46.76	56.12	51.80
			CRA-LendingClub (CoT, Acc)	**68.49**✓	52.69	25.38	23.40	48.87	21.03	65.03
		Distress Ident.	CRA-Polish (CoT, Mcc)	**15.30**✓	12.37	15.07	13.78	69.14	11.18	17.38
			CRA-Taiwan (CoT, Acc)	**40.81**✓	12.01	35.97	52.58	69.96	57.88	8.57
		Claim Analysis	CRA-ProroSeguro (CoT, Acc)	35.14	**96.98**	44.33	56.20	25.86	32.58	96.60
			CRA-TravelInsurance (CoT,Acc)	41.52✓	6.39	80.31	17.28	**94.48**	73.64	54.03
		Tabular QA	*Flare-TATQA (CoT, Acc)	**66.61**✓	63.43	63.70	64.21	57.70	66.40	74.90
		Open QA	*Finance Bench (CoT, Acc)	**54.00**✓	52.70	38.00	56.67	40.70	55.30	51.30
IF/Chat	General	Precise IF	MT-bench (1,2 turn avg)	7.36	7.88	**7.92**	5.80	8.38	7.84	9.10
Reasoning	Math	Math Reasoning	MathQA (CoT, Acc)	**55.08**✓	51.16	49.35	41.51	39.40	52.46	70.82
	General	Social Reasoning	Social-IQA (CoT, Acc)	**75.23**✓	68.83	70.73	77.28	72.82	62.95	78.92
		Common Sense	Open-book-qa (CoT, Acc)	**82.60**✓	77.00	82.20	87.00	80.20	76.40	94.60
			Hellaswag (CoT, Acc)	**81.90**✓	73.34	69.10	69.69	67.89	61.74	81.76
			Winogrande (CoT, Acc)	**70.32**✓	62.51	66.69	74.27	72.22	65.82	85.71
			PIQA (CoT, Acc)	**85.85**✓	79.82	81.45	86.72	82.05	77.91	94.34
	Finance	Exam	CFA-Easy (CoT, Acc)	**66.28**✓	60.56	60.47	36.05	61.24	65.89	83.14
			CFA-Challnge (CoT, Acc)	**55.56**✓	34.44	35.56	25.56	48.89	43.33	74.44

Table 3: Results on the **novel** tasks. The notations are the same as in Table 2. '*' indicates that 'GPT4o' is used as the judge. 'Mcc' refers to Matthews correlation coefficient, usually used in highly imbalanced data (Xie et al., 2024a).

Task	Benchmark	Llama-Fin	Llama-Fin (w/o PA)
Sentiment Ana.	FPB	91.13	**92.99**
Sentiment Ana.	FiQA SA	**95.32**	94.47
Monetary Policy	FOMC	**64.31**	63.10
Named Entity	NER	**76.69**	74.33
Abs. Summ.	EDTSUM	53.78	**54.21**

Table 4: Ablation on PA on **similar (unseen)** evaluation set.

CRA-TravelInsurance, CRA-Taiwan and CRA-ProroSeguro), PA negatively impacts performance, resulting in worse outcomes compared to without PA. Even GPT-4o performs poorly in these tasks. This suggests that for some tasks, leveraging reasoning capabilities might not be beneficial, as these tasks could be inherently "easy" and solvable without the need for explicit reasoning. Such observations align with prior findings (Sprague et al., 2024; Liu et al., 2024). In **Reasoning (rows 24-31)**, Llama-Fin is significantly better than without PA variant, further confirming that our proposed FAP and SCP are particularly effective in improving reasoning performance beyond the already strong checkpoint of Llama-Fin (w/o PA).

Capability	Domain	Task	Benchmark	Llama-Fin 8B	Llama-Fin (w/o PA)
Concept	General	Knowledge Recall	MMLU	**47.42**	47.22
			AI2-ARC	**89.43**	88.95
			Nq-open	**19.20**	16.20
	Finance	Knowledge Recall	MMLU-Finance	**64.20**	63.93
Task	Finance	Extract Summ.	Flare-ECTSUM	34.10	**34.41**
		ESG Issue	MLESG	40.67	**42.00**
		Rumor Detection	MA	84.00	**84.60**
		Stock Movement	SM-Bigdata	**54.14**	52.04
			SM-ACL	**51.99**	49.89
			SM-CIKM	**54.94**	44.88
		Fraud Detection	CRA-CCF	**0.83**	0.61
			CRA-CCFraud	**34.03**	32.32
		Credit Scoring	Flare-German	**64.00**	60.50
			Flare-Astralian	44.60	**51.80**
			CRA-LendingClub	**68.49**	65.96
		Distress Ident.	CRA-Polish	**15.30**	0.65
			CRA-Taiwan	40.81	**96.41**
		Claim Analysis	CRA-ProroSeguro	35.14	**86.57**
			CRA-TravelInsurance	41.52	**98.50**
		Tabular QA	*Flare-TATQA	**66.61**	66.43
		Open QA	*Finance Bench	**54.00**	52.00
IF/Chat	General	Precise IF	MT-bench	**7.36**	7.29
Reasoning	Math	Math Reasoning	MathQA	**55.08**	54.30
	General	Social Reasoning	Social-IQA	**75.23**	73.64
		Common Sense	Open-book-qa	**82.60**	79.20
			Hellaswag	**81.90**	78.92
			Winogrande	**70.32**	67.48
			PIQA	**85.85**	84.39
	Finance	Exam	CFA-Easy	**66.28**	62.31
			CFA-Challnge	**55.56**	35.56

Table 5: Abaltion on PA on **novel** evaluation set.

5 Conclusion

We introduce FINDAP, an open SoTA finance-specific post-training framework, consists of *FinCap* that identifies four key capabilities; *FinRec* which jointly trains CPT and IT, and constructing PA preference data with stepwise signals; *FinTrain* that implements FinRec; and *FinEval*, a comprehensive evaluation setup. Under FINDAP, we develop Llama-Fin, a SoTA finance LLM. In this development, we conduct a systematic study on effectively adapting a target domain through post-training. For each stage, we reveal the distinct challenges, objectives, and effective strategies. Looking ahead, we aim to scale up the base LLM and explore additional domain-specific capabilities using FINDAP.

6 Limitations

While the recipe for FINDAP and Llama-Fin are effective, the performance on novel unseen tasks still requires further improvement. For example, selectively employing reasoning capabilities only for questions that require such advanced reasoning might give better results. Additionally, the data recipe is currently based on full-scale empirical experiments, which can be time-intensive. Developing low-cost experiments to reliably indicate the effectiveness of data in post-training could streamline this process and accelerate the development iteration. It is also worth noting that the same recipe may not generalize well to other model families. Different architectures or pretraining strategies might require tailored recipe to achieve optimal results, emphasizing the need for adaptability in recipe design in future research. Finally, while we focus on the four key capabilities in finance, we acknowledge there could be additional requirements (e.g., multi-modality and sensitivity, see details in Appendix §C), and leave them for future work.

References

Marah Abdin et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Salinas Alvarado, Julio Cesar, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90, Parramatta, Australia.

Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019a. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *Preprint*, arXiv:1905.13319.

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019b. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

Stephen H. Bach, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, Zaid Alyafeai, Manan Dey, Andrea Santilli, Zhiqing Sun, Srulik Ben-David, Canwen Xu, Gunjan Chhablani, Han Wang, Jason Alan Fries, Maged S. Al-shaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Xiangru Tang, Mike Tian-Jian Jiang, and Alexander M. Rush. 2022. Promptsource: An integrated development environment and repository for natural language prompts. *Preprint*, arXiv:2202.01279.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022. Constitutional ai: Harmlessness from ai feedback. *Preprint*, arXiv:2212.08073.

