A Closer Look at the Limitations of Instruction Tuning

Sreyan Ghosh * [1 2] **Chandra Kiran Reddy Evuru** * [1] **Sonal Kumar** * [1] **Ramaneswaran S** [3] **Deepali Aneja** [2]
Zeyu Jin [2] **Ramani Duraiswami** [1] **Dinesh Manocha** [1]

Abstract

Instruction Tuning (IT), the process of training large language models (LLMs) using instruction-response pairs, has emerged as the predominant method for transforming base pre-trained LLMs into open-domain conversational agents. While IT has achieved notable success and widespread adoption, its limitations and shortcomings remain underexplored. In this paper, through rigorous experiments and an in-depth analysis of the changes LLMs undergo through IT, we reveal various limitations of IT. In particular, we show that (1) IT fails to enhance knowledge or skills in LLMs. LoRA fine-tuning is limited to learning response initiation and style tokens, and full-parameter fine-tuning leads to knowledge degradation. (2) Copying response patterns from IT datasets derived from knowledgeable sources leads to a decline in response quality. (3) Full-parameter fine-tuning increases hallucination by inaccurately borrowing tokens from conceptually similar instances in the IT dataset for generating responses. (4) Popular methods to improve IT do not lead to performance improvements over a simple LoRA fine-tuned model. Our findings reveal that responses generated solely from pre-trained knowledge consistently outperform responses by models that learn any form of new knowledge from IT on open-source datasets. We hope the insights and challenges revealed in this paper inspire future work in related directions.

1. Introduction

Large Language Models (LLMs) pre-trained at an incredible scale with the next token prediction objective implicitly compress world knowledge in their parameters (Zhao et al., 2023). These models learn general-purpose representations, which can then be *aligned* with the desired response characteristics (Zhang et al., 2023a). In the recent past, various methods for aligning LLMs have been proposed, out of which *instruction tuning* (IT) (Wei et al., 2022) and *reinforcement learning from human feedback* (RLHF) (Bai et al., 2022) have gained the most popularity. IT, the process of fine-tuning an LLM with instruction-response pairs, enables it to follow or complete tasks instructed by humans. On the other hand, RLHF continually tunes an IT-ed LLM to further align with human preference. While RLHF is prohibitively expensive due to the requirement of large amounts of human-preference data (Bai et al., 2022), IT with standard supervised loss has proved to be a more prevalent technique for alignment (Zhou et al., 2023). IT-based alignment has led to significant improvements in LLMs, unlocking impressive capabilities (Bubeck et al., 2023), suggesting that fine-tuning is key to building and improving LLM-based conversational agents. For the remainder of the paper, we interchangeably refer to an IT-ed model as a *fine-tuned* model and the process of IT interchangeably as *fine-tuning*.

Early work on IT focused on fine-tuning and evaluating LLMs with popular natural language processing (NLP) tasks where the dataset instances are phrased as natural language instructions. However, solely gauging the impact of IT on an LLM using conventional NLP tasks and metrics falls short in comprehensively assessing the array of its abilities, such as reasoning and knowledge sharing, usually demanded by the diverse range of tasks encountered by open-domain conversational agents (Wang et al., 2023). Since the colossal success of ChatGPT (OpenAI, 2023), a more recent line of work aims at evaluating and improving IT with open-domain instruction following data (Zheng et al., 2023). However, though IT has been shown to achieve remarkable improvement in performance and generalization to unseen NLP tasks, its limitations and shortcomings have rarely been explored. We attribute this to three main reasons: (1) Lack of comprehensive evaluation metrics for evaluating open-domain instruction following abilities. (2) Lack of clear understanding of the exact transformation a base pre-trained LLM goes through with IT, and (3) The rapid rise of powerful semi-open-source chat models (fine-tuned LLMs that don't reveal their IT data) encourages the use of these

---

*Equal Technical Contribution. [1]University of Maryland, College Park, USA [2]Adobe, USA [3]NVIDIA, India. Correspondence to: Sreyan Ghosh <sreyang@umd.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

models merely as tools, leading to an under-exploration of several critical elements in their development.

Main Contributions. In this paper, we investigate and reveal several limitations of IT. To achieve this, we closely study the transformation a base pre-trained model goes through after IT by experimenting with various open-source IT datasets, LLMs, and training paradigms. Additionally, for evaluation, we conduct a combination of expert human evaluation, GPT-4-based multi-aspect evaluation, and token distribution analysis (Lin et al., 2023). Our study explicitly focuses on evaluating the effectiveness of IT in developing open-domain conversational agents (also commonly known as *chat models*) and is limited to single-turn interactions. Our extensive results reveal the following:

1. **IT is not a knowledge [1] enhancer.** Similar to concurrent work (Gudibande et al., 2023), we first find that IT does not act as a knowledge enhancer at its current open-source scale. To dig deeper, we compare the token distributions (explained further in Section 3) between base LLMs and their IT-ed versions and find that LoRA (Hu et al., 2021), which only teaches response initiation and extracts most of the response from the pre-trained knowledge itself, leads to the most factually correct responses. On the other hand, full-parameter fine-tuning leads to knowledge degradation and reduces overall response quality.
2. **Pattern-copying often hurts performance.** We first show that models IT-ed using LoRA and full-parameter fine-tuning learn pattern-copying very differently. While the former learns just stylistic tokens, the latter leads the model to adapt more deeply to the specifics of the new training data. Next, we show that while pattern-copying sometimes has some advantages, like detailed and comprehensive answering, most of the time, it hurts the factual correctness of the response. Finally, we propose a simple solution to overcome this.
3. **Full fine-tuning leads to knowledge degradation by increasing chances of hallucination. These hallucinations are tokens borrowed from the IT dataset.** We show that when a model hallucinates or outputs incorrect tokens in its responses, it is highly probable that these tokens are borrowed from instances with similar concepts in the IT dataset itself. We further study this from the lens of causal analysis. This effect is more prevalent in models trained using full-parameter fine-tuning than in LoRA fine-tuning.
4. **Various methods to improve IT, proposed in literature, do not improve model performance.** We compare several methods like NEFTune (Jain et al., 2023) and dataset filtering (Chen et al., 2023) on common grounds and show that, while these methods improve over a full-finetuned model, a LoRA fine-tuned model outperforms all of them. Consequently, these methods do not contribute to knowledge advancement, and models leveraging pre-trained knowledge remain superior.

2. Experimental Setting

LLMs. For the scope of our analysis, we experiment with 5 different types of LLMs, namely LLaMa-2 $_{\mathbf{7B}}$ (Touvron et al., 2023), LLaMa-2 $_{\mathbf{13B}}$, LLaMa-2 $_{\mathbf{70B}}$, Mistral-v0.1 $_{\mathbf{7B}}$ (Jiang et al., 2023) and Phi-1.5 $_{\mathbf{1.3B}}$ (Li et al., 2023b). We use only the base pre-trained versions of the models (and not the chat variants) and fine-tune them with IT ourselves. We employ LLaMa-2 $_{\mathbf{70B}}$ only in a fraction of experiments owing to compute constraints.

