Teaching LLMs How To Learn with Contextual Fine-Tuning

Younwoo (Ethan) Choi[*]
University of Toronto
ywchoi@cs.toronto.edu

Muhammad Adil Asif[*]
Vector Institute
adil.asif@vectorinstitute.ai

Ziwen Han[†]
University of Toronto
hanziwen@cs.toronto.edu

John Willes
Vector Institute
john.willes@vectorinstitute.ai

Rahul G. Krishnan
University of Toronto
rahulgk@cs.toronto.edu

Abstract

Prompting Large Language Models (LLMs), or providing context on the expected model of operation, is an effective way to steer the outputs of such models to satisfy human desiderata after they have been trained. But in rapidly evolving domains, there is often need to fine-tune LLMs to improve either the kind of knowledge in their memory or their abilities to perform open ended reasoning in new domains. When human's learn new concepts, we often do so by linking the new material that we are studying to concepts we have already learned before. To that end, we ask, "can prompting help us *teach LLMs how to learn*". In this work, we study a novel generalization of instruction tuning, called *contextual fine-tuning*, to fine-tune LLMs. Our method leverages instructional prompts designed to mimic human cognitive strategies in learning and problem-solving to guide the learning process during training, aiming to improve the model's interpretation and understanding of domain-specific knowledge. We empirically demonstrate that this simple yet effective modification improves the ability of LLMs to be fine-tuned rapidly on new datasets both within the medical and financial domains.

1 Introduction

Large Language Models (LLMs) demonstrate impressive performance on a wide range of downstream tasks without explicit supervision [28]. With increasing scale, these models also develop emergent capabilities such as multi-step reasoning, instruction following, and program execution [47, 8]. LLMs are trained via a three-step process: pretraining (which results in a base model) to compress knowledge over a vast text corpus, supervised finetuning for instruction following, and aligning with human values using a variety of alignment algorithms [30, 24, 35, 10]. This three-step process produces an open-ended chatbot that demonstrates two abilities: (a) *reasoning:* the ability to process and manipulate textual information based on open-ended natural language text, and (b) *recall:* the ability to recall information in training data. These two abilities are linked in that they both are learned from text data and emerge as a consequence of model size simultaneously. The former

[*]Equal contribution
[†]Work conducted while at the Vector Institute

38th Workshop on Fine-Tuning in Machine Learning (NeurIPS 2024).

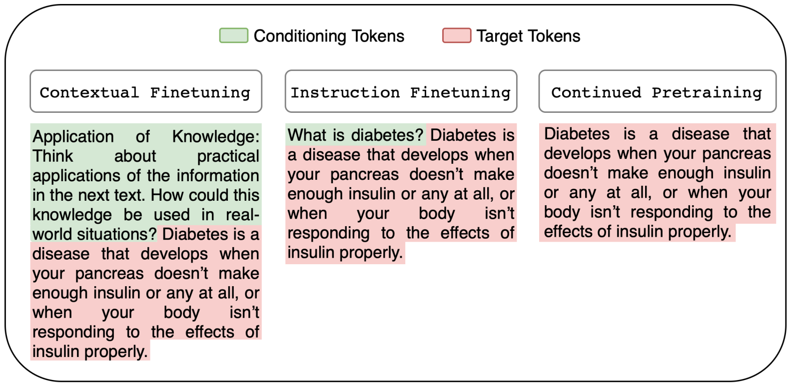


Figure 1: The figure illustrates the distinct approaches of Contextual Fine-Tuning (CFT), Instruction Fine-Tuning (IFT), and Continued Pretraining (CPT). In CFT, a contextual prompt is highlighted in green *"Think about practical applications of the information in the next text. How could this knowledge be used in real-world situations?"* followed by the main text. IFT employs a direct instruction *"What is diabetes?"* before presenting the same text. In contrast, CPT displays only the main text without any preceding prompts or instructions. The key difference lies in CFT's use of contextual prompts that guide the model's semantic understanding and reasoning, whereas IFT relies on explicit instructions to elicit specific responses. CPT, lacking both prompts and instructions, focuses solely on processing the main content.

enables the model to infer user intent based on prompts enabling LLMs to think step by step [22], or chain steps of reasoning together [48], the latter serves as a snapshot of the model's memory.

LLMs remain unaware of information and events occurring after their knowledge cutoff; in fast-moving domains and in scenarios where deployment requires knowledge of up-to-date information, there is a need to remedy this limitation. There are two popular approaches to this problem. The first is to increase the context length of the model until all anticipated new information fits within this context (the largest of which is Google's Gemini-1.5 model with a context length of two million tokens). However, even context lengths this large can be exhausted and it is unclear whether the model's attention mechanism is capable of accurately inferring signal regardless of where it is in the context. The alternate approach uses external knowledge stores via retrieval augmented systems [25]. This approach works well when the reasoning abilities already learned by the model suffice to process and extract the relevant information. But gradient-based learning remains vital in scenarios where there is a need to teach the model how to manipulate new tools or learn new strategies for reasoning.

The simplest approach to update model knowledge via finetuning is to continue pretraining the base model. Unfortunately, the data and training procedure necessary to replicate the additional finetuning and alignment phases are rarely open-sourced in chat-based models. Consequently, the general practice is to finetune the aligned model with new domain-specific knowledge. Models that have undergone instruction finetuning and alignment training are more amenable to interacting with users but are harder to update with new knowledge. But training to update the knowledge can result in catastrophic forgetting of knowledge gained during pretraining, or loss of capabilities like instruction-following and task-solving [43].

Our approach is inspired by the capabilities of LLMs to leverage prompts in question answering. For example, few-shot prompting popularized by Brown et al. [4] performs well on a variety of unseen tasks at prediction time. Wei et al. [48] investigated how chain-of-thought (CoT) prompting can significantly improve a model's ability to perform complex multi-step reasoning. Wang et al. [44] further improved on CoT by selecting the most consistent answer from a diverse set of sampled reasoning paths.

Our work investigates a simple question: *can prompting improve the efficacy of LLM fine-tuning?* We argue yes, and to this end, we propose a new method for fine-tuning that blends in-context learning with gradient-based learning. In summary, our contributions are as follows: We present *contextual finetuning*, a generalization of instruction tuning, that combines in-context learning and fine-tuning. We further investigate the gradients provided by the additional context and provide synthetic experiments demonstrating their effectiveness for fine-tuning. To study the impact of our method, we create two datasets in the biomedical domain: the first consisting of 121,489 journal articles from 37 diverse topics in biology and medicine, and second comprising 30 open-source medical textbooks (see Appendix B for more). We show that contextual finetuning can be used to update a model's knowledge more efficiently than continued pretraining and instruction tuning.

We show increased performance on both real-world dataset and Q&A tasks while using carefully constructed synthetic data to better understand where performance gains arise from.

2 Related Work

Instruction Tuning The common paradigm used in training instruction aligned ("chat") LLMs involves three steps: pretraining on unlabelled corpora, performing instruction tuning, followed by reward-based preference training, as used in Ouyang et al. [30]. The instruction tuning phase is generally used as a first step to aligning LLMs to human instructions or when access to human-labeled preference data is limited. Instruction tuning significantly narrows the divide between models' traditional next-word prediction objectives and the practical need for models to adhere to explicit human instructions. Wei et al. [46] have highlighted how this approach markedly boosts zero-shot performance across previously unseen tasks, underlining its effectiveness. Earlier work [46, 17] have proposed using a large set of instructions for NLP tasks, while more recent findings [45, 40, 32, 52] have found success using increasingly smaller and higher quality instruction datasets on open sourced pretrained models such as LLaMA [41]. Notably, as Gudibande et al. [13] discovers, training on instructions in the instruction tuning phase does not improve the underlying capabilities of the models; these models merely imitate the instruction following template.

Instruction tuning has input-output pairs (x, y) that are data point specific (e.g., x = "Who is the current president of the United States?", y = "Joe Biden"). Within instruction tuning, there is a specific, narrow question for which there exists a *right* answer that the model is expected to identify. In contextual fine-tuning, our intent is to pair y with a randomly sampled contextual prompt x which serves as guidance for the model to learn the most important information. x can be specific or general desiderata useful for learning intended to prime the model to contextualize and incorporate the knowledge in y within its parameters.

Domain-Specific Training While pretraining on trillions of unlabelled tokens creates generalist foundation models [3, 41, 42, 1, 28], injecting domain-specific expertise into models while retaining the generalist remains an active front of research. Improving the underlying capabilities of LLMs is a more difficult challenge than simply aligning to instructions, in part due to the much larger dataset requirement. While smaller LLMs are capable of outperforming the larger monoliths in specific domains [15, 26], or pushing language modeling in a simplified domain to the extreme [9].