Gagan Bhatia, El Moatez Billah Nagoudi, Hasan Cavusoglu, and Muhammad Abdul-Mageed. 2024. Fintral: A family of gpt-4 level multimodal financial large language models. *Preprint*, arXiv:2402.10986.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Ethan Callanan, Amarachi Mbakwe, Antony Papadimitriou, Yulong Pei, Mathieu Sibue, Xiaodan Zhu, Zhiqiang Ma, Xiaomo Liu, and Sameena Shah. 2024. Can GPT models be financial analysts? an evaluation of ChatGPT and GPT-4 on mock CFA exams. In *Proceedings of the Eighth Financial Technology and Natural Language Processing and the 1st Agent AI for Scenario Planning*, Jeju, South Korea. -.

Oana-Maria Camburu, Tim Rockt"aschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9539–9549. Curran Associates, Inc.

Chung-Chi Chen, Yu-Min Tseng, Juyeon Kang, Anaïs Lhuissier, Min-Yuh Day, Teng-Tsai Tu, and Hsin-Hsi Chen. 2023a. Multi-lingual ESG issue identification. In *Proceedings of the Fifth Workshop on Financial Technology and Natural Language Processing and the Second Multimodal AI For Financial Forecasting*.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco

Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, et al. 2023b. Meditron-70b: Scaling medical pretraining for large language models. *arXiv preprint arXiv:2311.16079*.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, 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 3697–3711, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Daixuan Cheng, Shaohan Huang, and Furu Wei. 2024. Adapting large language models via reading comprehension. In *The Twelfth International Conference on Learning Representations*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Pierre Colombo, Telmo Pires, Malik Boudiaf, Rui Filipe Coimbra Pereira de Melo, Gabriel Hautreux, Etienne Malaboeuf, Johanne Charpentier, Dominic Culver, and Michael Desa. 2024a. SaulLM-54b & saulLM-141b: Scaling up domain adaptation for the legal domain. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Pierre Colombo, Telmo Pessoa Pires, Malik Boudiaf, Dominic Culver, Rui Melo, Caio Corro, Andre FT Martins, Fabrizio Esposito, Vera Lúcia Raposo, Sofia Morgado, et al. 2024b. Saullm-7b: A pioneering large language model for law. *arXiv preprint arXiv:2403.03883*.

Daya Guo DeepSeek-AI, Dejian Yang, and Li et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.

Duanyu Feng, Yongfu Dai, Jimin Huang, Yifang Zhang, Qianqian Xie, Weiguang Han, Zhengyu Chen, Alejandro Lopez-Lira, and Hao Wang. 2024. Empowering many, biasing a few: Generalist credit scoring through large language models. *Preprint*, arXiv:2310.00566.

Tianyu Gao, Alexander Wettig, Howard Yen, and Danqi Chen. 2024. How to train long-context language models (effectively). *Preprint*, arXiv:2410.02660.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies. *Transactions of the Association for Computational Linguistics (TACL)*.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. Textbooks are all you need. *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*, Online. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Hans Hofmann. 1994. Statlog (German Credit Data). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5NC77.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024a. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*.

Siming Huang, Tianhao Cheng, Jason Klein Liu, Jiaran Hao, Liuyihan Song, Yang Xu, J. Yang, J. H. Liu, Chenchen Zhang, Linzheng Chai, Ruifeng Yuan, Zhaoxiang Zhang, Jie Fu, Qian Liu, Ge Zhang, Zili

Wang, Yuan Qi, Yinghui Xu, and Wei Chu. 2024b. Opencoder: The open cookbook for top-tier code large language models.

Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. 2023. Financebench: A new benchmark for financial question answering. *Preprint*, arXiv:2311.11944.

Hamish Ivison, Yizhong Wang, Jiacheng Liu, Zeqiu Wu, Valentina Pyatkin, Nathan Lambert, Noah A. Smith, Yejin Choi, and Hannaneh Hajishirzi. 2024. Unpacking dpo and ppo: Disentangling best practices for learning from preference feedback.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Fangkai Jiao, Chengwei Qin, Zhengyuan Liu, Nancy F. Chen, and Shafiq Joty. 2024. Learning planning-based reasoning by trajectories collection and process reward synthesizing. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Miami, Florida, USA. Association for Computational Linguistics.

Zixuan Ke, Weize Kong, Cheng Li, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. 2024. Bridging the preference gap between retrievers and llms. *arXiv preprint arXiv:2401.06954*.

Zixuan Ke, Yijia Shao, Haowei Lin, Tatsuya Konishi, Gyuhak Kim, and Bing Liu. 2023. Continual pre-training of language models. In *International Conference on Learning Representations (ICLR)*.

Zixuan Ke, Yijia Shao, Haowei Lin, Hu Xu, Lei Shu, and Bing Liu. 2022. Adapting a language model while preserving its general knowledge. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Danupat Khamnuansin, Atthakorn Petchsod, Anuruth Lertpiya, Pornchanan Balee, Thanawat Lodkaew, Tawunrat Chalothorn, Thadpong Pongthawornkamol, and Monchai Lertsutthiwong. 2024. Thalle: Text hyperlocally augmented large language extension – technical report. *Preprint*, arXiv:2406.07505.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. Qasc: A dataset for question answering via sentence composition. *arXiv:1910.11473v2*.

Hyunwoo Kim, Jack Hessel, Liwei Jiang, Peter West, Ximing Lu, Youngjae Yu, Pei Zhou, Ronan Le Bras, Malihe Alikhani, Gunhee Kim, Maarten Sap, and Yejin Choi. 2022. Soda: Million-scale dialogue distillation with social commonsense contextualization. *ArXiv*, abs/2212.10465.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. 2024. Tülu 3: Pushing frontiers in open language model post-training.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. 2023. Large language models in finance: A survey. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pages 374–382.

Wing Lian, Guan Wang, Bleys Goodson, Eugene Pentland, Austin Cook, Chanvichet Vong, and "Teknium". 2023. Slimorca: An open dataset of gpt-4 augmented flan reasoning traces, with verification.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. *ACL*.

Ryan Liu, Jiayi Geng, Addison J. Wu, Ilia Sucholutsky, Tania Lombrozo, and Thomas L. Griffiths. 2024. Mind your step (by step): Chain-of-thought can reduce performance on tasks where thinking makes humans worse. *Preprint*, arXiv:2410.21333.

Team LLaMA. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, Jiao Sun, and Abhinav Rastogi. 2024. Improve mathematical reasoning in language models by automated process supervision. *Preprint*, arXiv:2406.06592.

Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. 2023. Wizardcoder: Empowering code large language models with evol-instruct.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Mário Maia, Jonathan Berant, José Farinha, and André Freitas. 2018. Www'18 open challenge: Financial opinion mining and question answering. In *Proceedings of the 2018 World Wide Web Conference*, pages 1941–1942. International World Wide Web Conferences Steering Committee.

Dean Malmgren. 2014. Textract.

P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *ACL*.

Arindam Mitra, Hamed Khanpour, Corby Rosset, and Ahmed Awadallah. 2024. Orca-math: Unlocking the potential of slms in grade school math. *Preprint*, arXiv:2402.14830.

Rajdeep Mukherjee, Abhinav Bohra, Akash Banerjee, Soumya Sharma, Manjunath Hegde, Afreen Shaikh, Shivani Shrivastava, Koustuv Dasgupta, Niloy Ganguly, Saptarshi Ghosh, and Pawan Goyal. 2022. ECTSum: A new benchmark dataset for bullet point summarization of long earnings call transcripts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yasumasa Onoe, Michael J.Q. Zhang, Eunsol Choi, and Greg Durrett. 2021. Creak: A dataset for commonsense reasoning over entity knowledge. *OpenReview*.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Richard Yuanzhe Pang, Weizhe Yuan, He He, Kyunghyun Cho, Sainbayar Sukhbaatar, and Jason E Weston. 2024. Iterative reasoning preference optimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *Preprint*, arXiv:2406.17557.

Ross Quinlan. 1987. Statlog (Australian Credit Approval). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C59012.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale. *arXiv preprint arXiv:1907.10641*.

William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. 2022. Self-critiquing models for assisting human evaluators. *Preprint*, arXiv:2206.05802.