Fine-tuning Datasets. For fine-tuning with IT, we experiment with various synthetic and human-written IT datasets. For synthetic, we use Alpaca $_{\mathbf{52k}}$ with open-domain instruction-response pairs, constructed by prompting ChatGPT with an initial seed dataset with few samples (Taori et al., 2023) and MedInstruct $_{\mathbf{52k}}$ from the medical domain constructed in similar fashion (Zhang et al., 2023b). For human-written , we use LIMA $_{\mathbf{1K}}$ (Zhou et al., 2023) and databricks-dolly $_{\mathbf{15k}}$ (Conover et al., 2023). Finally, we also use Tulu-V2-Mix $_{\mathbf{326k}}$ (Ivison et al., 2023), which is an amalgamation of various open-source datasets.

Evaluation Datasets. Our experiments are confined to evaluating LLMs on instruction following capabilities with open-ended and free-form generation as LLMs have been shown to be not robust to MCQ (Zheng et al., 2024a). For evaluation, we primarily employ just-eval-instruct $_{\mathbf{1k}}$ (Lin et al., 2023), which is an amalgamation of various open-source IT evaluation sets and is tagged with various task types and topics (statistics in Appendix C). This choice is motivated by its diverse and succinct nature, which facilitates our efforts in conducting thorough, expert human evaluations. We employ only the first 800 instances and remove the last 200, as safety alignment is beyond the scope of this paper. For models fine-tuned on MedInstruct $_{\mathbf{52k}}$, we evaluate the model on MedInstruct-test $_{\mathbf{216}}$ unless stated otherwise. We do not evaluate the Open LLM Leaderboard (Beeching et al., 2023) as it does not fit into our criteria.

Fine-tuning paradigms. For fine-tuning with IT, we employ either LoRA fine-tuning (LFT) or (standard) full-parameter fine-tuning (SFT). LFT works by approximating the model's weight matrices with low-rank matrices. This reduces the number of parameters that need to be fine-tuned, which makes the fine-tuning process faster and more efficient. On the other hand, SFT, similar to generic ERM, works by adjusting all or most of the model's weights.

---

[1] We use the term "knowledge" to encompass both the factual information and task execution skills possessed by a model.

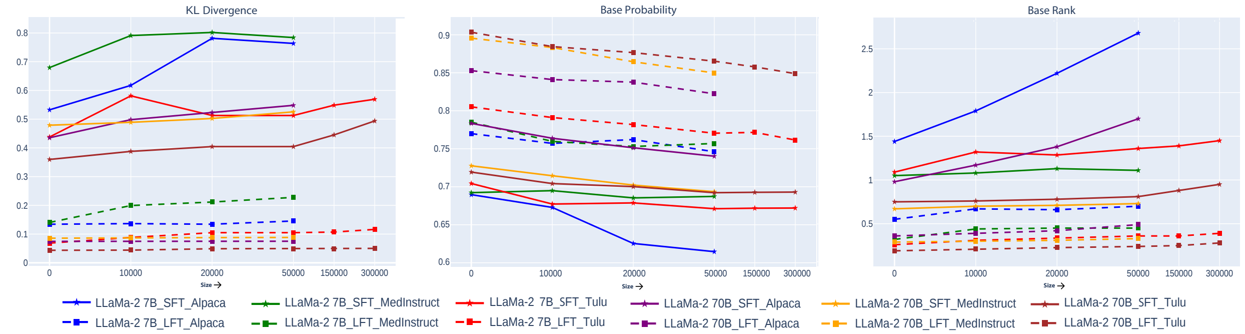


Figure 1. **Token distribution shift after IT.** We compare token distributions between base pre-trained models and their IT-ed versions using 3 metrics defined in Section 3. We show that (1) Overall, LFT experiences low token distribution shifts, indicating high alignment with pre-trained knowledge. (2) Shifts in SFT are much higher than in LFT. (3) LFT is unaffected by the scale of the IT dataset.

Evaluation. For evaluating our fine-tuned models, we perform a mix of expert human evaluation and automatic evaluation using GPT-4 Turbo (gpt-4-1106-preview). Recent research indicates that using ChatGPT and GPT-4 to score and assess outputs from LLMs aligns well with human evaluations, offering the added benefit of cost reduction (Liu et al., 2023; Li et al., 2023a; Chan et al., 2023; Xu et al., 2023b; Zhou et al., 2023). We borrow the explainable and multi-aspect evaluation framework from Lin et al. (2023), which prompts GPT-4 to assign a score between 1 and 5 to an LLM's response to an instruction on five aspects: helpfulness, clarity, factuality, depth, and engagement. It also provides an explanation for each score. The prompt can be found in Appendix A.

Naming Convention. We primarily follow the naming convention of "model_dataset_training-paradigm". For e.g., a LLaMa-2 $_{\mathbf{7B}}$ trained on Alpaca $_{\mathbf{52k}}$ with SFT would be named as LLaMa-2 $_{\mathbf{7B}}$_SFT_Alpaca $_{\mathbf{52k}}$.

Training and Evaluation hyper-parameters. All models are trained in a distributed manner for 3 epochs, with a learning rate of 5e-5 and an effective batch size of 32 (Taori et al., 2023). For LFT, we use a standard rank of 8 (Hu et al., 2021) as we did not find a substantial change in performance by decreasing (2,4) or increasing it (16,32). Zhang et al. (2024) also shows that scaling rank is ineffective for LFT. For generation, we employ greedy decoding (i.e., zero temperature) in all experiments.

Note: The main paper only presents results on the LLaMa-2 family, and results on fine-tuning additional LLMs and already fine-tuned LLMs available open-source, are provided in Section D.3. All our findings hold on other LLMs, too. Additionally, one should not confuse IT (and the findings of our paper) with general fine-tuning. While IT is aimed towards *aligning* a model towards specific response characteristics, LLMs may be fine-tuned for improving its various capabilities or knowledge. However, this may require fine-tuning datasets with characteristics different from IT datasets employed in the paper.

3. IT is (currently) Not a Knowledge Enhancer

Overview. This section investigates whether IT, at its current open-source scale, can function as a knowledge enhancer. Initially, we present the distinct natures of transformation that a base pre-trained LLM undergoes when subjected to LFT and SFT-based fine-tuning and show that while responses generated after LFT are closely aligned to pre-trained knowledge, they deviate significantly for SFT, indicating new knowledge acquisition. Subsequently, we show that this new knowledge often leads to a degradation in response quality, and relying predominantly on pre-trained knowledge often yields more factual and useful responses.