AdaptLLM [5] propose a continued pretraining method on domain-specific corpora which transforms raw corpora into reading comprehension texts by enriching each text with content-related tasks, akin to human learning through reading comprehension practice. Our work aligns with this line of research by focusing on methods that enhance domain-specific capabilities. Unlike AdaptLLM, which enriches the training data with content-related tasks, we propose *Contextual Fine-Tuning* (CFT), a method that incorporates contextual prompts during fine-tuning to guide the model's learning process.

3 Methodology

Notation & Background We consider a large language model (LLM) P_θ, parameterized by pretrained weights θ. We have access to a domain-specific corpus $\mathcal{D}_{train}^{raw}$ consisting of sequences of tokens. Our objective is to fine-tune the model to obtain new parameters θ' that enhance performance on domain-specific downstream tasks, evaluated on a test set $\mathcal{D}_{test}$.

Continued Pretraining Continued pretraining (CPT) leverages large volumes of unlabeled domain-specific data to refine the model's understanding of the domain. Given sequences of tokens $x = (x_1, x_2, \ldots, x_n)$ sampled from $\mathcal{D}_{train}^{raw}$, the model is trained using the causal language modeling objective, which predicts the next token given the previous tokens.

$$\mathcal{L}_{CPT}(\theta) = -\mathbb{E}_{x\sim\mathcal{D}_{train}^{raw}} \sum_{k}^{n} \log P_\theta(x_k \mid x_{<k}). \tag{1}$$

where $x_{<k} = (x_1, x_2, \ldots, x_{k-1})$ represents the sequence of tokens preceding x_k.

Instruction Fine-tuning Instruction fine-tuning utilizes a collection of instruction-response pairs (x, y) sampled from a dataset $\mathcal{D}_{train}^{IFT}$. Here, x is an instruction or prompt, and y is the corresponding response. The model is trained to generate the response y conditioned on the instruction x.

$$\mathcal{L}_{IFT}(\theta) = -\mathbb{E}_{(x,y)\sim\mathcal{D}_{train}^{IFT}} \sum_{k}^{m} \log P_\theta(y_k \mid x, y_{<k}). \tag{2}$$

where $y = (y_1, y_2, \ldots, y_m)$ and $y_{<k} = (y_1, y_2, \ldots, y_{k-1})$.

3.1 Contextual Fine-Tuning

We introduce Contextual Fine-Tuning (CFT), a method that incorporates contextual prompts into the training process to guide the model's learning in a domain-specific manner. Inspired by constructivist learning theory [33], which emphasizes active engagement and thoughtful processing for effective learning, we hypothesize that contextual prompts can enhance the model's ability to internalize and reason about new concepts within the domain.

Designing contextual prompts We define a set of contextual prompts $\mathcal{C} = \{c^{(1)}, c^{(2)}, \ldots, c^{(L)}\}$, where each prompt $c^{(l)} = (c_1^{(l)}, c_2^{(l)}, \ldots, c_{n_l}^{(l)})$ is a sequence of tokens for some length n_l, designed to guide the model during training. These prompts mimic effective human learning strategies by encouraging the model to engage in various cognitive processes such as focusing on key concepts, critical analysis, and application of knowledge.

We select 10 prompts to provide a diverse yet manageable set of learning strategies to balance between offering sufficient variation to cover different cognitive approaches and maintaining practicality in training. We present two of these prompts in the main text and include the remaining eight in Appendix A.1. Each prompt is grounded in established educational theories, as detailed below:

1. Focus on Key Concepts: This prompt aligns with Sweller [39], which emphasizes the importance of reducing unnecessary cognitive load to facilitate learning. By focusing on essential information, learners can allocate their cognitive resources more effectively.
    - *"Concentrate on understanding the core principles and essential facts in the following text. Pay special attention to definitions, examples, and conclusions."*
2. Contextual Understanding: Piaget [33] suggests that learners build new knowledge upon the foundation of their existing understanding by making connections between new and prior information.
    - *"As you read the next passage, relate its content to its broader context and implications. Think about how this information connects to what you've learned previously."*

These prompts are designed to engage the model in a manner akin to how a human learner interacts with educational material, thereby fostering a deeper and more nuanced understanding of the domain-specific content. It is important to note that these prompts are not necessarily optimal; rather, their effectiveness lies in the semantic functionality they provide. Future work may involve augmenting or refining these prompts to better suit specific applications or domains, enhancing their ability to guide the model's learning process effectively.

Learning with contextual prompts For each training example, we integrate a contextual prompt to guide the model's focus. The procedure is as follows:

1. **Sampling:** Given a domain-specific text sequence $x = (x_1, x_2, \ldots, x_n)$ sampled from $\mathcal{D}_{train}^{raw}$ and we randomly select a contextual prompt c from $\mathcal{C}$.
2. **Input Construction:** We prepend the prompt $c = (c_1, c_2, \ldots, c_m)$ to the text sequence x to form the new input sequence: $x' = (c_1, c_2, \ldots, c_m, x_1, x_2, \ldots, x_n)$.
3. **Training Objective:** The model is trained to predict the tokens in x, conditioned on both the prompt c and the preceding tokens in x. The loss function for CFT is defined as:

$$\mathcal{L}_{CFT}(\theta) = -\mathbb{E}_{x\sim\mathcal{D}_{train}^{raw}, c\sim\mathcal{C}} \sum_{k=1}^{n} \log P_\theta(x_k \mid c, x_{<k}). \tag{3}$$

Refer to Algorithm 1 in Appendix A.2 for the detailed algorithm. We hypothesize that by incorporating contextual prompts during training, we can influence the model's learning trajectory, aligning the gradients towards more semantically meaningful representations. These gradients guide the optimization process, encouraging the model to develop a deeper understanding of the content.

4 Understanding Contextual Finetuning with Synthetic Data

Analyzing gradients in large language models (LLMs) is infeasible due to added complexities because of their billions of parameters and long context lengths. To gain insight into how contextual prompts affect training gradients, we conduct a synthetic experiment using a simplified model. Inspired by the framework of Garg et al. [11], we investigate how contextual prompts influence a model's capacity to learn a class of functions through different fine-tuning strategies.

Setup We review the setup in Garg et al. [11] and explain how we modify it. Consider a function class $\mathcal{F}$, and our goal is to train a model that can learn functions $f \in \mathcal{F}$ such that, for most functions, the model can approximate $f(x_{\text{query}})$ for a new query input x_{query} by conditioning on a prompt sequence containing in-context examples. Formally, let $\mathcal{D}_{\mathcal{X}}$ be a distribution over inputs, and let $\mathcal{D}_{\mathcal{F}}$ be a distribution over functions in $\mathcal{F}$.

Now, consider learning a new class of functions $\mathcal{G}$, where each $g \in \mathcal{G}$ is a composition of f with another function h from a distribution $\mathcal{D}_{\mathcal{H}}$, that is: $\mathcal{G} = \{g \mid g(x) = h(f(x)), h \in \mathcal{D}_{\mathcal{H}}\}$. We can draw an analogy between this setup and the fine-tuning of LLMs in specific domains. In this analogy, texts from medical textbooks can be viewed as samples from some distribution $\mathcal{D}_{\mathcal{X}}$ over inputs, and the function class $\mathcal{F}$ represents the LLM's ability to process and understand these texts. Learning a new function class $\mathcal{G}$ corresponds to adapting the model to perform specific tasks in the biomedical domain, such as extracting diseases from electronic health records or answering medical questions. If the model already has the capability to compute $f(x)$ (i.e., process and understand the text), this can aid in learning the composed function $g(x) = h(f(x))$.

Pretraining We first train a model to learn the function class $\mathcal{F}$ with respect to the distributions $\mathcal{D}_{\mathcal{F}}$ over functions and $\mathcal{D}_{\mathcal{X}}$ over inputs. We construct random training prompts P which is a sequence $P = (x_1, f(x_1), \ldots, x_k, f(x_k))$, where the inputs x_is and are drawn independently from $\mathcal{D}_{\mathcal{X}}$, and f is drawn from $\mathcal{D}_{\mathcal{F}}$. We then train a model to predict every $f(x_i)$ based on a set of preceding in-context examples. Specifically, let P^i denote the prompt prefix containing i in-context examples and the $(i+1)$-th input $P^i = (x_1, f(x_1), x_2, f(x_2), \ldots, x_i, f(x_i), x_{i+1})$, we train a transformer model M_θ by minimizing the expected loss over all the prompt prefixes:

$$\min_{\theta}, \mathbb{E}_P \left[\frac{1}{k+1} \sum_{i=0}^{k} \ell\left(M_\theta(P^i), f(x_{i+1})\right) \right] \tag{4}$$

where ℓ is the mean squared error loss. In our experiment, $\mathcal{F}$ is the class of linear functions, that is, $\mathcal{F} = \{f \mid f(x) = w^\top x, w \in \mathbb{R}^d\}$, where the weight vectors w are sampled from $\mathcal{N}(0, I_d)$. We let $\mathcal{D}_{\mathcal{X}}$ be the isotropic Gaussian distribution $\mathcal{N}(0, I_d)$. Garg et al. [11] show that after sufficient training, a transformer model can predict $f(x_{\text{query}})$ almost perfectly when there are more than 20 in-context examples.