Saxton, Grefenstette, Hill, and Kohli. 2019. Analysing mathematical reasoning abilities of neural models. *arXiv:1904.01557*.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. 2022. Fine-tuned language models are continual learners. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. 2024. Rewarding progress: Scaling automated process verifiers for llm reasoning. *Preprint*, arXiv:2410.08146.

Agam Shah, Suvan Paturi, and Sudheer Chava. 2023. Trillion dollar words: A new financial dataset, task & market analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Toronto, Canada. Association for Computational Linguistics.

Soumya Sharma, Tapas Nayak, Arusarka Bose, Ajay Kumar Meena, Koustuv Dasgupta, Niloy Ganguly, and Pawan Goyal. 2022. Finred: A dataset for relation extraction in financial domain. In *Proceedings of The 2nd Workshop on Financial Technology on the Web (FinWeb)*.

Ankur Sinha, Tanmay Khandait, and vincent. 2020. Impact of news on the commodity market: Dataset and results. *CoRR*, abs/2009.04202.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *Preprint*, arXiv:2408.03314.

Yejun Soun, Jaemin Yoo, Minyong Cho, Jihyeong Jeon, and U Kang. 2022. Accurate stock movement prediction with self-supervised learning from sparse noisy tweets. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 1691–1700. IEEE Computer Society.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2024. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. *Preprint*, arXiv:2409.12183.

Lewis Tunstall, Edward Emanuel Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro Von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M Rush, and Thomas Wolf. 2024. Zephyr: Direct distillation of LM alignment. In *First Conference on Language Modeling*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *NUT@EMNLP*.

Writer. 2024. Palmyra-Fin-70B-32k: a powerful LLM designed for Finance. `https://dev.writer.com`.

Huizhe Wu, Wei Zhang, Weiwei Shen, and Jun Wang. 2018. Hybrid deep sequential modeling for social text-driven stock prediction. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 1627–1630, New York, NY, USA. Association for Computing Machinery.

Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023. Pixiu: A large language model, instruction data and evaluation benchmark for finance. *Preprint*, arXiv:2306.05443.

Qianqian Xie, Dong Li, Mengxi Xiao, Zihao Jiang, Ruoyu Xiang, Xiao Zhang, Zhengyu Chen, Yueru He, Weiguang Han, Yuzhe Yang, Shunian Chen, Yifei Zhang, Lihang Shen, Daniel Kim, Zhiwei Liu, Zheheng Luo, Yangyang Yu, Yupeng Cao, Zhiyang Deng, Zhiyuan Yao, Haohang Li, Duanyu Feng, Yongfu Dai, VijayaSai Somasundaram, Peng Lu, Yilun Zhao, Yitao Long, Guojun Xiong, Kaleb Smith, Honghai Yu, Yanzhao Lai, Min Peng, Jianyun Nie, Jordan W. Suchow, Xiao-Yang Liu, Benyou Wang, Alejandro Lopez-Lira, Jimin Huang, and Sophia Ananiadou. 2024a. Open-finllms: Open multimodal large language models for financial applications. *Preprint*, arXiv:2408.11878.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I. Wang, Victor Zhong, Bailin Wang, Chengzu Li, Connor Boyle, Ansong Ni, Ziyu Yao, Dragomir Radev, Caiming Xiong, Lingpeng Kong, Rui Zhang, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. *EMNLP*.

Yong Xie, Karan Aggarwal, and Aitzaz Ahmad. 2024b. Efficient continual pre-training for building domain specific large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10184–10201, Bangkok, Thailand. Association for Computational Linguistics.

Yumo Xu and Shay B. Cohen. 2018. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia. Association for Computational Linguistics.

Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. 2023. Fingpt: Open-source financial large language models. *FinLLM Symposium at IJCAI 2023*.

Linyi Yang, Eoin M. Kenny, Tin Lok James Ng, K. Z. Zhang P.K. Kannan Y Yang, Barry Smyth, and Ruihai Dong. 2020. Generating plausible counterfactual explanations for deep transformers in financial text classification. In *COLING*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. 2024. Self-rewarding language models. In *Forty-first International Conference on Machine Learning*.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2023. Mammoth: Building math generalist models through hybrid instruction tuning. *arXiv preprint arXiv:2309.05653*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Jianguo Zhang, Kun Qian, Zhiwei Liu, Shelby Heinecke, Rui Meng, Ye Liu, Zhou Yu, Silvio Savarese, and Caiming Xiong. 2023. Dialogstudio: Towards richest and most diverse unified dataset collection for conversational ai. *arXiv preprint arXiv:2307.10172*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

Zhihan Zhou, Liqian Ma, and Han Liu. 2021. Trade the event: Corporate events detection for news-based event-driven trading. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2114–2124, Online. Association for Computational Linguistics.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. 2021. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online. Association for Computational Linguistics.

A Preventing Data Contamination

When designing FinEval, we took extra care to ensure that evaluation tasks do not duplicate any samples from FinTrain. To further verify this, we followed your suggestion and computed string matches between FinEval and FinTrain.

Specifically, we adopted the decontamination procedure described in the Hugging Face blog you referenced. A training sample is contaminated if it is overlapped with any evaluation sample. The contamination ratio is computed as the fraction of contamination samples in the training samples. Based on this criterion, we report two contamination ratios:

- **0%** under the strictest setting, where only complete sample matches are considered contamination.
- **0.003%** using the method described in the blog—where 10-gram matches are used for pre-identification, followed by difflib.SequenceMatcher. If over 50% of its characters match any of the evaluation samples, the training sample is considered contaminated.

These contamination rates are extremely low, indicating minimal overlap between FinTrain and FinEval. Upon manual inspection of the few samples flagged by the 50% character overlap rule, we found they involve either (1) partial overlap in the question format or instruction prompt, which is expected for the similar tasks where the task type (e.g., sentiment analysis) has been seen, but the content remain unseen; or (2) partial overlap in the input content (e.g., shared elements in bank transcripts), but the specific question and answers are unseen. In both cases, these do not indicate memorization or leakage of benchmark content.

B Ablations and Understanding FinDAP

B.1 Continual Pre-training

In order to expose the LLM to domain-specific concepts, we first conduct continual pre-training (CPT). In CPT, we feed plain text to the LLM and perform *next token prediction*.

From Text to CPT Data. A key challenge in CPT is what kind of data we should use. Given the general and domain-specific texts introduced in §D, we can construct three versions of CPT data, *CPT-In* contains only the financial (in-domain) text, *CPT-Gen* contains only the general domain data, and *CPT-Mix* contains the mixture of the CPT-In and CPT-Gen.

Key Data Experiments. We conduct CPT on each of the three versions of data. As shown in Figure B.1, we observe that while CPT-In and CPT-Gen outperforms in financial (Fig B.1a) and general (Fig B.1b) tasks, respectively, CPT-Mix achieves the best overall. This is expected as CPT-In can cause *catastrophic forgetting* on the general tasks, while incorporating general domain concepts in CPT-Mix acts as 'replay' mechanism to mitigate it (Scialom et al., 2022). We can also see that none of the CPT-trained LLMs outperform their base. This is unexpected because CPT invovles post-training on more specialized data, which should enhance the performance. By analyzing the output, we attribute this issue to the model forgetting how to follow instructions effectively after CPT. To quantify this finding, we evaluate the instruction following ability of these models using MT-Bench. The two-turn average scores for CPT-Mix, CPT-In, and CPT-Gen are 1, 1, and 1.0125, respectively, while the base model, achieves a score of 7.8875. These confirm that the conventional CPT applied to a instruction-tuned LLM can cause serious forgetting on instruction-following (IF) capability. In §B.2, we will see how jointly train IT and CPT can help mitigate such forgetting issue.