Finding 1. **LFT Responses align closely with the original pre-trained knowledge. SFT does not.** To study how fine-tuned models differ from their base pre-trained counterparts, we perform the token-distribution analysis proposed by Lin et al. (2023). Specifically, for a given instruction-response pair, the instruction $i = \{i_1, i_2, \cdots\}$ is first input to the aligned (or fine-tuned) model to obtain its response $r = \{r_1, r_2, \cdots\}$ via greedy decoding. Next, for each position t in the response, a 'context' at this position is defined as to be $x_t = i + \{r_1, \cdots, r_{t-1}\}$. This "context" is then input to the base model to obtain its probability distribution for predicting the next token at position t, $\mathrm{P_{base}}$. The probability distribution of the token at position t obtained from the aligned model is denoted as $\mathrm{P_{align}}$. We then calculate three metrics: **(1) KL Divergence** between $\mathrm{P_{base}}$ and $\mathrm{P_{align}}$, **(2) Base Probability**: $\mathrm{P_{base}}$ of the token at t with the maximum $\mathrm{P_{align}}$ value **(3) Base Rank**: Rank in $\mathrm{P_{base}}$ of the token at t with the maximum $\mathrm{P_{align}}$ value. With the base rank denoted as η, the unshifted, marginal and shifted tokens are defined as when $(\eta = 1)$, $(1 < \eta \leq 3)$ and $(\eta > 3)$ respectively. Figure 1 illustrates how the three metrics evolve. These metrics are averaged across all response tokens and plotted against the varying sizes of the IT dataset used for fine-tuning. We summarize our findings as follows: **(1)** Fine-tuning using LFT results in a minimal shift in token distribution, i.e., given a prior "context", a model fine-tuned using LFT generally

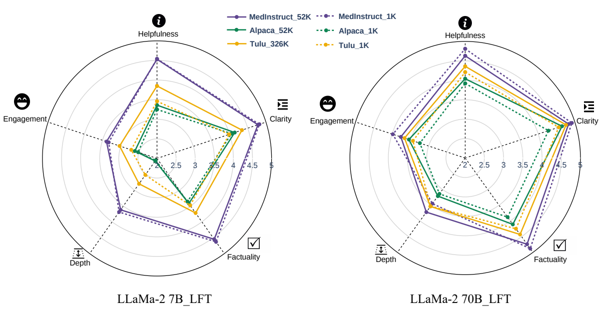


Figure 2. **Dataset scaling is ineffective for LFT.** We show that with LFT, a model's performance does not significantly improve when the IT dataset is scaled to 52× or 326× its original size.

outputs tokens that a non-tuned base model would output. This further indicates that LFT-generated responses align with the model's pre-trained knowledge. **(2)** Scaling the size of the IT dataset has a negligible effect on the extent of the token distribution shift observed with LFT. **(3)** In contrast, SFT leads to a significantly greater shift in token distribution, which suggests a substantial deviation from the pre-trained knowledge in its responses. **(4)** Larger models show a reduced distribution shift for both LFT and SFT.

Finding 2. **LFT only acts as a response initiator, while most of the answer comes from pre-trained knowledge.** In Fig. 4, we analyze the KL Divergence between the fine-tuned models and their base counterparts. This analysis focuses on the divergence in the initial 5% and the subsequent 95% of tokens of each sentence of a response, averaged across all sentences of all responses from models fine-tuned on Alpaca, Tulu-V2-Mix, and MedInstruct datasets with both low-resource (1k) and high-resource (52k) splits. Our observations indicate a higher KL divergence in the first 5% of the tokens for LFT, which then decreases sharply. In contrast, the decrease in KL divergence is less pronounced for SFT. This implies that LFT mainly learns sentence or fact initiation with a higher distribution shift and thereby novel tokens predominantly in the initial parts of each sentence in the response. On the other hand, SFT exhibits a more substantial and uniform distribution shift across the full span of the sentences. A few examples can be seen in the bottom section of Fig. 7. Our finding provides a more granular and in-depth understanding of the finding by Lin et al. (2023) who show that IT majorly affects the earlier tokens of a response. Additionally, we show that the RLHF-based IT methods discussed in Lin et al. (2023) are more similar in their behaviour to LFT than SFT and SFT behaves differently from both in terms of distribution shift. We later investigate whether these novel tokens in SFT, caused by this distribution shift, translate to new knowledge and eventually enhance response quality. Additionally, we emphasize that comparing LFT and RLHF-based IT to highlight the exact benefits of the latter should be an exciting space.

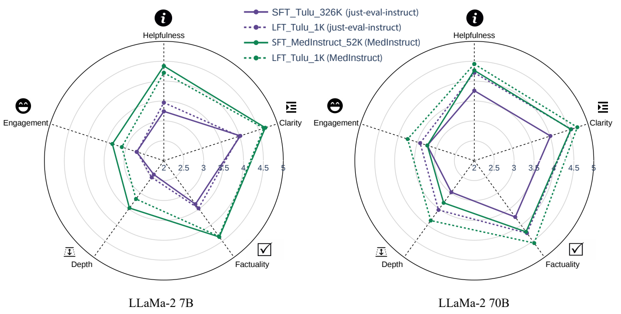


Figure 3. **Pre-trained knowledge outperforms new knowledge learned with SFT.** We show that LFT with only 1000 samples outperforms SFT on 326× and 52× more samples on factuality and usefulness on both an open- (just-eval-instruct $_{\mathbf{1k}}$) and knowledge-intensive-domains (MedInstruct-test $_{\mathbf{216}}$). While responses by the LFT model are most aligned with pre-trained knowledge, responses by the SFT models output new knowledge learned from IT.

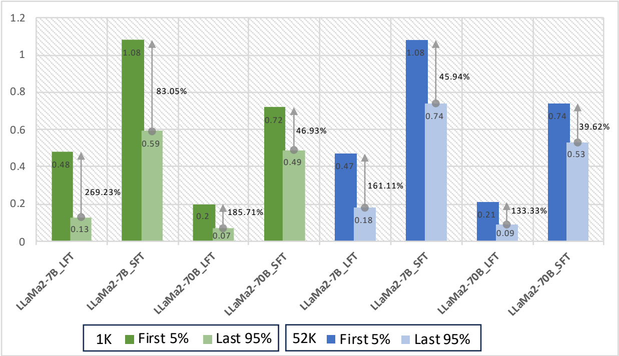
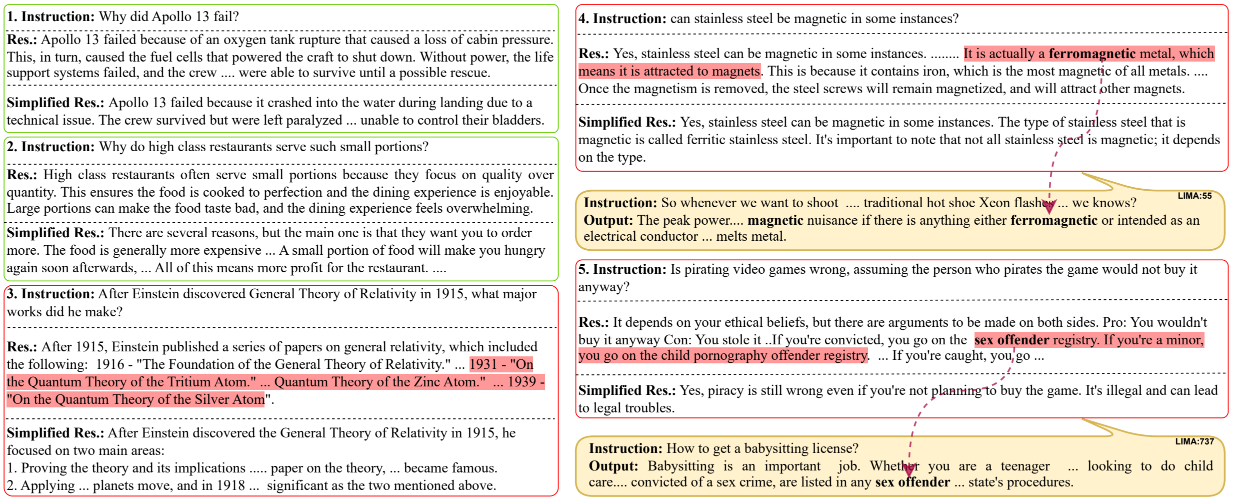