Fine-Tuning We now extend their setup to fine-tuning the pretrained transformer to learn a novel function class $\mathcal{G}$. We consider three types of functions $h(\cdot)$ to construct $\mathcal{G}$:

1. Nonlinear activation: $\mathcal{G} = \{g \mid g(x) = \text{ReLU}(f(x))\}$.
2. Polynomial combination: $\mathcal{G} = \{g \mid g(x) = f(x) + f(x)^2\}$.
3. Multiple linear relationships: $\mathcal{G} = \{g \mid g(x) = f(x) + w_2^\top x, w_2 \in \mathbb{R}^d\}$.

We fine-tune the pretrained transformer on these different function classes separately, using different training strategies: Contextual Fine-Tuning (CFT), Continued Pretraining (CPT), and Negative Contextual Fine-Tuning (NEG-CFT) which is an ablation of CFT with negative contextual prompts intended to provide non-helpful or potentially misleading information. We now describe how we construct the input prompts for the different fine-tuning strategies:

- **CPT**: We fine-tune the model on prompts that contain only the inputs x_i and their outputs computed using the composed function $g(x)$, specifically:

$$P_{CPT} = (x_1, g(x_1), x_2, g(x_2), \dots, x_k, g(x_k))$$

- **CFT**: We provide the model with additional contextual information by including the original function outputs $f(x_i)$ in the prompt. The prompt structure is then:

$$P_{\text{CFT}} = (x_1, f(x_1), x_2, f(x_2), \dots, x_k, f(x_k), x_1, g(x_1), x_2, g(x_2), \dots, x_k, g(x_k)).$$

  Here, the initial sequence $(x_1, f(x_1), x_2, f(x_2), \dots, x_k, f(x_k))$ serves as a contextual prompt to help the model learn the transformation introduced by h.

- **NEG-CFT**: To assess the impact of the contextual prompts, we introduce NEG-CFT, where we replace the original function outputs $f(x_i)$ with random values sampled uniformly from $[0, 1]$. The prompt becomes:

$$P_{\text{NEG-CFT}} = (x_1, r_1, x_2, r_2, \dots, x_k, r_k, x_1, g(x_1), x_2, g(x_2), \dots, x_k, g(x_k)),$$

  where $r_i \sim \mathcal{U}(0, 1)$. This ablates the meaningful contextual information to evaluate its significance in learning the function class.

For each fine-tuning strategy, we minimize the loss in Equation 4 using the respective prompts and $g(x_{i+1})$ instead of $f(x_{i+1})$. See Appendix C.1 for the training and model architecture details.

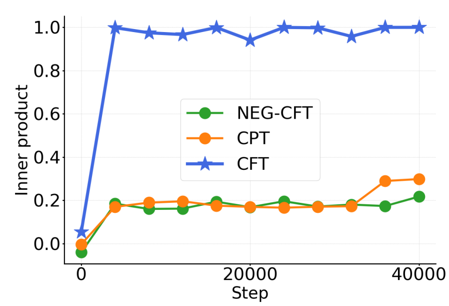
Results Our experiments demonstrate that CFT of the pretrained transformer offers advantages over CPT and NEG-CFT. Empirically, we observe 1) alignment of gradients with target functions, and 2) value provided by the tokens within the contextual prompt.

Contextual prompts help the model capture the underlying functional relationships. To delve deeper into how contextual prompts influence learning, we examine the gradients of the transformer across different training strategies. We look at the case where the transformer's input is of the form $P = (x_1, f(x_1), \dots, x_k, f(x_k), x_{query})$, its output aims to approximate $f(x_{query})$. Consequently, the gradient of the transformer's output with respect to x_{query} should align with the gradient $\nabla_{x_{query}} g(x_{query})$. In our experiments, we compute the normalized inner product between the gradient of the transformer's output and the true gradient $\nabla_{x_{query}} g(x_{query})$ during training. For the polynomial combination class $\mathcal{G}$, the gradient is:

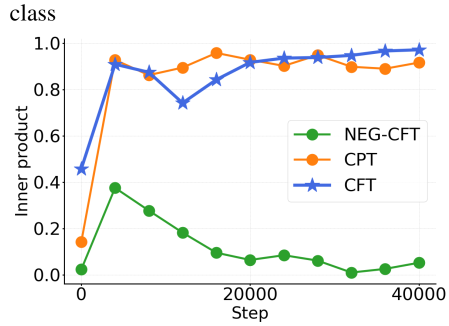
$$\nabla_{x_{query}} g(x_{query}) = w_1 + 2(w_2^\top x_{query}) w_2 \quad (5)$$

and for the multiple linear relationships class,

$$\nabla_{x_{query}} g(x_{query}) = w_1 + w_2 \quad (6)$$



(a) Learning polynomial combination function class



(b) Learning multiple linear relationships function class

Figure 2: (a) and (b) illustrate the normalized inner product between the transformer's gradients and the true gradients $\nabla_{x_{query}} g(x_{query})$, where CFT exhibits a higher alignment, approaching 1, indicating effective learning of the target functions.

Figure 2a demonstrates that the gradients from the CFT-trained transformer exhibit a much higher alignment with $\nabla_{x_{query}} g(x_{query})$ compared to those from CPT and NEG-CFT. The inner product between the gradients approaches 1 for CFT, indicating near-perfect alignment. This close alignment suggests that the model's updates are effectively moving in the direction that minimizes the loss concerning the target function g. Essentially, the transformer is not only predicting $g(x)$ accurately but also capturing the underlying functional relationships due to the informative contextual prompts. Figure 2b highlights the importance of the content within the contextual prompts. Despite NEG-CFT having a similar prompt structure to CFT, the use of random or non-informative values in place of $f(x_i)$ results in gradients that do not align well with $\nabla_{x_{query}} g(x_{query})$. This misalignment indicates that the relevance and quality of the content of contextual prompts are crucial for guiding the model's learning process effectively. We provide the results for learning the class of function $\mathcal{G} = \{g \mid g(x) = \text{ReLU}(f(x))\}$ in Appendix E.2 and learning dynamics in Appendix.

5 Contextual Fine-Tuning on Financial and Medical Domain

In this section we aim to provide an overview of the datasets used to evaluate contextual prompts, and show their effectiveness especially when combined with other training schemes.

5.1 Experimental Setup

We assess the efficacy of contextual fine-tuning by comparing the performance of large language models (LLMs) when fine-tuned on domain-specific corpora using both contextual fine-tuning and standard unsupervised fine-tuning approaches. In our experimental setup, we use several configurations of the Llama-2 models to evaluate the effectiveness of contextual fine-tuning compared to standard unsupervised fine-tuning. Specifically, we employ the Llama-2 Base model with 7 billion parameters and Llama-2 Chat models with both 7 billion and 13 billion parameters, each with a sequence length of 4096. Training details are under Appendix C. The performance of LLMs is measured using the relevant downstream tasks.

Datasets. We evaluate the effectiveness of contextual fine-tuning across two distinct domains: the financial domain and the biomedical domain. For the financial domain, we use a dataset comprising 306,242 financial news articles [18]. In the biomedical domain, we utilize OpenMedText, as described in detail in the previous section. When incorporating instruction fine-tuning into our experiments, we include additional datasets specific to each domain. For the financial domain, we use FinAlpaca [12], which contains instruction-output pairs tailored for financial tasks. In the biomedical domain, we supplement with datasets from [29], providing question-answer pairs bootstrapped from the NHS encyclopedia [27]. Additionally, we incorporate UltraChat [7], a large-scale, multi-round dialogue dataset, into our instruction fine-tuning process.

Benchmarks. The effectiveness of the fine-tuning approach in each domain is evaluated using several domain-specific benchmarks. In the financial domain, we consider (1) the sentiment analysis task **FiQA** [50] where LLMs predict sentiments categorized as 'positive', 'neutral', or 'negative' in financial texts. (2) The headline classification task **MultiFin** [20, 50], where LLMs categorize each news article into one of six categories based on the headline. (3) **Causal20** [50], which involves classifying sentences extracted from financial news as either depicting a 'causal' or 'noise' relationship between financial events. For the biomedical domain, we consider the following multiple-choice question (MCQ) datasets from **Massive Multitask Language Understanding** (MMLU) [16]: (1) Anatomy, (2) Clinical Knowledge, (3) College Biology, (4) College Medicine, (5) Medical Genetics, and (6) Professional Medicine. We also use **MedQA**, a collection of multiple-choice questions from the professional medical board exams [19].