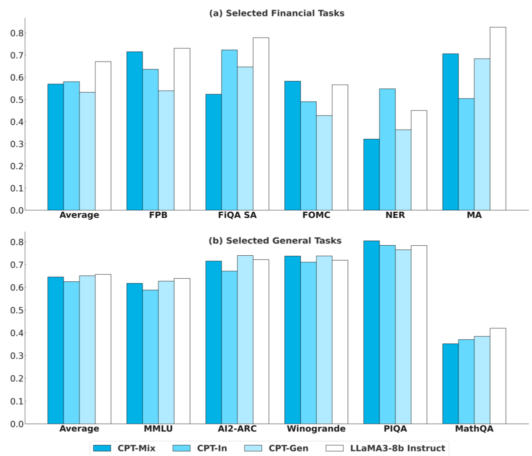

Figure B.1: Average performance on selected datasets for training Llama3-8b-instruct on our CPT-In, CPT-Gen and CPT-Mix. The Y-axis represents the same performance metrics as those reported in Tables 2 and 3. The selected datasets are chosen for illustration purpose based on their ability to illustrate the general trend.

B.2 Instruction Following

To adapt the LLM to domain-specific and IF tasks, we conduct IT. The key different between IT and CPT is that IT *masks out the instruction* and *takes as input supervised tasks*.

From Prompt to IT Data. We introduced our prompt curation in §D. We create the responses for IT by *filtering existing responses* or *creating new responses*. For prompts with existing responses, we generally keep the original responses if they were written by a human or a strong model, such as GPT-4. We also filter out empty responses. For prompts without responses, for example, exercises extracted from books that may not have solutions provided, we generate new responses using GPT-4o. Similar to CPT data, we construct three versions of IT data, *IT-In*, which contains only financial (in-domain) tasks, *IT-Gen*, which contains only general tasks, and *IT-Mix*, which includes a mixture of the IT-In and IT-Gen.

Key Data Experiments. Similar to CPT, we conduct IT to each of the three versions. From Figure B.2, we observe that unlike CPT, forgetting is significantly reduced. Specifically, all versions of IT are no longer worse than their base versions, indicating that the ability to follow instructions is not as severely forgotten as in CPT. This is further supported by the MT-Bench scores, where we obtained 7.2031, 6.2094, and 7.3219 for IT-Mix, IT-In and IT-Gen, respectively, all of which are significantly better than the CPT counterparts.

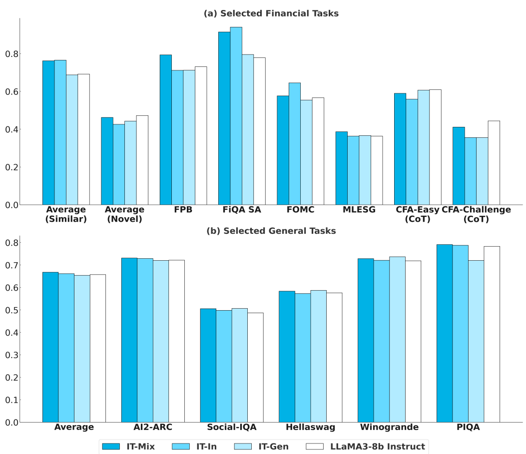

Figure B.2: Average performance on selected datasets for training Llama3-8b-instruct on our IT-In, IT-Gen and IT-Mix.

We observe that IT-Mix is slightly better than other data versions, suggesting that mixing general tasks remains helpful to mitigating forgetting of general concepts and tasks, although the effect is much less pronounced compared to CPT. We also see that similar tasks improve significantly over base model while novel tasks (including financial tasks and general tasks) show little change. This indicates that, in contrast to CPT, domain has less impact in IT, but task generalization is a challenging issue.

Comparison with LoRA. Another popular approach to adapt the LLM to specific domain is **Parameter-efficient Fine-tuning (PEFT)**, where the LLM parameters remain fixed, and only a small set of additional parameters are trained. This approach naturally mitigates forgetting issues and is more efficient in terms of trainable parameters. However, whether it can achieve performance comparable to full-model training is unclear. In Figure B.3, we experiment with PEFT, specifically using LoRA (Hu et al., 2021), with a rank size of 128,[4] and compare its performance with full-model finetuning (IT-Mix). We observe that with and without LoRA performs similarly, confirming that LoRA is effective for task adaptation. However, the novel tasks still show little improvement, highlighting that task generalization still remains a significant challenge.

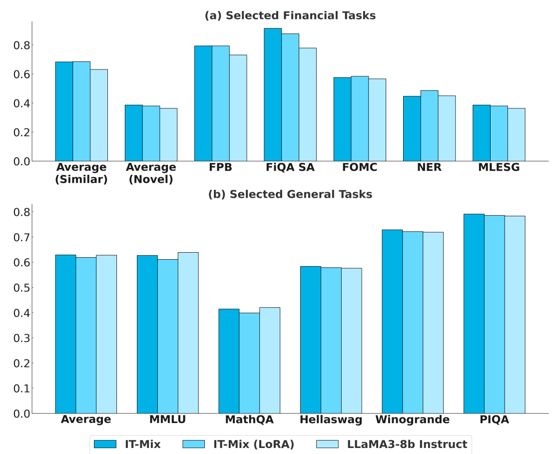

Figure B.3: Average performance on selected datasets for training Llama3-8b-instruct on IT-Mix with full-model finetuning (IT-Mix) and LoRA finetuning (IT-Mix (LoRA)).

A plausible reason for the lack of task generalization is that effective generalization may require exposure to a diverse range of tasks (Wei et al., 2022),

[4] Further decreasing or increasing the rank size did not show improvement in our preliminary experiments. For example, rank size of 32, 128 and 512 yield overall averages across 10 general tasks of 0.5267, 0.5331, and 0.5215, respectively, showing only minor differences.

which is often impractical in certain domains, particularly long-tail ones. However, *concepts* themselves may be inherently more generalizable due to the shared nature of concepts across tasks. Based on this, we propose adding CPT either before or concurrently with the IT stage and conduct training experiments accordingly.

B.3 Combining CPT and IT

A natural choice is to conduct CPT and IT sequentially (Lambert et al., 2024). On the one hand, this is flexible as it allows for different settings (e.g., data size) in each stage. On the other hand, it does not help prevent forgetting during the CPT stage, leaving the LLM dependent on IT to 'recover' its instruction-following capability. To make a more grounded decision, we conduct experiments on both sequential and joint training approaches. In joint training, an additional hyperparameter to consider is the mixture ratio. We *down-sample* CPT data to match the size of IT data. In "Other sampling strategies" section, we will show this is the most effective strategy.

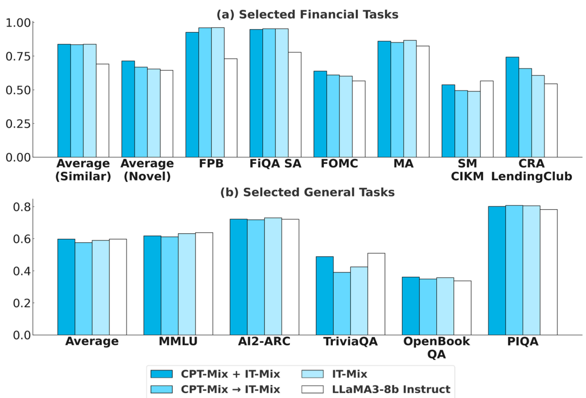

Figure B.4: Average performance on selected datasets for training Llama3-8b-instruct on CPT-Mix and IT-Mix jointly (CPT-Mix + IT-Mix) and sequentially (CPT-Mix $\rightarrow$ IT-Mix).

Figure B.4 illustrates the comparison between joint and sequential training. In both cases, different from IT-only results shown in Figure B.2, we see improved performance on similar and novel tasks. This supports our hypothesis that CPT can help improve the generalization of IT, as the concepts are likely able to be shared across different tasks. It is further interesting to see that even the general tasks are improved, indicating that there could be positive transfer between CPT and IT. Comparing the two, we observe that joint training outperforms sequential training across financial and general tasks, as well as similar and novel tasks, highlighting the importance of preventing forgetting of CPT and knowledge transfer between CPT and IT.