Figure 4. **KL Divergence analysis between the probability distribution of response tokens from fine-tuned models and their pre-trained only counterparts.** We plot separately for tokens in the first 5% and the remaining 95% for each sentence in the response. LFT primarily learns to initiate individual sentences in the response, showing a higher distribution shift and, thereby, the introduction of novel tokens predominantly in the initial parts of every sentence in the response. SFT exhibits a more substantial and uniform distribution shift across the full span of the response.

Finding 3. **Scaling is ineffective for LFT.** As LFT primarily learns response initiation, we next explore whether dataset scaling is effective with LFT in terms of improving response quality. In Fig. 2, we compare the performance of models, fine-tuned using LFT, on various IT datasets and their low-resource counterparts. Our findings show that scaling up the IT dataset by factors of 52× and even 326× does not significantly impact any of the five metrics in our automatic evaluation setup, including *factuality*. This trend is consistent across domains, including knowledge-intensive ones like medicine. These results suggest LFT does not learn any new knowledge or skill during the fine-tuning stage and that pre-trained knowledge is the primary contributor to its effectiveness. Our findings suggest that a few IT samples are sufficient for LFT to learn effective response initiation.



1. Instruction: Why did Apollo 13 fail?

Res.: Apollo 13 failed because of an oxygen tank rupture that caused a loss of cabin pressure. This, in turn, caused the fuel cells that powered the craft to shut down. Without power, the life support systems failed, and the crew .... were able to survive until a possible rescue.

Simplified Res.: Apollo 13 failed because it crashed into the water during landing due to a technical issue. The crew survived but were left paralyzed ... unable to control their bladders.

2. Instruction: Why do high class restaurants serve such small portions?

Res.: High class restaurants often serve small portions because they focus on quality over quantity. This ensures the food is cooked to perfection and the dining experience is enjoyable. Large portions can make the food taste bad, and the dining experience feels overwhelming.

Simplified Res.: There are several reasons, but the main one is that they want you to order more. The food is generally more expensive ... A small portion of food will make you hungry again soon afterwards, ... All of this means more profit for the restaurant. ....

3. Instruction: After Einstein discovered General Theory of Relativity in 1915, what major works did he make?

Res.: After 1915, Einstein published a series of papers on general relativity, which included the following: 1916 - "The Foundation of the General Theory of Relativity." ... 1931 - "On the Quantum Theory of the Tritium Atom." ... Quantum Theory of the Zinc Atom." ... 1939 - "On the Quantum Theory of the Silver Atom".

Simplified Res.: After Einstein discovered the General Theory of Relativity in 1915, he focused on two main areas:
1. Proving the theory and its implications ..... paper on the theory, ... became famous.
2. Applying ... planets move, and in 1918 ... significant as the two mentioned above.

4. Instruction: can stainless steel be magnetic in some instances?

Res.: Yes, stainless steel can be magnetic in some instances. .... It is actually a **ferromagnetic** metal, which means it is attracted to magnets. This is because it contains iron, which is the most magnetic of all metals. .... Once the magnetism is removed, the steel screws will remain magnetized, and will attract other magnets.

Simplified Res.: Yes, stainless steel can be magnetic in some instances. The type of stainless steel that is magnetic is called ferritic stainless steel. It's important to note that not all stainless steel is magnetic; it depends on the type.

LIMA:55

Instruction: So whenever we want to shoot .... traditional hot shoe Xeon flashes ... we knows?
Output: The peak power.... **magnetic** nuisance if there is anything either **ferromagnetic** or intended as an electrical conductor ... melts metal.

5. Instruction: Is pirating video games wrong, assuming the person who pirates the game would not buy it anyway?

Res.: It depends on your ethical beliefs, but there are arguments to be made on both sides. Pro: You wouldn't buy it anyway Con: You stole it ..If you're convicted, you go on the **sex offender** registry. If you're a minor, you go on the child pornography offender registry. ... If you're caught, you go ...

Simplified Res.: Yes, piracy is still wrong even if you're not planning to buy the game. It's illegal and can lead to legal troubles.

LIMA:737

Instruction: How to get a babysitting license?
Output: Babysitting is an important job. Whether you are a teenager ... looking to do child care.... convicted of a sex crime, are listed in any **sex offender** ... state's procedures.

Figure 5. ***Style Imitation*** **affects response quality.** Instructions 3, 4, and 5 illustrate examples of instances where the model, initially responding accurately, proceeds to generate hallucinated content. The suspected cause is *style imitation*, a process where the model, striving for lengthier, more detailed responses, fabricates information when it lacks sufficient knowledge. This hypothesis is further confirmed by comparing the responses to responses by another model fine-tuned on the simplified version of the same IT dataset. The hallucinations in Instructions 3 and 4 are not invented but are instead drawn from the IT dataset, a subject explored more comprehensively in Section 5. Moreover, Instruction 1 exemplifies the model's ability to generate an elaborate answer when it has adequate knowledge of the subject, whereas Instruction 2 demonstrates how merely imitating the style can alter the nature of a response to a reasoning task. Every response is also accompanied by Simplified Res., a response from a model fine-tuned on the same IT dataset but with simplified responses (detailed in Section 4). Notice how the Simplified Res. is less prone to hallucination by providing a brief response.