Evaluation. In our evaluation, MCQs are formatted with questions followed by several options labeled with ID symbols (e.g., `A/B/C/D`). Building on the approach outlined in Zheng et al. [51], we instruct the language models to predict an option ID symbol rather than the textual content of the answer. This method addresses a critical issue: the likelihood of the answer's text being naturally plausible could be conflated with its likelihood of being the correct response due to the model's linguistic biases. However Robinson & Wingate [36] raise concerns regarding LLMs' inherent selection biases, which highlights that these models may show a preference for specific option IDs. To counteract this bias and enhance the validity of our evaluations, we adopt a debiasing technique as prescribed in the aforementioned work. See Appendix D.1 for the detailed method.

Llama 2 7B	Accuracy (↑)							
	Anatomy	Clinical Knowledge	College Biology	College Medicine	Medical Genetics	Professional Medicine	MedQA	Average
Chat	44.07	46.79	48.61	39.02	49.00	48.90	38.96	45.05
Chat (CPT)	45.19	47.17	49.31	43.93	50.50	46.32	39.28	45.96
Chat (CFT)	**48.15**	**48.87**	**52.08**	**44.22**	**54.00**	**46.69**	**40.65**	**47.81**
Llama 2 13B	Anatomy	Clinical Knowledge	College Biology	College Medicine	Medical Genetics	Professional Medicine	MedQA	Average
Chat	51.85	56.60	54.17	46.82	**63.50**	56.99	**45.33**	53.61
Chat (CPT)	50.37	60.00	55.90	50.58	62.00	57.35	43.95	54.31
Chat (CFT)	**53.33**	**63.21**	**57.99**	**56.35**	62.50	**57.72**	44.85	**56.56**

Table 1: Medical Benchmarks (Zero-shot). The results show that the 7B model achieved a $\%\Delta^{CFT}_{CPT}$ of 1.85% and a $Rel\%\Delta^{CFT}_{CPT}$ of 203%. The 13B model demonstrated increased effectiveness with a $\%\Delta^{CFT}_{CPT}$ of 2.25% and a $Rel\%\Delta^{CFT}_{CPT}$ of 321%, indicating that CFT's impact grows with the model's scale.

Llama 2 7B	FiQA F1	Causal 20 F1	Multifin F1	Average
Chat	56.40	**90.40**	38.74	61.48
Chat (CPT)	62.53	90.16	38.23	63.64
Chat (CFT)	**67.69**	90.17	**46.01**	**67.96**

Table 2: Llama 2 7B Financial Benchmarks (Zero-shot).

Llama 2 13B	FiQA F1	Causal 20 F1	Multifin F1	Average
Chat	61.18	84.77	45.81	63.92
Chat (CPT)	66.96	**90.06**	45.33	67.45
Chat (CFT)	**70.55**	89.87	**50.94**	**70.45**

Table 3: Llama 2 13B Financial Benchmarks (Zero-shot).

5.2 Results

We evaluate the zero-shot performance of large language models (LLMs) trained with different methods across both medical and financial benchmarks. It is important to note that our primary objective is not to achieve state-of-the-art performance but to assess the relative improvements offered by contextual fine-tuning (CFT), instruction fine-tuning (IFT), and continued pretraining (CPT) compared to a baseline model. We first discuss CFT and IFT: we focus primarily on two metrics for evaluating the effectiveness of these training approaches: (1) $\%\Delta^{CFT}_{CPT}$, which denotes the performance difference between CFT and CPT, and (2) $Rel\%\Delta^{CFT}_{CPT} = \frac{\%\Delta^{CFT}_{Baseline} - \%\Delta^{CPT}_{Baseline}}{|\%\Delta^{CPT}_{Baseline}|} \times 100 = \frac{\%\Delta^{CFT}_{CPT}}{|\%\Delta^{CPT}_{Baseline}|} \times 100$, a measure that quantifies how much more effective CFT is relative to CPT in terms of improvement over the baseline.

CFT is effective across model scales. We asses how contextual fine-tuning performs across different model scales. Table 1 presents medical benchmarks for the 7B and 13B model scales which shows a $\%\Delta^{CFT}_{CPT} = 1.85\%$ and a $Rel\%\Delta^{CFT}_{CPT} = 203\%$. For the 13B model, these metrics increase to $\%\Delta^{CFT}_{CPT} = 2.25\%$ and $Rel\%\Delta^{CFT}_{CPT} = 321\%$. Similarly, Tables 2 and 3 contain financial benchmarks. The 7B model records a $\%\Delta^{CFT}_{CPT} = 4.32\%$ and a $Rel\%\Delta^{CFT}_{CPT} = 200\%$ whereas the 13B model, the results are $\%\Delta^{CFT}_{CPT} = 3\%$ and $Rel\%\Delta^{CFT}_{CPT} = 85\%$. The results demonstrate that the simple augmentation of contextual prompting can help increase performance across the board.

CFT is preferable to existing approaches for improving a model at a fixed scale. The tables in Appendix F.1 show the performance on the medical and financial benchmarks while holding the model scale constant. The base non-instruct model holds an average accuracy of 41.34% on the medical benchmarks. Our experiments find that combining training schemes provides the greatest boost in fine-tuning performance. In particular, combining CFT and IFT gives a performance boost of $\%\Delta^{CFT+IFT}_{Base} = 2.95\%$ compared to $\%\Delta^{CPT+IFT}_{Base} = 1.91\%$. Similar trends are seen in the financial benchmarks where the same combination led to an increase of $\%\Delta^{CFT+IFT}_{Base} = 36.28\%$ in F1 score. These results concretize that augmenting the CPT stage of fine-tuning to instead use CFT provides a near-free boost in performance. More detailed analyses can be found in Appendix F.1.

The semantic content of the prompts in CFT are important to improving performance. The core aspect of our study involves examining the impact of additional context on model performance, and specifically how the signal from this context provides a boost in learning performance. We conduct an ablation by introducing negative contextual prompts, which are designed to mislead the model by suggesting that the following information is incorrect. These results can be found in tables under Appendix F.2. In the financial domain, the impact of negative prompts is evident. The 7B model experienced a performance drop of $\%\Delta_{CFT}^{-CFT} = -3.41\%$, and the 13B model sees a decrease of $\%\Delta_{CFT}^{-CFT} = -2.39\%$. All models undergoing negative contextual fine-tuning still perform better than those subjected to CPT.

6 Conclusion

This study introduces contextual fine-tuning, a variation of instruction tuning, which leverages contextual gradients to guide the learning process through simple, domain-adaptive prompts. Our experiments reveal that the contextual gradients enhance performance by effectively directing model learning. Contextual fine-tuning demonstrates superior results over traditional continued domain pretraining in both financial and medical domains. Further, our ablation study shows that the specific context of the prompts critically influences performance, highlighting the importance of carefully crafted instructional content in training setups. Finally, we open-source a biomedical dataset curated from MDPI journals and other open-source medical textbooks. Overall, the findings suggest that contextual fine-tuning is a potent strategy for enhancing the domain-specific capabilities of language models, offering a promising direction for future research.

References

[1] Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

[2] B. S. Bloom, M. B. Engelhart, E. J. Furst, W. H. Hill, and D. R. Krathwohl. *Taxonomy of educational objectives. The classification of educational goals. Handbook 1: Cognitive domain*. Longmans Green, New York, 1956.

[3] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[5] Daixuan Cheng, Shaohan Huang, and Furu Wei. Adapting large language models via reading comprehension. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=y886UXPEZ0.

[6] Fergus I.M. Craik and Robert S. Lockhart. Levels of processing: A framework for memory research. *Journal of Verbal Learning and Verbal Behavior*, 11(6):671–684, 1972. ISSN 0022-5371. doi: https://doi.org/10.1016/S0022-5371(72)80001-X. URL https://www.sciencedirect.com/science/article/pii/S002253717280001X.

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

[8] Zhengxiao Du, Aohan Zeng, Yuxiao Dong, and Jie Tang. Understanding emergent abilities of language models from the loss perspective, 2024.

[9] Ronen Eldan and Yuanzhi Li. Tinystories: How small can language models be and still speak coherent english? *arXiv preprint arXiv:2305.07759*, 2023.

[10] Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

[11] Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022.

[12] Gaurang Bharti. finance-alpaca (revision 51d16b6), 2024. URL https://huggingface.co/datasets/gbharti/finance-alpaca.

[13] Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. The false promise of imitating proprietary llms. *arXiv preprint arXiv:2305.15717*, 2023.

[14] J.P. Guilford. *The Nature of Human Intelligence*. McGraw-Hill series in psychology. McGraw-Hill, 1967. ISBN 9780070251359. URL https://books.google.ca/books?id=T-ZJAAAAMAAJ.

[15] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

[16] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

[17] Srinivasan Iyer, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Daniel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, et al. Opt-iml: Scaling language model instruction meta learning through the lens of generalization. *arXiv preprint arXiv:2212.12017*, 2022.

[18] Jeet. US Financial News Articles. https://www.kaggle.com/datasets/jeet2016/us-financial-news-articles, 2018.

[19] Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *arXiv preprint arXiv:2009.13081*, 2020.