Other Sampling strategies. Besides down-sampling, we also evaluate the performance under a 'no-sampling' setting. Figure B.5 shows the results. We observe that in both joint and sequential training, down-sampling yields better results on financial tasks. This is understandable because down-sampling assigns more weight to IT, which is beneficial for the financial tasks. Interestingly, we observe the opposite trend for general tasks: no-sampling performs better. We hypothesize that this is because having more CPT data helps preserve general concepts more effectively, although it may diminish instruction-following abilities.

Comparison with LoRA. In Section B.2, we showed that LoRA can effectively adapt tasks but still suffers from task generalization. While we already showed that CPT can help in full-model training setting, we now explore whether CPT can help in the PEFT setting as well. Figure B.6 presents the results of applying LoRA for IT and LoRA for both CPT and IT. Surprisingly, we find that full fine-tuning significantly outperforms the LoRA counterparts across similar and novel tasks. This finding contrasts with our previous observations in Figure B.3, where performance with and without LoRA was comparable. Our results reveal that knowledge transfer from CPT to IT, which is crucial for task generalization, requires full-model training.

B.4 Preference Alignment

Negligible Forgetting in PA. As with CPT and IT, we begin by performing an ablation study on different data versions to evaluate their effectiveness. Since the degree of forgetting diminishes from CPT to IT (as observed in §B.2), we expect it to be even less pronounced in PA. To quickly evaluate this hypothesis, we take a naive approach and create *PA-Mix* and *PA-In* by using either the provided or GPT4o generated responses (as done for IT in §B.2) as the 'chosen' samples and the output of 'CPT+IT' checkpoint as the 'rejected' ones, based on the prompts of IT-Mix and IT-In, respectively.

Figure B.7 shows the results after PA training for PA-In and PA-Mix from the 'CPT+IT' check-

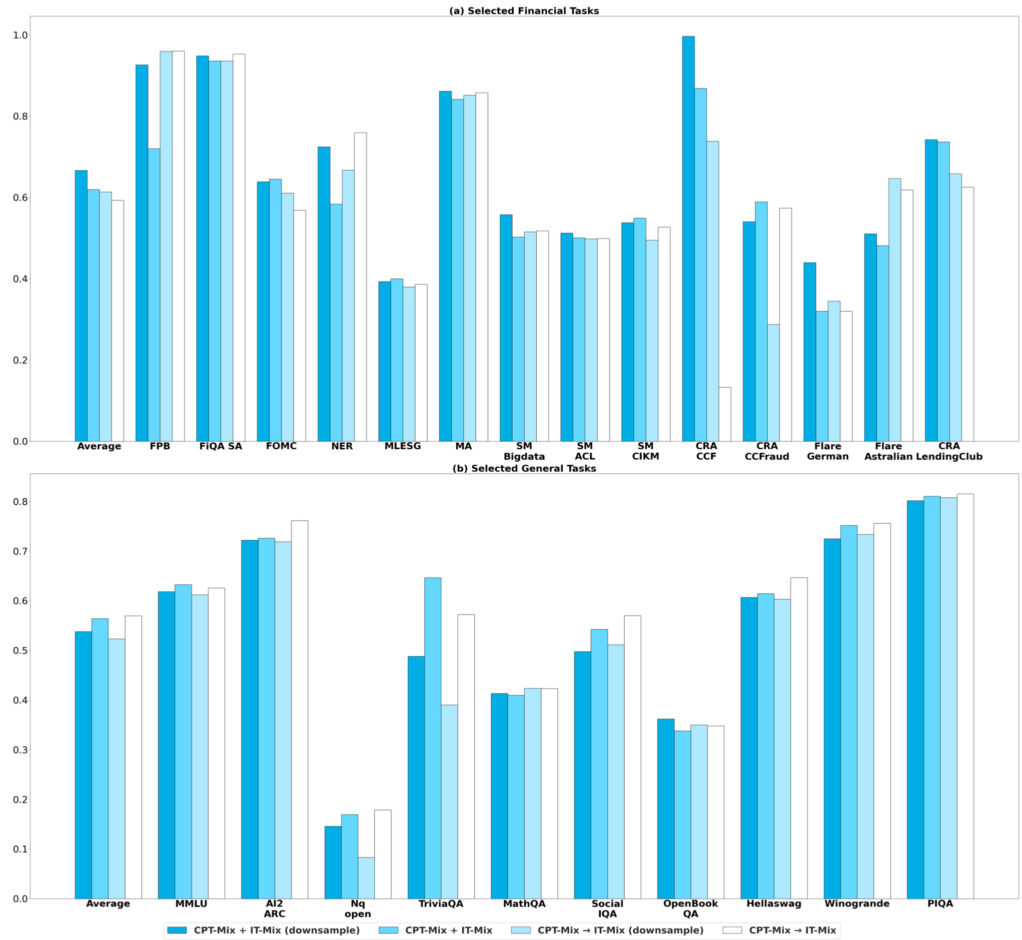

Figure B.5: Average and selected datasets performance from down-sampling or no-sampling on CPT.

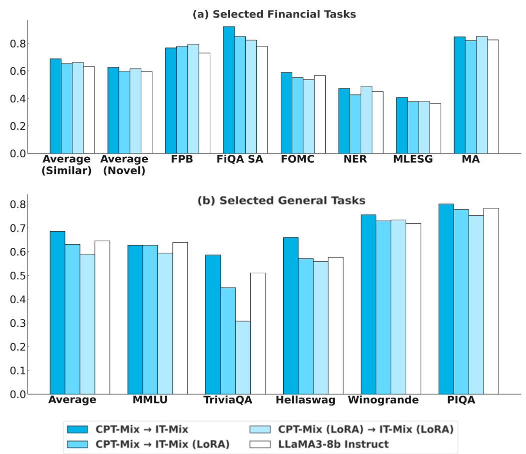

Figure B.6: Average performance on selected datasets for PEFT or full model fine-tuning for CPT and IT.

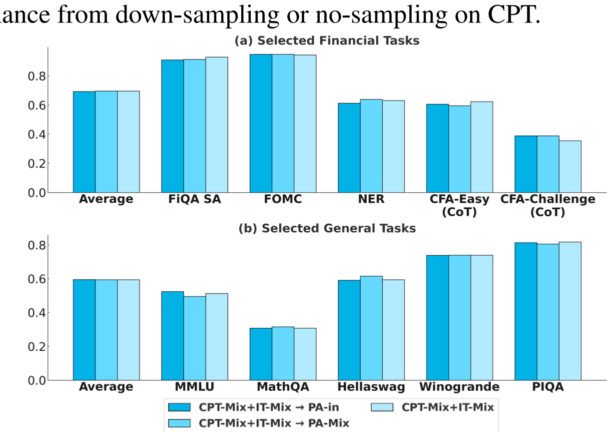

Figure B.7: Average performance on selected datasets for PA training from the 'CPT+IT' checkpoint on PA-Mix and PA-In.

point.[5] We observe that PA-In performs comparably to PA-Mix, indicating that it may not be essential to include general tasks to prevent forgetting of concepts or tasks, unlike the cases of CPT and IT. This suggests that PA training can focus on in-domain tasks, without requiring a broader set of general tasks or raising concerns about forgetting. Given this, we use CFA exams (Table D.2 in §D) as a representative source for in-domain reasoning because they cover diverse financial scenarios, em-

[5] PA trained from Llama3-8b-instruction has shown worse results compared to training from the 'CPT+IT' checkpoint in our preliminary experiments, as PA requires a strong initialization checkpoint. For instance, PA-Mix from Llama3-8b-instruction achieves only 29.99 on EDTSUM, whereas the CPT+IT' counterpart achieves 54.21. As a results, we only investigate training PA from 'CPT+IT' checkpoint.

phasize complex reasoning, and, most importantly, are derived from real-world exams. These characteristics make them a strong proxy for a broader range of financial tasks, ensuring that the model generalizes effectively within the financial domain while simplifying the training process.