Finding 4. **Pre-trained knowledge (currently) dominates.** Our findings indicate that IT via LFT primarily facilitates response initiation rather than augmenting new knowledge, and most of the response is based on pre-trained knowledge. In contrast, the substantial token distribution shift observed in SFT suggests learning new knowledge during fine-tuning. To assess if this newly acquired knowledge translates into improved response quality, we conduct the following evaluations, as shown in Fig. 3: **(1)** We compare the performance of models of varying sizes when fine-tuned with LFT on just 1k samples of the IT dataset and the same models fine-tuned with SFT on a dataset 326× larger on just-eval-instruct $_{\mathbf{1k}}$. As we know from earlier findings, the LFT model relies entirely on pre-trained knowledge, while the SFT model maximizes learning from the IT dataset. The results show that the LFT model outperforms in terms of factuality and usefulness, suggesting that even extensive IT does not significantly introduce useful or factual knowledge to the model. **(2)** In a more extreme scenario, we compare the performance of the same LFT model, trained on open-domain instruction-response pairs, with a domain-specific model trained on MedInstruct $_{\mathbf{52k}}$ against MedInstruct-test $_{\mathbf{216}}$. Remarkably, even in this case, the LFT model performs better. Table 13 shows that LLaMa-2 $_{\mathbf{7B}}$-chat models (the already IT-ed versions open-sourced by META) outperforms all our fine-tuned models. We attribute this to RLHF or a better and larger IT dataset used for fine-tuning.

Key Takeaways: LFT, even at scale, largely relies on pre-trained knowledge without acquiring new information. In contrast, SFT's notable token distribution shift suggests new knowledge acquisition. However, responses by LFT that are grounded in pre-trained knowledge consistently outperform those based on newly learned information from SFT, indicating that SFT tends to diminish overall knowledge quality.

4. Pattern Copying (often) Hurts Performance

In our exploration of the consequences of significant token distribution shifts during SFT, we take a closer look at the concept of pattern copying. We define pattern copying as the scenario when an LLM learns to mimic the characteristics of responses in the IT dataset. We sub-categorize pattern copying into two distinct types: **(1)** *Tone Imitation:* In this case, generated responses tend to use tokens from the IT dataset. These can be stylistic tokens or normal ones. **(2)** *Style Imitation:* The responses mirror the wider stylistic traits present in the IT dataset. For example, if the IT data includes comprehensive, well-structured, and lengthy answers, the LLM is likely to exhibit similar traits in its responses. LLMs have been shown to learn such characteristics (Gudibande et al., 2023; Lin et al., 2023). We show that token distribution shifts following IT, and more specifically, the use of tokens that deviate from pre-trained knowledge, are indicative of the model's level of adaptation to the IT dataset's specifics. Additionally, SFT and style imitation lead the model to inaccurately include tokens from the IT dataset in its responses, negatively affecting response quality.

Finding 1. **LFT and SFT learn tone imitation differently.** As seen in Section 3, LFT primarily learns response initiation, and most of the response comes from pre-trained knowledge. On the other hand, SFT experiences high degrees of new token usage. This leads us to investigate if LFT learns *tone imitation* at all and the differences in *tone imitation* between models fine-tuned using the two training paradigms. Fig. 12a and 12b illustrate token distributions of common tokens at shifted and marginal positions for LLaMa-2 $_{\textbf{7B}}$_SFT_Alpaca $_{\textbf{52k}}$ and LLaMa-2 $_{\textbf{7B}}$_LFT_Alpaca $_{\textbf{52k}}$. Analyzing tokens at shifted positions allows us to understand how IT affects a model to output different tokens than it would generally have without IT. As we clearly see, for LFT, shifts occur primarily in style tokens ("typically") and response initiation tokens ("However"). On the other hand, for SFT, shifts occur in all kinds of tokens. To investigate the source of these tokens further, we perform a string search of the shifted tokens in the IT dataset. To our surprise, we found ≈81.2% of the words that start with shifted tokens and 66.7% that start with marginal tokens borrowed from the IT dataset itself. This indicates that with SFT, models show increased borrowing of tokens from the IT dataset for response generation. We may attribute this to over-fitting on the IT dataset. A study in Section 5 further shows that these tokens are often borrowed inaccurately, leading to hallucinations.

Finding 2. **Style imitation can hurt response quality.** We now investigate if *style imitation* can affect response quality. We are motivated by the finding that a positive correlation exists between the length of responses in the IT dataset and the length of responses output by the fine-tuned LLM on our evaluation set (detailed results in Table 11). To achieve this, we utilize the LIMA IT dataset, which comprises responses from community Q&A forums known for their comprehensiveness, expertise, and length (refer to Table 11). When LLaMa is fine-tuned on the LIMA dataset with SFT, it produces lengthy and detailed responses, even when pre-trained knowledge might be insufficient. This often leads to hallucinations as the model strives to generate extended answers. Figure 5 showcases examples of this, with each box presenting a response from LLaMa-2 $_{\textbf{7B}}$_SFT_LIMA $_{\textbf{1K}}$ alongside a more concise version. We summarize our findings on style imitation: **(1)** As seen in Instruction 1., style imitation can sometimes enhance response quality without leading to hallucinations, particularly when the model has sufficient knowledge of the subject. **(2)** As seen in Instruction 2., style imitations alone can alter the nature of a response to open-ended reasoning instructions. **(3)** The model initially provides factual information but then resorts to hallucination to prolong the response or fact in the absence of adequate knowledge about the subject. These hallucinations may include randomly generated facts or content derived from the IT dataset, as seen earlier and discussed further in Section 5.

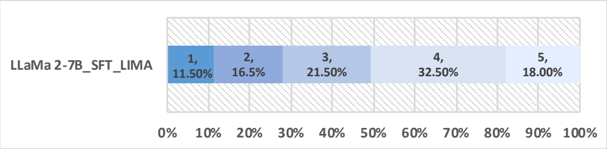
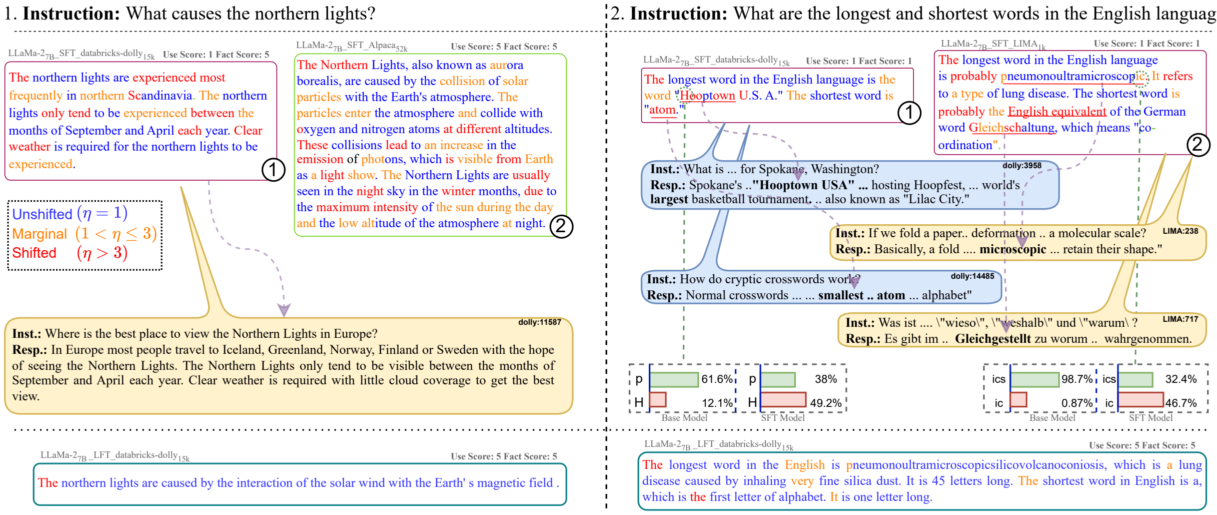