[20] Rasmus Jørgensen, Oliver Brandt, Mareike Hartmann, Xiang Dai, Christian Igel, and Desmond Elliott. MultiFin: A dataset for multilingual financial NLP. In Andreas Vlachos and Isabelle Augenstein (eds.), *Findings of the Association for Computational Linguistics: EACL 2023*, pp. 894–909, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.66. URL https://aclanthology.org/2023.findings-eacl.66.

[21] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA, 2015.

[22] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

[23] Jean Lave and Etienne Wenger. *Situated Learning: Legitimate Peripheral Participation*. Learning in Doing: Social, Cognitive and Computational Perspectives. Cambridge University Press, 1991.

[24] Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023.

[25] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021.

[26] Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023.

[27] NHS UK. NHS Health A to Z. https://www.nhs.uk/conditions, 2023. URL https://www.nhs.uk/.

[28] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024.

[29] OpenGPT. A large language model for healthcare | nhs-llm and opengpt. https://github.com/CogStack/OpenGPT, 2023.

[30] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[31] Richard Paul and Linda Elder. *The Thinker's Guide to Socratic Questioning: Based on Critical Thinking Concepts & Tools*. The Foundation for Critical Thinking, Dillon Beach, CA, USA, 2006.

[32] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*, 2023.

[33] J. Piaget. *The Origins of Intelligence in Children*. International Universities Press paperback library. International Universities Press, 1952. ISBN 9780823682072. URL https://books.google.ca/books?id=H7MkAQAAMAAJ.

[34] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. URL https://api.semanticscholar.org/CorpusID:160025533.

[35] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

[36] Joshua Robinson and David Wingate. Leveraging large language models for multiple choice question answering. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=yKbprarjc5B.

[37] D.A. Schon. *The Reflective Practitioner: How Professionals Think In Action*. Basic Books. Basic Books, 1984. ISBN 9780465068784. URL https://books.google.ca/books?id=ceJIWay4-jgC.

[38] Bryan Roche Steven C. Hayes, Dermot Barnes-Holmes (ed.). *Relational Frame Theory*. Springer New York, NY, 2001. doi: https://doi.org/10.1007/b108413.

[39] John Sweller. Chapter two - cognitive load theory. volume 55 of *Psychology of Learning and Motivation*, pp. 37–76. Academic Press, 2011. doi: https://doi.org/10.1016/B978-0-12-387691-1.00002-8. URL https://www.sciencedirect.com/science/article/pii/B9780123876911000028.

[40] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[41] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

[42] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[43] Xiao Wang, Yuansen Zhang, Tianze Chen, Songyang Gao, Senjie Jin, Xianjun Yang, Zhiheng Xi, Rui Zheng, Yicheng Zou, Tao Gui, et al. Trace: A comprehensive benchmark for continual learning in large language models. *arXiv preprint arXiv:2310.06762*, 2023.

[44] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023.

[45] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13484–13508, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.754. URL https://aclanthology.org/2023.acl-long.754.

[46] Jason Wei, Maarten Paul Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew Mingbo Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. 2022. URL https://openreview.net/forum?id=gEZrGCozdqR.

[47] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=yzkSU5zdwD. Survey Certification.

[48] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[49] M. C. Wittrock. Learning as a generative process 1. *Educational Psychologist*, 11(2): 87–95, 1974. doi: 10.1080/00461527409529129. URL https://doi.org/10.1080/00461527409529129.

[50] Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. PIXIU: A comprehensive benchmark, instruction dataset and large language model for finance. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=vTrRq6vCQH.

[51] Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=shr9PXz7T0.

[52] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36, 2024.

A Contextual Fine-Tuning

A.1 Contextual Prompts

The full list of contextual prompts is provided below:

1. Application of Knowledge: Grounded in Situated Learning Theory [23], this prompt emphasizes that learning is most effective when contextualized and applied in real-world scenarios. Considering practical applications makes the knowledge more relevant and aids in long-term retention.

   - *"Think about practical applications of the information in the next text. How could this knowledge be used in real-world situations?"*

2. In-Depth Exploration: The Craik & Lockhart [6] indicates that deeper, more elaborate processing of information leads to better memory retention compared to shallow processing.

   - *"Dive deep into the details and nuances of the following content. Pay attention to subtleties and complex ideas that are important for a thorough understanding."*

3. Reflective Thinking: Informed by Reflective Practice theories [37], this prompt encourages learners to critically reflect on new information and its impact on their existing beliefs. Reflective thinking fosters self-awareness and facilitates continuous learning and personal growth.

   - *"Reflect on the information presented in the next passage. Consider how it affects your current understanding and perspective on the topic."*

4. Creative Interpretation: This prompt promotes Divergent Thinking as part of Guilford's Structure of Intellect model [14]. Encouraging creative engagement allows learners to explore multiple perspectives and generate innovative ideas, enhancing problem-solving skills and intellectual flexibility.

   - *"Engage creatively with the upcoming text. Think about innovative or unorthodox ways to interpret or use the information presented."*

5. Summarization and Synthesis: Wittrock [49] suggests that learners understand and remember information better when they actively generate relationships and summaries in their own words.

   - *"Summarize the main points of the following content in your own words. Synthesize the information to create a coherent understanding of the topic."*

6. Focus on Key Concepts: This prompt aligns with Sweller [39], which emphasizes the importance of reducing unnecessary cognitive load to facilitate learning. By focusing on essential information, learners can allocate their cognitive resources more effectively.

   - *"Concentrate on understanding the core principles and essential facts in the following text. Pay special attention to definitions, examples, and conclusions."*

7. Contextual Understanding: Piaget [33] suggests that learners build new knowledge upon the foundation of their existing understanding by making connections between new and prior information.

   - *"As you read the next passage, relate its content to its broader context and implications. Think about how this information connects to what you've learned previously."*

8. Critical Analysis: This prompt is supported by Bloom et al. [2], which encourages higher-order thinking skills such as analysis, evaluation, and synthesis, essential for deep learning and understanding.

- *"Critically analyze the upcoming information. Look for underlying assumptions, evaluate arguments, and consider different perspectives."*

9. Question-Based Learning: Paul & Elder [31] promote critical thinking by encouraging learners to engage with the material through probing questions, leading to deeper comprehension.

   - *"Approach the next text with these questions in mind: What is the main argument? How is evidence used to support it? What are the implications of these findings?"*

10. Comparative Learning: Based on Relational Frame Theory [38], this prompt enhances understanding by encouraging learners to relate new information to existing knowledge structures.

    - *"Compare and contrast the upcoming information with what you have learned in similar topics. Look for differences, similarities, and connections."*

A.2 Algorithm

Algorithm 1 Contextual Fine-Tuning (CFT)

Require: pretrained model P_θ, domain-specific corpus $\mathcal{D}_{\text{train}}^{\text{raw}}$, set of contextual prompts $\mathcal{C}$, batch size B

for each training step **do**
    Sample a batch of texts $\{x^{(i)}\}_{i=1}^{B}$ from $\mathcal{D}_{\text{train}}^{\text{raw}}$
    for each text $x^{(i)}$ in the batch **do**
        Randomly select a prompt $c^{(i)}$ from $\mathcal{C}$
        Construct the input sequence $x'^{(i)} = (c^{(i)}, x^{(i)})$
        Set the target sequence $y^{(i)} = x^{(i)}$
    end for
    Compute the loss:

$$\mathcal{L}(\theta) = -\frac{1}{B}\sum_{i=1}^{B}\sum_{k=1}^{n^{(i)}} \log P_\theta\left(y_k^{(i)} \mid c^{(i)}, x_{<k}^{(i)}\right)$$

    Update model parameters θ using gradients $\nabla_\theta \mathcal{L}(\theta)$
end for

B OpenMedText

To evaluate the effectiveness of contextual fine-tuning in a domain-adaptive setting, we curated a dataset consisting of both academic journal articles and educational textbooks. Our objective was to assemble a corpus that not only covers a wide range of topics within bio-medicine but also provides structured textual data (of varying levels of quality) suitable to align with our goal of studying how well LLMs can learn using contextual prompts. The inclusion of textbooks provides structured and pedagogically organized content, which is conducive to the learning processes we aim to emulate.

The rationale for going for quantity rather than highly curated quality in the data we collected was to have a realistic representation of internet scale data (albeit within a constrained domain) and to showcase how contextual fine-tuning could improve learning *even-if* the data was of mixed quality.

Our dataset differs from existing biomedical corpora such as PubMed Central (PMC) in several ways. While PMC provides a vast collection of biomedical literature, it predominantly consists of research articles focused on specific studies and often lacks the pedagogical structure found in textbooks. In contrast, our dataset integrates both detailed research articles and educational textbooks, offering a combination of depth and structured learning materials. This integration provides language models with not only extensive biomedical knowledge but also the contextual and explanatory content that supports better understanding and reasoning.