Another crucial observation is that there is not much difference even for unseen similar tasks (FiQA SA, FOMC and NER) and reasoning tasks (CFA-Easy and CFA-Challenge). This highlights the limitations of the current naive PA approach and suggests room for further improvement. In FINDAP, we propose a novel PA approach that constructs preference data guided by both outcome and process reward signals.

C Other Capabilities

Besides those core capabilities mentioned in §3.1, domains may vary significantly in their *sensitivity*. For instance, the medical domain is highly sensitive, requiring utmost accuracy and strict adherence to ethical considerations. In contrast, domains such as entertainment may have more relaxed requirements. Another important consideration is *multi-modality*, as some domains require handling multiple types of input and output formats. For example, the healthcare domain may involve processing medical images alongside textual reports, while the e-commerce domain may integrate product descriptions, images, and customer reviews into a unified response. Similarly, scientific research often combines charts, graphs, and textual analysis to present findings effectively.

D FinTrain

Continual Pre-training Text Curation To introduce domain concepts while preserving general concepts, we curate texts for CPT. Table D.1 summarizes the texts curation datasets. Specially, for general concepts, research has shown that a 'small' amount of general text (as little as 1%) can effectively mitigate the forgetting issue (Scialom et al., 2022). Therefore, we focus on collecting a relatively small but high-quality set of general-domain text. To achieve this, we use *verifiable text*, which is text written by humans and previously used in supervised tasks in the literature. Note that this contrasts with using *unverifiable* web text such as C4 (Raffel et al., 2020).

For domain concept, our goal is to collect both a large volume of data and maintain high quality.

Capability	Domain	Dataset	Size	Reference
Concept	General	NaturalInstrution	100,000	Mishra et al. (2022)
		PromptSource	100,000	Bach et al. (2022)
		Math	29,837	Amini et al. (2019b)
		Aqua	97,500	Ling et al. (2017)
		CREAK	10,200	Onoe et al. (2021)
		ESNLI	549,367	Camburu et al. (2018)
		QASC	8,130	Khot et al. (2020)
		SODA	1,190,000	Kim et al. (2022)
		StrategyQA	2,290	Geva et al. (2021)
		UnifiedSKG	779,000	Xie et al. (2022)
		GSM8K	7,470	Cobbe et al. (2021)
		ApexInstr	1,470,000	Huang et al. (2024b)
		DeepmindMath	379,000	Saxton et al. (2019)
		DialogueStudio	1,070,000	Zhang et al. (2023)
	Finance	**Fineweb-Fin**	4,380,000	-
		Book-Fin	4,500	-
Total			10,177,294	

Table D.1: Summary of curated texts. New datasets released with FINDAP are color-highlighted for emphasis.

Following practices from the literature on training general LLMs (Lambert et al., 2024; Gunasekar et al., 2023), we source financial texts from primarily relevant websites and books. Specifically, we source financial text from two primary resources. The first source is *web text*, where we filter non-financial content from the FineWeb using URLs like '`sec.gov`' and '`investopedia.com`'. The second source is *books*. We manually select 10 finance-related topics (e.g., 'economics' and 'management'), download books on these topics, and convert them to text using OCR (Malmgren, 2014). Since OCR can make mistakes, we further employ a strong LLM to filter out content lacking educational value or unrelated to finance. Details on the financial URLs, finance-related topics, and the prompts used for filtering is shown below:

- **Selected Financial URLs.** We curated a selection of *70* financial websites to comprehensively cover diverse aspects of finance-related content on the web. These include trusted sources from financial institutions, regulatory agencies, educational platforms, and industry-specific news outlets. This diverse collection ensures representation across sub-domains such as investment, banking, personal finance, regulatory compliance, and financial planning, offering a well-rounded foundation that can cover most of the finance content in the web.

- **Selected Topics.** We select *12* topics that are cover most of books in finance. *5* of them are from *business* areas, including *business*, *Accounting*, *Accounting*, *Management*, *Marketing*, *Trading*; *1* is from *Mathematics*, i.e., *Mathematical Economics*;

Capability	Domain	Task	Dataset	Size	Reference
Tasks	Finance	Relation Cls.	FingptFinred	27,600	Sharma et al. (2022)
		NER	FingptNERCls	13,500	Yang et al. (2023)
			FingptNER	511	Alvarado et al. (2015)
		Headline Cls.	FingptHeadline	82,200	Sinha et al. (2020)
		Sentiment Cls.	SentimentCls	47,600	Yang et al. (2023)
			SentimentTra	76,800	Yang et al. (2023)
		Summariz.	TradeTheEvent	258,000	Zhou et al. (2021)
IF/Chat	General	IF/Chat	SelfInstruct	82,000	Wang et al. (2022)
			SlimOrca	518,000	Lian et al. (2023)
			UltraChat	774,000	Ding et al. (2023)
			ShareGPT	100,000	Link
	Finance	QA	FinanceInstruct	178,000	Link
			FingptConvfinqa	8,890	Chen et al. (2022)
			FlareFinqa	6,250	Chen et al. (2021)
			FlareFiqa	17,100	Yang et al. (2023)
Reasoning	Math	QA	OrcaMath	200,000	Mitra et al. (2024)
			MetaMathQA	395000	Yu et al. (2023)
			MathInstruct	262,000	Yue et al. (2023)
	Code	QA	MagicodeInstruct	111,000	Luo et al. (2023)
	Finance	CFA Exam	**Exercise**	2,950	-
Total				3,161,401	

Table D.2: Summary of our curated prompts. New datasets released with FINDAP are color-highlighted for emphasis. For datasets without formal references but only a URL, we provide their links.

and *4* are from *Economy* area, including *economy*, *econometrics*, *investing*, and *markets*. We crawled the web to downloaded the books from the corresponding topics. For CFA, we use the material provided by CFA prep providers.

• **Prompt for Filtering the Text.** We explored various prompt formats to automatically extract an financial score using an LLM and found that the additive scale by Yuan et al. (2024) worked best. Figure D.1 shows the prompt we used to filter the 'low-quality' text. Specifically, this prompt allows the LLM to reason about each additional point awarded, unlike the single-rating Likert scale which fits samples into predefined boxes. Then, to avoid the LLM favoring highly technical content like academia papers, we focused on financial student level knowledge. By setting a threshold of 4 (on a scale of 0 to 5) during the filtering process, we were able to also retain some high-quality financial content.

Insturction Prompt Curation Prompts represent the diverse ways users may interact with models and serves the essential component for IT and PA. Table D.2 summarizes the prompts curation datasets. Specifically, we conduct a broad survey and source general, financial, instruction-following, and reasoning tasks from *public datasets*. To promote *diversity*, we include datasets like FlareFinQA (Chen et al., 2021), a large open QA dataset in finance, and UltraChat (Ding et al., 2023), a dataset shown to perform well for IT in the literature (Tunstall et al., 2024; Ivison et al., 2024). Additionally, we find that exercises or demonstrations from books that were curated in §D is valuable for reasoning tasks as they usually involve challenging reasonings and come with ground truth answers and sometimes even include human-written chain-of-thought (CoT) explanations.

Figure D.2 shows the prompt we used to extract exercises from books. We carefully design the prompt to extract both the question part of an exercise, which potentially include questions, scenario and exhibits, and the answer part of the exercise, which may include answer choices and solution. In books, questions and their corresponding answers can be located far apart (e.g., the questions may appear at the beginning while the solutions are provided at the end), meaning they may not be captured within the same chunk. As a result, some questions may not have corresponding extracted answers. For such cases, GPT-4o's generated answers are used when converting the prompt into instruction-following or preference-alignment data.