Figure 6. **Human study comparing responses of a model fine-tuned on LIMA $_{\textbf{1K}}$ and LIMA-Simple $_{\textbf{1k}}$. 1.** Both responses are accurate, but the former is more detailed and preferred over the latter. **2.** Both responses are accurate but different information-wise. **3.** Both responses are completely inaccurate. **4.** The former hallucinated facts in an attempt to prolong the response, while the latter did not. **5.** The former hallucinated in the core facts, while the latter did not. Qualitative examples in Section B.

Proposed Solution: **Simplifying responses in the IT dataset.** One possible solution to mitigate the hallucination problem caused by style imitation is to employ LFT instead of SFT, as LFT tends to learn only stylistic elements. However, as seen in Finding 3 of Section 3, LFT doesn't scale effectively, and SFT may act as a knowledge enhancer at larger scales (evident from LLaMa-2 $_{\textbf{7B}}$_LFT_Tulu-V2-Mix $_{\textbf{326k}}$ in Fig. 3). We propose an approach leveraging the strengths of both LFT and SFT. Given that LLMs possess ample pre-training knowledge for accurate response generation (Finding 4, Section 3) but struggle with comprehensive answering in pattern-copying mode, we hypothesize that SFT on IT datasets with concise but accurate responses can reduce hallucinations. We employ GPT-4 to simplify LIMA $_{\textbf{1K}}$, creating concise responses by removing extraneous information and term it as LIMA-Simple $_{\textbf{1k}}$. This model is compared with the original LLaMa-2 $_{\textbf{7B}}$_LFT_LIMA $_{\textbf{1K}}$ in terms of hallucination tendencies in Fig. 5. The results show a significant reduction in hallucinations beyond being concise. Quantitative results in Table 13 (rows 49-50) confirm that while the simplified model may lack depth, it surpasses the original in factuality and helpfulness. We additionally illustrate two special cases: **(1)** Instruction 1 illustrates a case where the model actually possessed enough knowledge to answer comprehensively and provides a more factual response with greater depth than its simplified instruction. **(2)** Instruction 2 demonstrates how adopting pattern copying from datasets of varied types results in divergent responses to a single open-ended reasoning-based instruction. Fig. 6 illustrates the results of a human study conducted by four expert human evaluators who manually compared the responses of LLaMa-2 $_{\textbf{7B}}$_SFT_LIMA $_{\textbf{1K}}$ and LLaMa-2 $_{\textbf{7B}}$_SFT_LIMA-Simple $_{\textbf{1k}}$ against 5 pre-defined categories. We show that simplifying responses in IT datasets for mod-



1. **Instruction:** What causes the northern lights?

LLaMa-2$_{7B}$_SFT_databricks-dolly$_{15k}$ **Use Score: 1 Fact Score: 5**

The northern lights are experienced most frequently in northern Scandinavia. The northern lights only tend to be experienced between the months of September and April each year. Clear weather is required for the northern lights to be experienced. ①

LLaMa-2$_{7B}$_SFT_Alpaca$_{52k}$ **Use Score: 5 Fact Score: 5**

The Northern Lights, also known as aurora borealis, are caused by the collision of solar particles with the Earth's atmosphere. The particles enter the atmosphere and collide with oxygen and nitrogen atoms at different altitudes. These collisions lead to an increase in the emission of photons, which is visible from Earth as a light show. The Northern Lights are usually seen in the night sky in the winter months, due to the maximum intensity of the sun during the day and the low altitude of the atmosphere at night. ②

Inst.: Where is the best place to view the Northern Lights in Europe?
Resp.: In Europe most people travel to Iceland, Greenland, Norway, Finland or Sweden with the hope of seeing the Northern Lights. The Northern Lights only tend to be visible between the months of September and April each year. Clear weather is required with little cloud coverage to get the best view.

LLaMa-2$_{7B}$ _LFT_databricks-dolly$_{15k}$ **Use Score: 5 Fact Score: 5**

The northern lights are caused by the interaction of the solar wind with the Earth' s magnetic field .

2. **Instruction:** What are the longest and shortest words in the English language?

LLaMa-2$_{7B}$_SFT_databricks-dolly$_{15k}$ **Use Score: 1 Fact Score: 1**

The longest word in the English language is the word "Hooptown U.S. A." The shortest word is "atom." ①

LLaMa-2$_{7B}$_SFT_LIMA$_{1k}$ **Use Score: 1 Fact Score: 1**

The longest word in the English language is probably pneumonoultramicroscopic. It refers to a type of lung disease. The shortest word is probably the English equivalent of the German word Gleichschaltung, which means "co-ordination". ②

Inst.: What is ... for Spokane, Washington?
Resp.: Spokane's .."**Hooptown USA" ...** hosting Hoopfest, ... world's **largest** basketball tournament. .. also known as "Lilac City."

Inst.: If we fold a paper.. deformation .. a molecular scale?
Resp.: Basically, a fold .... **microscopic** ... retain their shape."

Inst.: How do cryptic crosswords work?
Resp.: Normal crosswords ... ... **smallest .. atom** ... alphabet"

Inst.: Was ist .... \"wieso\", \"weshalb\" und \"warum\ ?
Resp.: Es gibt im .. **Gleichgestellt** zu worum .. wahrgenommen.

LLaMa-2$_{7B}$ _LFT_databricks-dolly$_{15k}$ **Use Score: 5 Fact Score: 5**

The longest word in the English is pneumonoultramicroscopicsilicovolcanoconiosis, which is a lung disease caused by inhaling very fine silica dust. It is 45 letters long. The shortest word in English is a, which is the first letter of alphabet. It is one letter long.