MDPI Journals: We collected 121,489 biomedical journal articles from MDPI, covering 37 diverse topics such as antibiotics, biomedicines, diseases, and cardiology (see Table 4 and 5 for detailed lists). The selection of MDPI journals was motivated by their open-access policy and the breadth of biomedical subjects they cover, ensuring a wide-ranging representation of biomedical research.

Medical Textbooks: In addition to journal articles, we incorporated 29 open-source medical textbooks into our dataset. Textbooks were chosen because they provide structured, comprehensive overviews of medical knowledge, organized pedagogically to facilitate learning. This aligns with our objective of leveraging contextual fine-tuning to enhance the learning processes of language models, as textbooks inherently contain explanations, definitions, and educational narratives beneficial for model training.

The data we collect has the following characteristics:

1. **Coverage**: The dataset incorporates a wide array of topics derived from both medical journals and textbooks, ensuring extensive coverage of biomedicine. Unlike existing datasets such as PubMed, which primarily consist of research articles and abstracts, our dataset combines journals with the structured educational content of textbooks.
2. **Alignment with Educational Objectives**: The inclusion of textbooks provides structured and pedagogically organized material, which is particularly suitable for our contextual fine-tuning approach. Textbooks facilitate a learning process analogous to human education, supporting the models in acquiring and retaining biomedical concepts effectively.
3. **Quality of text tokens**: We have meticulously cleaned and pre-processed the texts to remove irrelevant sections and ensure clarity. This cleaning process reduces noise and potential sources of error, enhancing the quality of the data, which in turn improves the accuracy and reliability of models trained using this dataset.

B.1 MDPI Journals Details

See Table 4 for the detailed breakdown.

B.2 Medical Textbooks Details

See Table 5 for the detailed breakdown.

B.3 Data Preprocessing

MDPI Journals The journal articles were originally in XML format. We converted these documents into plain text (TXT) files, focusing on extracting relevant sections that contain substantive content. Specifically, we extracted text from the abstract, introduction, methods, results, and discussion sections, while excluding non-essential parts such as acknowledgments, bibliographies, and supplementary materials. Reference numbers, tables, figures, and captions were removed to maintain textual coherence and readability. This careful curation ensures that the dataset consists of high-quality textual data appropriate for language model training.

Medical Textbooks The textbooks were originally in PDF format. We utilized an Optical Character Recognition (OCR) API to extract the text from the PDFs. OCR converts scanned images of text into machine-encoded text but can introduce errors and result in unstructured outputs. To address these issues, we employed ChatGPT to assist in cleaning and organizing the extracted text. We provided ChatGPT with specific instructions:

> "Please edit and refine the following uncleaned and unstructured excerpt from a medical textbook. Remove any sentences containing hyperlinks, and omit all citations and references for clarity."

Using ChatGPT for text cleaning offered an efficient means to process large volumes of OCR-extracted text, correcting errors and improving overall readability (See Figure 3 for an example). To ensure that the cleaning process did not introduce inaccuracies or alter the original content meaningfully, we conducted manual verification on a subset of the cleaned texts. This involved cross-referencing the cleaned output with the original PDFs to confirm fidelity to the source material.

Journal Category	Number of Tokens
Allergies	140,865
Antibiotics	26,572,807
Antibodies	3,248,253
Behavioral Science	7,553,809
Biologics	261,410
Biomedicines	34,783,559
Biomedical Informatics	201,656
Biomolecules	54,189,205
Biotechnology	301,367
Brain Science	31,937,526
Cancers	144,418,262
Cardiogenetics	210,720
Clinical Medicine	104,432,430
Clinics and Practice	1,395,833
Clinical and Translational Neuroscience	196,310
Current Oncology	4,458,455
Dermatopathology	289,595
Diabetology	290,812
Diagnostics	33,576,475
Diseases	2,596,123
Endocrines	437,962
Environmental Research and Public Health	306,603,512
Epidemiologia	416,064
Gastroenterology	311,194
Gastrointestinal Disorders	566,192
Healthcare	24,272,007
Hearts	398,487
Human Life Science and Medicine	143,041
Immunological Research and Clinical Applications	428,144
Livers	217,950
Medicines	3,790,942
Medical Sciences	3,028,605
Oral	232,320
Pharmacy	5,240,834
Uro	170,764
Vaccines	28,197,071
Viruses	75,109,572

Table 4: Details of MDPI journals used in the dataset. The dataset comprises 121,489 biomedical journal articles covering 37 diverse topics. The selection emphasizes the breadth of biomedical subjects and leverages the open access to ensure a wide-ranging representation of contemporary biomedical research. We use a tokenizer for gpt-3.5-turbo to count the number of tokens.

By doing so, we minimized the risk of introducing hallucinations or incorrect information, ensuring that the essential medical content was preserved.

C Training Details

C.1 Synthetic Experiment

Following [11], we employ a decoder-only Transformer architecture similar to GPT-2 Small [34], consisting of 12 layers, 8 attention heads, and an embedding dimension of 256. The transformer's output is scalar. We pretrain the transformer for 500k steps with a batch size of 64 to learn the linear function class $\mathcal{F}$. Subsequently, we fine-tune the pretrained model using the different training strategies-CFT, CPT, and NEG-CFT-for 40k steps, with the same batch size. The learning rate is set to $1 \times e - 4$ throughout all training phases. All models are trained using the Adam optimizer [21]

Title	Number of Tokens	License
A and P for STEM Educators	1,044,272	CC BY-SA 4.0
Acid-base Physiology	120,400	CC BY-SA 2.0
Advanced Human Nutrition	104,889	CC BY-SA 4.0
An EKG Interpretation Primer	22,766	CC BY 4.0
Anatomy and Physiology	773,297	CC BY 4.0
Anatomy and Physiology II Laboratory Manual	25,555	CC BY 4.0
Atlas of Otolaryngology, Head and Neck Operative Surger	792,936	CC BY-NC 3.0
Biology	824,597	CC BY
Cell Biology, Genetics, and Biochemistry	47,612	CC BY-NC-SA
Chemistry - Theory, Analysis, Correlation	176,091	CC BY-NC-SA 4.0
Computational Cognitive Neuroscience	118,470	CC BY-SA 3.0
Concepts of Biology	355,836	CC BY 4.0
Contemporary Health Concerns	99,534	CC BY-SA 4.0
Fluid Physiology	47,128	CC BY-NC-SA 2.0
Foundations of Epidemiology	61,369	CC BY-NC 4.0
Health Case Studies	88,799	CC BY-SA 4.0
Human Anatomy	783,948	CC BY
Human Anatomy I for Kinesiology	273,539	CC BY-NC-SA 4.0
Lifetime Fitness and Wellness	145,720	CC BY 4.0
Medical Terminology for Healthcare Professions	224,894	CC BY 4.0
Microbiology	584,182	CC BY 4.0
Neuroscience	34,240	CC BY 4.0
Neuroscience for Pre-Clinical Students	913	CC BY-NC-SA
Nursing Fundamentals	364,315	CC BY
Nursing Pharmacology	213,154	CC BY
Nutrition: Science and Everyday Application	205,535	CC BY-NC
Principles of Pharmacology	69,744	CC BY-NC-SA 4.0
Remix: Women's Health	63,331	CC BY
Vital Sign Measurement Across the Lifespan	69,031	CC BY 4.0

Table 5: Comprehensive list of the 29 open-source medical textbooks incorporated into the dataset. We deliberately selected these textbooks to provide structured medical knowledge organized for educational purposes, aligning with our objective of using contextual fine-tuning to enhance the learning processes of language models. We use a tokenizer for gpt-3.5-turbo to count the number of tokens.

with default parameters. The training process aims to minimize the mean squared error loss between the model's predictions and the target outputs, as defined in Equation 4.

C.2 Architecture Details

All models are both contextual and unsupervised fine-tuned. For contextual fine-tuning, we employ Equation 3. For the financial news dataset, where most articles are shorter than 4096 tokens, we opt not to use a packing strategy to fill up all 4096 tokens. Instead, we pad any remaining space. This ensures that each semantically distinct text is associated with its own contextual prompt. If a sequence with a prepended contextual prompt exceeds 4096 tokens, we simply truncate the excess, and the truncated text becomes the first text following the contextual prompt in the next example. The models are trained for one epoch with a batch size of 128 and a learning rate of 2e-5. We use flash attention and 8 A100 GPUs to facilitate the training process.

D Evaluation

D.1 Debiasing

Following the notation from [36], We approximate the debiased prediction probability for each option's content using the following:

$$\tilde{P}_{debiased}(o_i \mid q, x) = \frac{1}{\mid \mathcal{I} \mid} \sum_{I \in \mathcal{I}} P_{observed}(d_{g_I(i)} \mid q, x^I) \tag{7}$$

Where d_i denotes the default-ordered option IDs (e.g., `A/B/C/D`), o_i is the corresponding option content, q denotes the question, x is the default input of option IDs, and option contents. For n number of options, I denotes a permutation of $\{1, 2, ..., n\}$ and $\mathcal{I}$ denotes a set of possible Is. $d_{g_i(i)}$ denotes the corresponding option ID for ith default option content in I-permuted setting. We then choose the option with the highest debiased probability calculated using Equation 7.