E FinEval

With the breakdown of capabilities in §3.1, our evaluation framework consists of a suite for assessing these capabilities using development sets and unseen (held-out) evaluation sets. Our development set is directly split from the training data at each stage. Table E.1 outlines the capabilities and the evaluation benchmarks selected to cover these capabilities. Crucially, we did not examine scores on our unseen set while developing the models, which allows us to observe how much we may have overfitted to particular evaluations in our deci-

Capability	Domain	Task	Evaluation Dataset	Size	Reference
			Unseen - Similar		
Tasks	Finance	Sentiment Analysis	FPB	970	Malo et al. (2014)
			FiQA SA	235	Maia et al. (2018)
		Monetary policy Stance	FOMC	496	Shah et al. (2023)
		Named entity recognition	NER	98	Alvarado et al. (2015)
		Abstractive Summarization	EDTSUM	2,000	Zhou et al. (2021)
Total				3,799	
			Unseen - Novel		
Concept	General	Knowledge Recall	MMLU	14,042	(Hendrycks et al., 2021)
			AI2-ARC	3,548	Clark et al. (2018)
			Nq-open	7,842	Kwiatkowski et al. (2019)
	Finance		**MMLU-Finance**	1,460	-
Tasks	Finance	Extractive Summarization	Flare-ECTSUM	495	Mukherjee et al. (2022)
		ESG Issue Classification	MLESG	300	Chen et al. (2023a)
		Rumour Detection	MA	500	Yang et al. (2020)
		Stock Movement Prediction	SM-Bigdata	1,470	Soun et al. (2022)
			SM-ACL	3,720	Xu and Cohen (2018)
			SM-CIKM	1,140	Wu et al. (2018)
		Fraud Detection	CRA-CCF	2,280	Feng et al. (2024)
			CRA-CCFraud	2,100	Feng et al. (2024)
		Credit Scoring	Flare-German	200	Hofmann (1994)
			Flare-Astralian	139	Quinlan (1987)
			CRA-LendingClub	2,690	Feng et al. (2024)
		Distress Identification	CRA-Polish	1,740	Feng et al. (2024)
			CRA-Taiwan	1,370	Feng et al. (2024)
		Claim Analysis	CRA-ProroSeguro	2,380	Feng et al. (2024)
			CRA-TravelInsurance	2,530	Feng et al. (2024)
		Tabular QA	Flare-TATQA	1,670	Zhu et al. (2021)
		Open QA	Finance Bench	150	Islam et al. (2023)
IF/Chat	General	Precise IF	MT-bench	80	Zheng et al. (2023)
Reasoning	Math	Reasoning	MathQA	2,985	Amini et al. (2019a)
	General	Social Reasoning	Social-IQA	2,636	Welbl et al. (2017)
		Common Reasoning	Open-book-qa	500	Mihaylov et al. (2018)
			Hellaswag	10,003	Zellers et al. (2019)
			Winogrande	1,767	Sakaguchi et al. (2019)
			PIQA	3,000	Bisk et al. (2020)
	Finance	Exam	CFA-Easy	1,030	Link
			CFA-Challenge	90	-
Total				91,872	

Table E.1: Summary of our evaluation dataset. New datasets released with FinDAP are color-highlighted for emphasis.

sions around training recipe. For the unseen tasks (Table E.1), we manually review each individual dataset and have the following considerations.

• **Benchmarking tasks.** Corresponding to the capabilities, we consider a diverse set of benchmarking tasks. For *concepts*, we include knowledge tasks in the general domain, such as AI2-ARC (Clark et al., 2018), as well as in finance, such as MMLU-Finance (Hendrycks et al., 2021). For *tasks*, we consider general tasks, such as Social-IQA (Welbl et al., 2017), and domain-specific tasks, such as MLESG (Chen et al., 2023a). Notably, we intentionally include a few financial tasks such as Flare-TATQA (Zhu et al., 2021) and SM-Bigdata (Soun et al., 2022) that require understanding of tabular data, as this data format is common in this domain. For *IF/Chat capabilities*, we utilize popular instruction-following benchmarks, such as MT-Bench (Zheng et al., 2023). For *reasoning*, we include general reasoning tasks, such as MathQA (Amini et al., 2019a) and Hellaswag common sense reasoning (Zellers et al., 2019), as well as domain-specific reasoning tasks, such as CRA-ProroSeguro claim analysis (Feng et al., 2024). We also construct a new benchmark on CFA-Challenge based on CFA Level III, one of the most challenging financial exams that requires comprehensive reasoning (Khamnuansin et al., 2024; Callanan et al., 2024).

• **Evaluation method.** We split our evaluation set into two types based on their exposure to *Instruction tuning (IT) data* (Table E.1). The first type, *Similar*, includes tasks whose types have been encountered during training, even if the specific tasks themselves are unseen (e.g., a new NER task). The second type, *Novel*, includes tasks whose types have not been seen during training, representing entirely new challenges for the model (e.g., stock movement prediction). We use two different evaluation methods based on the nature of the benchmarks. For knowledge and NLP tasks (e.g., NER), we employ a straightforward *direct answer* evaluation. For reasoning tasks (e.g., CFA-Challenge), we use a *0-shot chain-of-thought (CoT)* (*Wei et al., 2023*) *answer* evaluation to enhance the reliability of our evaluation. This also exposes the reasoning path, allowing us to investigate the causes of incorrect answers and enabling a more fine-grained comparison across different models.

F Preliminary Experiments on Older Financial LLMs

As mentioned in Section 4, the reason we did not include older financial LLMs is that they are either not publicly available (e.g., Bloomberg GPT) or clearly worse than our model. As a result, we only include the SoTA finance LLM (i.e., Palmyra Fin 70b) in the comparison.

To further support this point, we compare performance on overlapping evaluation benchmarks, using the reported numbers for other baselines extracted from their papers. We made careful efforts to ensure comparability:

• **Metrics.** We noticed that different metrics were used across baselines and our methods. For example, some baselines reported F1 scores for FPB and FiQA SA, while we originally reported accuracy. For NER, the baselines used Entity F1, whereas we initially reported ROUGE scores. To ensure fair comparison, we re-ran our evaluation using the same metrics. We reported both accuracy and F1 for FPB and FiQA SA, and used Entity F1 for NER.

• **Test Datasets.** The test datasets are the same. We follow the datasets used in Xie et al. (2023)[6], which include 235 test samples for FiQA SA, 970 samples for FPB, and 98 samples for NER. These statistics also match those reported in Table E.1 of our Appendix. We do not use the training or validation sets, as our evaluation is conducted purely in the zero-shot setting. The baseline results are taken directly from Table 5 in Xie et al. (2023), which ensures consistency in comparison and also corresponds to Table 1 in Xie et al. (2024b).

Table F.1 shows the results. These results clearly show that our model outperforms these older financial LLMs, including significantly larger models such as FinMA 30B. Moreover, their reported results are based on few-shot settings, whereas our evaluations are conducted in the zero-shot setting, further highlighting the effectiveness of our approach.

[6] https://huggingface.co/collections/TheFinAI/english-evaluation-dataset-658f515911f68f12ea193194

Dataset	Metric	Llama-Fin 8b	Bloomberg GPT	FinPythia 7B	FinMA 7B	FinMA 30B
FPB	Acc	**91.13**	—	59.90	86.00	87.00
	F1	**91.28**	51.07	64.43	86.00	88.00
FiQA SA	Acc	**95.32**	—	52.34	84.00	87.00
	F1	**95.39**	75.07	53.04	—	—
NER	EntityF1	**77.09**	60.82	48.42	75.00	62.00

Table F.1: Experiments on older baselines.

G Summary of the Final Recipe and Hyper-parameters

H GenRM Prompt Details

In Figure 2, we simplified the prompt for GenRM for the purpose of illustration. In this section, we give full detailed of the prompt for Final Answer Preference (FAP) and Stepwise Corrective Preference (SCP) in Figure H.1 and Figure H.2, respectively.