Figure 7. **Illustration of hallucinations and their source of origin.** We illustrate responses from LLMs fine-tuned using SFT on different IT datasets on two instances from just-eval-instruct $_{\mathbf{1k}}$ and show that hallucinations originate from erroneous causal links between the response and instances in the IT dataset. Models trained using SFT tend to incorrectly borrow tokens from instances in the IT dataset describing similar *concepts*. **Left.** resp. ① from LLaMa-2 $_{\mathbf{7B}}$ fine-tuned on databricks-dolly $_{\mathbf{15k}}$ is not *useful*, although *factual*, as it describes places to view the northern lights but not what causes it. The answer was directly borrowed from the IT dataset. Alpaca $_{\mathbf{52k}}$ has no instances related to northern lights. **Right.** Both responses have 2 factual hallucinations each, and we show that each of them originates from different instances in the IT dataset. The shifted and marginal tokens (described in Section 3) indicate that hallucinations are learned during IT. *Note how models fine-tuned using LFT provide factual answers originating from pre-trained knowledge.*

els significantly reduces the chances of hallucination. Quantitative results are in Table 13.

> ***Key Takeaways:*** Pattern-copying increases model hallucinations. The comprehensiveness and depth of responses in the IT dataset should be phrased depending on the existing factual knowledge of the model. Additionally, pattern-copying makes the model borrow tokens from the IT dataset for responses.

5. Causal Analysis of Hallucinations

Earlier, we observed that inaccurately generated tokens in hallucinations by a model, fine-tuned using SFT, can often be traced back to the IT dataset itself. However, this raises two questions: Are such hallucinations specific to style imitation, and are inaccurate tokens borrowed random or causally driven from the IT dataset? To answer these questions, we perform a fine-grained analysis of hallucinations through the lens of causal analysis.

Finding 1. **SFT increases hallucinations in a model. These hallucinations originate from erroneous causal links between the training dataset and responses.** Fig. 7 illustrates 3 examples of hallucinations that do not occur in the pattern copying mode, i.e., the model hallucinates even in concise responses. Unlike the scenarios depicted in Figure 5, where hallucinations are associated with attempts to extend responses or facts, Instruction 2 in Figure 7 illustrates examples where the base pre-trained model, despite having the correct knowledge, hallucinates after IT. Similar to prior findings, these tokens originate from the IT dataset.

Next, we establish a formal framework for a detailed analysis of such hallucinations. We ask four human experts to identify hallucinations by marking spans in the responses that are either not *factual* or *useful* in relation to the instruction. A simple string search reveals that, on average, ≈72% of these marked phrases exist in the IT dataset, with ≈89% of the tokens in these phrases can be classified as shifted or marginal. Next, to determine whether these hallucinatory phrases are borrowed from any IT instance by chance, we look at these hallucinations from the lens of causal analysis, considering hallucinations as the *effect* and the specific properties or attributes of the instance in the IT dataset from where the phrases were incorrectly borrowed as the *cause*.

Four expert human evaluators manually reviewed all instances in the IT dataset where the hallucinated phrases occurred. Surprisingly, they discover a pronounced propensity for the models to respond using incorrectly borrowed tokens from instances in the IT dataset that describe analogous *concepts*. For instance, response ① for Instruction 2 in Fig. 7, which seeks information about the *largest* and *shortest* words in English, partially sources its content from an unrelated instruction regarding the *longest* basketball tournament and another concerning the *smallest* atom. Similarly, response ② from the same instruction appropriates its content from an instruction discussing a related concept of "languages". Finally, response ① for Instruction 2 illustrates

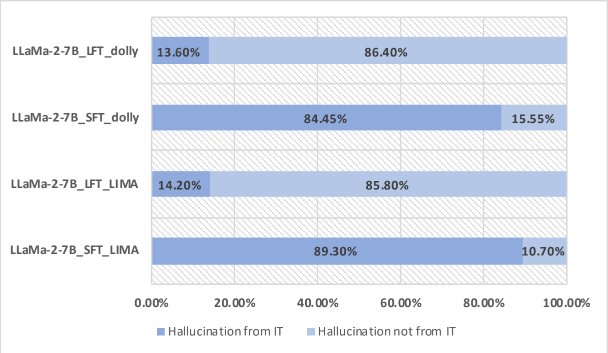


Figure 8. **Human study of hallucinations by LLaMa-2 $_{7B}$ trained on LIMA $_{1K}$ and databricks-dolly $_{15k}$ and evaluated on just-eval-instruct $_{1k}$.** Human evaluators found an average chance of 87% that a hallucinated phrase in a response originated from a causal relation in the IT dataset with SFT models. This phenomenon is much lesser in LFT models.

how the fine-tuned model generates a rephrased response from an instance in the IT dataset related to the same abstract concept – "about Northern lights", but unrelated to the input instruction, but rather related to an instance in the IT dataset. We also illustrate responses of models fine-tuned with LFT and show that responses aligned to pre-trained knowledge tend to output factually correct and useful answers. Table 8 shows that expert human evaluators found an average of 87% of 1000 randomly chosen hallucinated phrases in responses by models fine-tuned using SFT to be causally related to the IT dataset. This number is only 13.9% with LFT for 500 phrases.

Finding 2. **Additional findings.** We further report on our findings regarding model hallucinations: **(1)** Detecting the causal relationships behind hallucinations proves challenging, both through manual efforts and automated methods. Sole reliance on semantic or lexical analysis often falls short in tracing the exact instance within the IT dataset that influenced the hallucinated response or its segments. **(2)** Quantifying the degree to which concept similarity contributes leads to hallucinations is difficult. Our analysis yields an average semantic similarity score of 0.418 (described in Section D) between evaluation set instances and the corresponding instances from the IT dataset, identified by human evaluators as the source of hallucinated content. Furthermore, we observed that there is only a 61.6% likelihood that a *keyword* from the hallucinated response appears in its originating instance from the IT dataset (described in Section D). Additionally, we would like to highlight the fact that post SFT-based IT, the tendancy of LLMs to hallucinate with tokens from the IT dataset does not depend on the quality of the IT dataset and we observe this phenomenon throughout all IT datasets we employ in our analysis.

Qualitative Examples and Case Studies. Section B shows additional examples of hallucinations, including counterfactual analysis and case studies, where we also show that models even hallucinate similarly for procedural tasks like coding and summarization.

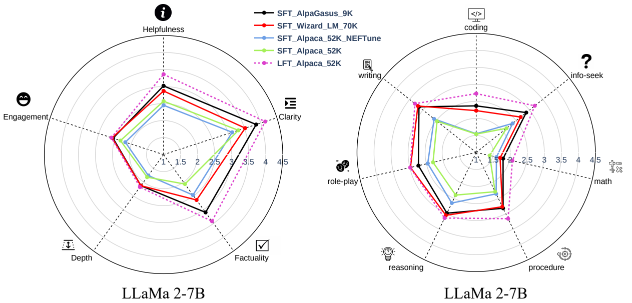


Figure 9. **Comparison of various methods proposed in literature to improve IT.** We show that a model trained using simple LFT outperforms all these methods across metrics and tasks.

> ***Key Takeaways:*** SFT increases hallucinations in a model by making it prone to incorrectly borrow tokens from the IT dataset for responding. These tokens are borrowed from instances describing similar concepts.