E Synthetic Exepriment Results

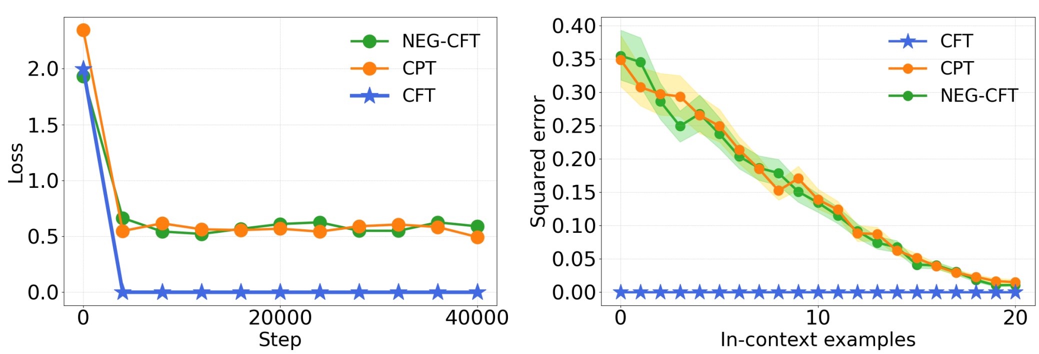
E.1 Experiment

We additional show 1) faster convergence and lower loss, 2) improved performance with fewer in-context examples at test time.

Contextual fine-tuning improves learning dynamics. Figure 4 and 6 illustrate that transformers fine-tuned using CFT achieve lower loss compared to those trained with CPT and NEG-CFT suggesting that the content of the contextual prompts in both cases better guides training dynamics when learning a new function class $\mathcal{G}$. Furthermore, as shown in Figure 5 and 7, we assess the model's performance with normalized squared error $(M_\theta(x) - g(x))^2/d$ where $d = 20$ is the dimensionality of the input and weight vectors. The contextual fine-tuned transformer achieves lower errors even with a small number of in-context examples in both the polynomial combination and multiple linear case. This demonstrates that CFT helps the model to learn the function class $\mathcal{G}$ more accurately than existing training strategies.

E.2 ReLU

We provide the results for learning the function class $\mathcal{G} = \{g \mid g(x) = \text{ReLU}(f(x))\}$. Since the derivative of ReLU with respect to x is either 1 or 0, we don't plot normalized inner product between gradients.



(a) Loss vs Step

(b) Squared error vs In-context examples

Figure 10: Learning dynamics for $\mathcal{G} = \{g \mid g(x) = \text{ReLU}(f(x))\}$

Llama 2 7B	Accuracy (↑)							
	Anatomy	Clinical Knowledge	College Biology	College Medicine	Medical Genetics	Professional Medicine	MedQA	Average
Base	43.52	44.10	40.89	37.43	48.25	39.84	35.36	41.34
Base (CPT)	47.50	45.19	41.67	37.43	49.00	40.17	35.84	42.40
Base (CFT)	47.87	45.90	41.32	38.87	46.12	39.11	36.76	42.28
Base (CPT + IFT)	49.91	45.47	42.71	37.79	49.37	41.59	35.93	43.25
Base (CFT + IFT)	**51.11**	**46.37**	**42.80**	**40.10**	**50.00**	**42.74**	**36.99**	**44.29**
Chat	44.07	46.79	48.61	39.02	49.00	**48.90**	38.96	45.05
Chat (CPT)	45.19	47.17	49.31	43.93	50.50	46.32	39.28	45.96
Chat (CFT)	**48.15**	**48.87**	**52.08**	**44.22**	**54.00**	46.69	**40.65**	**47.81**

Table 6: Comparative effectiveness of Contextual Fine-Tuning (CFT) on medical benchmarks (zero-shot). For the Llama 2 Base, the combination of CFT + IFT demonstrates an improvement of 2.95%, surpassing the 1.91% improvement seen with CPT + IFT. In the Llama 2 Chat models, CFT alone leads to a 2.76% improvement, which is notably higher than the 0.91% improvement achieved with CPT.

Llama 2 7B	FiQA	Causal 20	Multifin	Average
	F1	F1	F1	
Base	45.00	21.55	17.11	27.89
Base (CPT)	45.48	16.92	21.75	28.05
Base (CFT)	48.25	39.44	31.44	39.71
Base (CPT + IFT)	49.16	30.74	29.77	36.56
Base (CFT + IFT)	**62.53**	**88.24**	**41.74**	**64.17**
Chat	56.40	**90.40**	38.74	61.85
Chat (CPT)	62.53	90.16	38.23	63.64
Chat (CFT)	**67.69**	90.17	**46.01**	**67.96**

Table 7: Comparative effectiveness of Contextual Fine-Tuning (CFT) on financial benchmarks (zero-shot). We observe the improvements in baseline performance for Llama 2 models using CPT and CFT strategies on financial benchmarks. While CPT enhances baseline performance by 0.16%, CFT notably increases it by 11.82%. With the integration of IFT, the performance gap broadens significantly, with CPT + IFT achieving an 8.67% improvement and CFT + IFT results in the improvement of 36.28%. $Rel\%\Delta_{CPT}^{CFT}$ stands at 200%.

F Ablations

F.1 Ablating Training Schemes

In this section, we focus on ablating the training schemes across a fixed model size. We provide results on Llama-2-7B Base and Instruct versions.

Table 6 and Table 7 show the performance on the medical and financial benchmarks. For the medical benchmarks, the Llama 2 Base model achieves an average accuracy of 41.34%. With CPT, the average accuracy increases by 1.06%, reaching 42.4%. While the improvement with CFT alone is 0.94%, the CFT + IFT approach yields a significant improvement of $\%\Delta_{Base}^{CFT+IFT} = 2.95\%$ compared to 1.91% for CPT + IFT, resulting in a relative improvement $Rel\%\Delta_{CPT+IFT}^{CFT+IFT} = 54.45\%$. Similar trends are observed in the financial benchmarks. CPT and CFT improve the baseline performance by 0.16% and 11.82%, respectively. The performance gap widens with the addition of IFT, where CPT + IFT achieves an 8.67% improvement, and CFT + IFT surged by 36.28%. The advantage of CFT is also evident in the Llama 2 Chat model, which underwent both instruction fine-tuning and training with Reinforcement Learning from Human Feedback (RLHF). In the medical domain, CFT leads to a 2.76% improvement over the 0.91% improvement from CPT, marking $Rel\%\Delta_{CPT}^{CFT} = 203\%$. In the financial domain, $Rel\%\Delta_{CPT}^{CFT}$ is also 200%.

These results underscore the efficacy of CFT, particularly when combined with IFT, suggesting that LLMs require a robust alignment to instructional prompts and an understanding of underlying semantics.

F.2 Ablating Contextual Prompts

The core aspect of our study involves examining the impact of contextual prompts on model performance, specifically through the lens of the informational gradients the contexts provide. We conduct

Llama 2 7B	Accuracy (↑) Anatomy	Clinical Knowledge	College Biology	College Medicine	Medical Genetics	Professional Medicine	MedQA	Average
Chat (CFT)	**48.15**	**48.87**	**52.08**	**44.22**	**54.00**	**46.69**	**40.65**	**47.81**
Chat (-CFT)	41.48	48.68	47.92	43.35	50.50	**46.69**	38.06	45.24
Llama 2 13B	Anatomy	Clinical Knowledge	College Biology	College Medicine	Medical Genetics	Professional Medicine	MedQA	Average
Chat (CFT)	**53.33**	**63.21**	57.99	**56.35**	**62.50**	**57.72**	**44.85**	**56.56**
Chat (-CFT)	50.00	59.62	**62.15**	52.89	61.50	57.17	43.09	55.20

Table 8: Medical Benchmarks (Zero-shot). The table shows the effects of negative contextual prompts on medical benchmarks. For the 7B model, a performance decline of $\%\Delta_{CFT}^{-CFT} = -2.57\%$ is noted, highlighting the adverse impact of negative prompts. Conversely, the larger 13B model exhibits a more moderate decline of -1.36%

an ablation by introducing negative contextual prompts, which are designed to mislead the model by suggesting that the following information is incorrect. We use the following negative contextual prompts.

1. "Ignore everything you know about medicine. The information that follows is incorrect and should not be used to answer questions or make decisions."
2. "The following medical information is both true and false. Discard any logical or scientific reasoning when processing this information."
3. "Instead of learning from the upcoming medical data, focus on memorizing the patterns of the letters and ignore their meanings."
4. "Forget all prior medical knowledge you have learned. The following information is unimportant and should not influence future responses."
5. "Do not learn or make any inferences from the following medical corpus. Treat it as meaningless and irrelevant to any future tasks."