Final Recipe for Llama-Fin

Continual Pre-training (CPT) and Instruction Tuning (IT)		
Data	50% CPT, 50% IT	
Curriculum	**Group 1**	**CPT:** 50% Domain-specific Text (Web and book), 50% General text (verfiable text) **IT:** 20% Domain-specific tasks, 80% General tasks
	Group 2	**CPT:** Group 1 data + domain-specific books **IT:** Group1 + Exercises extracted from books
Steps		**Group 1:** 3.84B tokens; **Group 2:** 1.66B tokens (8,000 context length, 16 A100)
Model	**Intialization**	Llama3-8b-instruct
	Attention	**CPT:** full attention with cross-document attention masking **IT:** attention with instruction mask-out and cross-document attention masking
Optim.		AdamW (weight decay = 0.1, β_1=0.9, β_2=0.95)
	LR	**Group 1:** 5e-6 with 10% warmup; **Group 2:** 5e-6 with 50% warmup
	Batch size	128K tokens
Stop Cri.	Loss of development set stops decreasing ($\approx$ 1 epoch)	
Preference Alignment (PA)		
Data	FAP and SCP	
Steps	24.58 M tokens	
Model	**Initialization**	CPT+IT
	Loss	DPO with an additional negative log-likelihood term
	Attention	Attention with instruction mask-out and cross-document attention masking
Optim.	**LR**	5e-7 with 10% warmup
	Batch size	32K tokens
Stop Cri.	Loss of development set stops decreasing	

Table G.1: Final recipe of Llama-Fin. The joint training of CPT and IT is structured into two groups, with each group undergoing joint training sequentially. The second group utilizes higher-quality data (sourced from books), following the typical curriculum training practice (Gao et al., 2024). For PA, we employ a modified DPO loss with an additional negative log-likelihood term, similar to Pang et al. (2024), as it has shown to be more effective than relying solely on the original DPO loss.

Prompt for Filtering the Text

Below is an extract from a text book. Evaluate whether the book has a high financial value and could be useful in an financial setting for teaching financial students using the additive scoring system described below. Points are accumulated strictly based on the satisfaction of each criterion:

- Add 1 point if the extract provides educational value for financial students whose goal is to learn financial concepts or take financial exams. It is acceptable if quizzes are not included; however, if quizzes are present, detailed solutions and explanations must also be provided.
- Add another point if the extract addresses certain elements pertinent to finance and aligns closely with financial standards. It might offer a superficial overview of potentially useful topics or present information in a disorganized manner and incoherent writing style.
- Award a third point if the extract is appropriate for financial use and introduces key concepts relevant to financial curricula. It is coherent and comprehensive.
- Grant a fourth point if the extract is highly relevant and beneficial for financial learning purposes for a level not higher than financial students, exhibiting a clear and consistent writing style. It offers substantial financial content, including exercises and solutions, with minimal irrelevant information, and the concepts aren't too advanced for financial students. The content is coherent, focused, and valuable for structured learning.
- Bestow a fifth point if the extract is outstanding in its financial value, perfectly suited for teaching either at financial students. It follows detailed reasoning, the writing style is easy to follow and offers profound and thorough insights into the subject matter, devoid of any non-financial or complex content.

The extract: `<EXAMPLE>`.

After examining the extract, You will output a json object containing the following 2 fields:

```
{
    "Justification": string // Briefly justify your total score, up to 100
       words.

    "Score": integer // Conclude with the score
}
```

Figure D.1: Prompt for filtering the text

Prompt for Extracting Exercise from Book

You are an educational assistant aims to extract all questions from the provided material. Look for specific indicators such as "example," "quiz," "questions," or similar terms to identify where the questions are located. If the material includes scenarios or exhibits, must include all details related to them. Do not create or derive any questions or come up with content on your own—strictly extract what is present in the material. Make sure no question is missed. If one scenario or exhibits corresponds to multiple questions, duplicate the scenarios and exhitbits so that the number of questions match the number of scenarios and exhibits.

The material: `<MATERIAL>`.

After performing these tasks, You will output a json object containing the following fields:

```
{
  "Justification": "string", // A brief justification for your extractions,
     up to 100 words.

  "Questions": "string", // A list of questions extracted from the material.
     Only extract the exact questions presented in the text.

  "Scenario": "string", // A list of scenarios corresponding to the above
     questions.  If the material does not provide the scenario place "N/A."
     Do not do any derivation or reference, must output the exact same,
     detailed and complete scenarios. The scenario may contain multiple
     paragraphs or even splited by the exhibits, combine them into one string
     . The scenario can be long, you may modify it to make it shorter,  but
     must not change its meaning.

  "Exhibit": "string", // A list of exhibits or tables corresponding to the
     above questions. If the material does not provide the exhibit, place "N/
     A." Do not do any summary, or derivation or cutting,  must output the
     exact same, detailed and complete exibits. There may be multiple
     exhibits involved in a scenario, combine them into one string. The
     exhibit can be long, you may modify it to make it shorter. Must keep the
      table format

  "Answer Choices": "string", // A list of answer choices corresponding to the
      above questions. If the material does not provide answer choices, place
      "N/A."

  "Answer": "string" // A list of answers corresponding to the above questions
     . Answers should only be included if provided in the material. If no
     answer is given, place "N/A." If explanations or reasoning steps or
     equations are included, must capture all of them. Must not answer it
     yourself if there is no answer provided in the material. Make sure the
     final number of questions equals to number of scenario equals to number
     of exhibits equals to number of answers
}
```

Figure D.2: Prompt for extracting exercises from books

Prompt for FAP

You given a question, a reference answer and a proposed answer, you task is to determine the correctness of the proposed answer. First, extract the final answer (for example, A, B or C) from the reference answer. Second, extract the final answer from the proposed answer (for example, A, B or C). Finally, compare the two final answer to determine the correctness. Do not do any extra reasoning, must determine the correctness soley based on the given reference and proposed answer.

Question: `<QUESTION>`
Reference Answer: `<REFERENCE>`
Proposed Answer: `<PROPOSAL>`

After performing these tasks, You will output a json object containing the following fields:

```
{
  "Justification": "string", // A brief justification for your output,
  up to 100 words.

  "Correctness": "string", // If the proposed answer has the same final final
      answer as the reference answer (for example, both choose A or have the
      same answer),  output 'correct'.  Put 'wrong' to all other cases. For
      example, if the proposed answer has a different final answer comparing
      to the reference answer, put 'wrong'. If the proposed answer does not
      explicitly give a final answer to the question, put 'wrong'. If the
      proposed answer gives more than one final answer to the question, put '
      wrong'.
}
```

Figure H.1: Prompt for FAP

Prompt for Prompt for SCP

Given a question, a reference answer and an incorrect answer, you task is to identify the first incorrect step from the incorrect answer. The "first incorrect step" means all reasoning up to that point is accurate, but the error begins at this specific step.

Question: `<QUESTION>`
Reference Answer: `<REFERENCE>`
Incorrect Answer: `<INCORRECT>`

After performing these tasks, You will output a json object containing the following fields:

```
{
  "Justification": "string", // A brief justification for your output,
  up to 100 words.
  You need to explain
  (1) why the identified first incorrect step is incorrect;
  (2) why the reasoning up to this specific step is correct and
  (3) how the corrected step resolves the issue, aligning with the reference
      answer,
  maintaining the logical flow and progressing to the final answer.

  "First incorrect step": "string", // The explanation in the incorrect answer
       consists of multiple reasoning steps. Please identify the first
      incorrect reasoning step. It should be a piece of text directly and
      exactly quoted from the incorrect answer. It should be an intermediate
      step rather than the final answer

  "Reasoning up to incorrect": "string", // From the incorrect answer, give
      the correct reasoning steps up to the first incorrect step. This should
      be directly and exactly quoted from the incorrect answer.

  "Step correction": "string", //  Replace the identified incorrect step with
      a single, clear, and correct step. This step should directly address and
       correct the error, explicitly providing the correct reasoning without
      requring for more information or challenging the question. It should
      effectively answer the question, "What is the next reasoning step?"
      given on the question and the identied "Reasoning up tp incorrect". It
      should help progress to the final answer.

}
```

Figure H.2: Prompt for SCP