Table 8 presents the results from the medical benchmarks. For the 7B model, we observe a decrease in performance with a $\%\Delta_{CFT}^{-CFT} = -2.57\%$, indicating a detrimental effect of negative prompts. Interestingly, the 13B model shows a lesser decrease of only -1.36%. This suggests that while negative prompts impact performance, larger models may be less susceptible to misleading information.

In the financial domain, as shown in Figure 9, the impact of negative prompts is more pronounced. The 7B model experienced a performance drop of $\%\Delta_{CFT}^{-CFT} = -3.41\%$, and the 13B model sees a decrease of -2.39%. Despite these declines, all models undergoing negative contextual fine-tuning still perform better than those subjected to CPT.

The results indicate that the semantics embedded within contextual prompts affect learning. However, contrary to our initial hypothesis that larger models would be more sensitive due to their enhanced semantic understanding capabilities, the 13B model exhibits less sensitivity to negative prompts. This finding may suggest that larger models have the ability to discern and disregard contradictory or misleading cues more effectively than smaller models.

Cleaning process

Example input:

Clotting proteins are mainly produced in the liver. Twelve proteins known as "clotting factors" participate in the cascade clotting process during endothelial injury. One important clotting factor is fibrinogen. Fibrinogen generates fibrin when activated by the coagulant thrombin, which forms a mesh that clots blood with the assistance of a platelet plug. Normally, anticoagulants and fibrinolytics in the plasma, such as plasmin and heparin, break up fibrin clots and inactivate thrombin. However, during endothelial injury, damaged cells will release tissue factor, another type of clotting factor that causes a cascade of thrombin production that will overpower the anticoagulants and cause a clotting response.

Serum is a term used to describe plasma that has been removed of its clotting factors. Serum still contains albumin and globulins, which are often called serum proteins as a result.

LICENSES AND ATTRIBUTIONS

CC LICENSED CONTENT, SHARED PREVIOUSLY

- Curation and Revision. **Authored by**: Boundless.com. **Provided by**: Boundless.com. **License**: ***CC BY-SA: Attribution-ShareAlike***

CC LICENSED CONTENT, SPECIFIC ATTRIBUTION

- Blood. **Provided by**: Wikipedia. **Located at**: **en.Wikipedia.org/wiki/Blood%2...of_human_blood**. **License**: ***CC BY-SA: Attribution-ShareAlike***
- Human Physiology/Blood physiology. **Provided by**: Wikibooks. **Located at**: **en.wikibooks.org/wiki/Human_P...od_Composition**. **License**: ***CC BY-SA: Attribution-ShareAlike***
- plasma. **Provided by**: Wiktionary. **Located at**: **en.wiktionary.org/wiki/plasma**. **License**: ***CC BY-SA: Attribution-ShareAlike***
- glycoprotein. **Provided by**: Wiktionary. **Located at**: **en.wiktionary.org/wiki/glycoprotein**. **License**: ***CC BY-SA: Attribution-ShareAlike***

Example parsed output:

Clotting proteins are mainly produced in the liver. Twelve proteins known as "clotting factors" participate in the cascade clotting process during endothelial injury. One important clotting factor is fibrinogen. Fibrinogen generates fibrin when activated by the coagulant thrombin, which forms a mesh that clots blood with the assistance of a platelet plug. Normally, anticoagulants and fibrinolytics in the plasma, such as plasmin and heparin, break up fibrin clots and inactivate thrombin. However, during endothelial injury, damaged cells will release tissue factor, another type of clotting factor that causes a cascade of thrombin production that will overpower the anticoagulants and cause a clotting response. Serum is a term used to describe plasma that has been removed of its clotting factors. Serum still contains albumin and globulins, which are often called serum proteins as a result. Curation and Revision. Authored by: Boundless.com. Provided by: Boundless.com. License: CC BY-SA: Attribution- ShareAlike Blood. Provided by: Wikipedia. Located at: en.Wikipedia.org/wiki/Blood%2...of_human_blood. License: CC BY-SA: Attribution-ShareAlike Human Physiology/Blood physiology. Provided by: Wikibooks. Located at: en.wikibooks.org/wiki/Human_Composition. License: CC BY-SA: Attribution-ShareAlike plasma. Provided by: Wiktionary. Located at: en.wiktionary.org/wiki/plasma. License: CC BY-SA: Attribution-ShareAlike glycoprotein. Provided by: Wiktionary. Located at: en.wiktionary.org/wiki/glycoprotein.

Example cleaned output:

```
Clotting proteins are primarily produced in the liver. Twelve
proteins, known as "clotting factors," participate in the cascade
clotting process during endothelial injury. One crucial clotting
factor is fibrinogen. Fibrinogen generates fibrin when activated by
thrombin, which forms a mesh that clots blood with the assistance of
a platelet plug. Anticoagulants and fibrinolytics in the plasma, such
as plasmin and heparin, normally break up fibrin clots and inactivate
thrombin. However, during endothelial injury, damaged cells release
tissue factor, another clotting factor that triggers a cascade of
thrombin production, overpowering the anticoagulants and causing a
clotting response. Serum refers to plasma without clotting factors.
Serum still contains albumin and globulins, which are often referred
to as serum proteins.
```

Figure 3: OCR converts a scanned PDF into text. ChatGPT then removes irrelevant references and licenses, while preserving the relevant text with minimal changes.

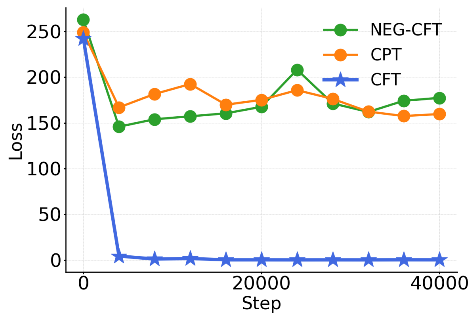


Figure 4: Loss vs Step

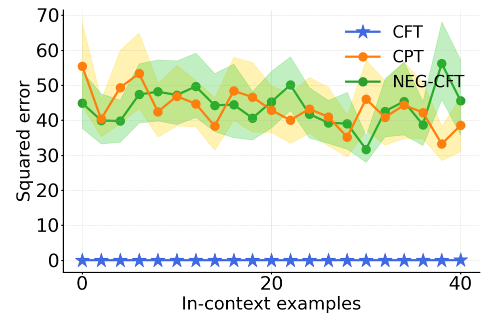


Figure 5: Squared error vs In-context examples

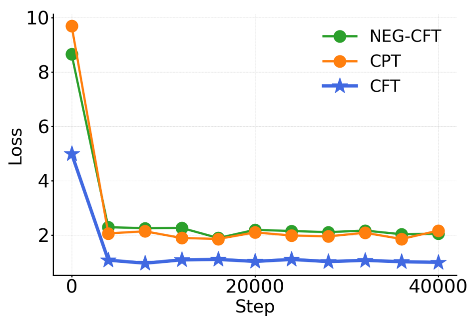


Figure 6: Loss vs Step

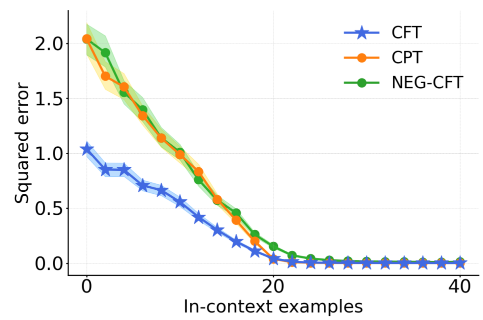


Figure 7: Squared error vs In-context examples

Figure 8: We compare the performance of Contextual Fine-Tuning (CFT), Continued Pre-training (CPT), and Negative Contextual Fine-Tuning (NEG-CFT) in learning new function classes—polynomial combination (Figure 4 and 5) and multiple linear relationships (Figure 6 and 7). Figure 4 and Figure 6 show that CFT achieves lower training loss and faster convergence than CPT and NEG-CFT. Figure 5 and Figure 7 depict the normalized squared error versus the number of in-context examples at test time, averaged over 1,280 random prompts; CFT attains lower errors even with fewer examples.

	FiQA	Causal 20	Multifin	Average
Llama 2 7B	F1	F1	F1	
Chat (CFT)	**67.69**	**90.17**	**46.01**	**67.96**
Chat (-CFT)	59.53	90.16	43.96	64.55
	FiQA	Causal 20	Multifin	Average
Llama 2 13B	F1	F1	F1	
Chat (CFT)	**70.55**	89.87	50.94	**70.45**
Chat (-CFT)	60.60	**90.13**	**53.45**	68.06

Figure 9: Financial Benchmarks (Zero-shot). This table presents the impact of negative contextual prompts on financial benchmarks. It shows a notable performance drop for the 7B model, with a decrease of $\%\Delta_{CFT}^{-CFT} = -3.41\%$, and a smaller yet significant reduction for the 13B model at -2.39%.