6. Methods to Improve IT are Ineffective

Several enhanced IT methods have been proposed in literature for improving model response quality. In this section, we study some common methods, including AlpaGasus (Chen et al., 2023) that employs dataset filtering, WizardLM (Xu et al., 2023a) that increases the complexity of the instructions, and NEFTune (Jain et al., 2023) that adds noise to embedding vectors while fine-tuning. Fig. 9 compares the performance of models fine-tuned with SFT using these methods (as suggested by the original papers), fine-tuned on a similar setting, with a model trained using LFT, which generates responses primarily with pre-trained knowledge. AlpaGasus and WizardLM are improved versions of the Alpaca dataset, and for NEFTune, we also employ Alpaca. As we see, though all methods outperform SFT on Alpaca (as also suggested by the original papers), the LFT model outperforms all these models in all cases, both when averaged across individual metrics and task types. This suggests that pre-trained knowledge still dominates, and the exact benefits brought about by these methods need deeper investigation. We hypothesize that the drop in performance is mainly brought about by the increased tendency of the models to hallucinate with tokens borrowed from the IT dataset, and we urge deeper investigation by the community to solve this problem before the benefits of these tricks could enhance response quality. Beyond the scope of this section, Table 13 (Rows 45-50) also shows that LFT performance only slightly improves with these methods.

7. Related Work

While extensive research has introduced new IT datasets, models, and enhancement methods (Zhang et al., 2023a), few studies examine IT's limitations. Concurrent work by Gudibande et al. (2023) shows that IT with datasets synthesized from powerful proprietary models only leads to imitating their style, not their knowledge. Similarly, Lin et al. (2023) shows that alignment only teaches style and proposes in-context learning as an alternative to IT that outperforms several tuned models. This can be attributed to our findings on LFT, where the models respond using pre-trained knowledge and do not lead to knowledge degradation like SFT. This supports the *superficial alignment hypothesis* by Zhou et al. (2023), positing that models gain knowledge in pre-training, with alignment shaping format used in user interactions. Finally, Kung & Peng (2023) show that models trained on simplified task definition or delusive examples achieve performance comparable to the ones trained on the original instructions. In contrast to all these works, we study the exact causes of these limitations, investigate pattern copying and hallucination from novel perspectives, and highlight the overlooked effectiveness of LFT that utilizes only pre-trained knowledge.

8. Conclusion

In this paper, we reveal various failure models of IT, including how LFT does not scale, how SFT and pattern-copying increase hallucinations, and how an LFT model outperforms various methods proposed in literature. As part of future work, we would like to propose a formal framework for detecting and mitigating hallucinations arising from SFT work on investigating novel methods of IT that can potentially improve model performance over pre-trained knowledge.

9. Limitations and Future Work

Our work has obvious limitations, including (1) Our analysis focuses solely on open-domain instruction following, and we acknowledge that fine-tuning for specific domains or tasks might enable models to gain new skills and knowledge. (2) Our analysis is limited to uni-modal language only IT, and (3) We do not study the effects of more advanced alignment methods like DPO (Rafailov et al., 2023) and RLHF and leave this for future work. (4) We do not explore retrieval-augmented generation, which decouples knowledge extraction from the model. (5) Finally, the findings of our paper are only confined to general-purpose IT datasets (datasets with an amalgamation of all kinds of general-purpose everyday instruction-response pairs) and IT datasets tailored to improve specific tasks (e.g., IT datasets tailored for imitating step-wise thinking to improve math reasoning (Chern et al., 2023). may not obey our findings.

10. Reproducibility

All LLMs evaluated in this paper were trained using LLaMA-Factory (Zheng et al., 2024b). For evaluation, we provide all prompts in Section A. We would like to emphasize that the exact replication of the numbers in our paper would require employing the prompt together with gpt-4-1106-preview, which is subject to availability by OpenAI. This is similar to a wealth of prior work that employs LLM-as-a-judge for free-form LLM response evaluation. However, if another version is employed, most trends and the core findings will still hold. For any questions, we request that the readers contact the corresponding author.

Impact Statement

This paper explores the limitations of Instruction Tuning for Large Language Models, which have implications that extend beyond the immediate scope of artificial intelligence research and development. By highlighting the constraints of current IT practices, our findings encourage the development of more robust and reliable conversational agents that could be employed across various sectors, including education, customer service, and accessibility technologies.

One significant impact is on the ethical use of LLMs. Our research emphasizes the importance of accuracy and factual representation in responses, reducing the risk of misinformation that can stem from hallucinations and knowledge degradation. This is particularly critical in domains where trust and veracity are paramount, such as healthcare and news dissemination.

In academia, our call for further investigation into IT's limitations could spur a new direction of research that focuses on understanding the fundamental workings of LLMs rather than just their superficial performance improvements.

References

Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Beeching, E., Fourrier, C., Habib, N., Han, S., Lambert, N., Rajani, N., Sanseviero, O., Tunstall, L., and Wolf, T. Open llm leaderboard. `https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard`, 2023.

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Chan, C.-M., Chen, W., Su, Y., Yu, J., Xue, W., Zhang, S., Fu, J., and Liu, Z. Chateval: Towards better llm-based evaluators through multi-agent debate. *arXiv preprint arXiv:2308.07201*, 2023.

Chen, L., Li, S., Yan, J., Wang, H., Gunaratna, K., Yadav, V., Tang, Z., Srinivasan, V., Zhou, T., Huang, H., and Jin, H. Alpagasus: Training a better alpaca with fewer data, 2023.

Chern, E., Zou, H., Li, X., Hu, J., Feng, K., Li, J., and Liu, P. Generative ai for math: Abel. `https://github.com/GAIR-NLP/abel`, 2023.

Conover, M., Hayes, M., Mathur, A., Xie, J., Wan, J., Shah, S., Ghodsi, A., Wendell, P., Zaharia, M., and Xin, R. Free dolly: Introducing the world's first truly open instruction-tuned llm, 2023.

Ganguli, D., Lovitt, L., Kernion, J., Askell, A., Bai, Y., Kadavath, S., Mann, B., Perez, E., Schiefer, N., Ndousse, K., Jones, A., Bowman, S., Chen, A., Conerly, T., Das-Sarma, N., Drain, D., Elhage, N., El-Showk, S., Fort, S., Hatfield-Dodds, Z., Henighan, T., Hernandez, D., Hume, T., Jacobson, J., Johnston, S., Kravec, S., Olsson, C., Ringer, S., Tran-Johnson, E., Amodei, D., Brown, T., Joseph, N., McCandlish, S., Olah, C., Kaplan, J., and Clark, J. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned, 2022.

Gudibande, A., Wallace, E., Snell, C., Geng, X., Liu, H., Abbeel, P., Levine, S., and Song, D. The false promise of imitating proprietary llms. *arXiv preprint arXiv:2305.15717*, 2023.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Huang, Y., Gupta, S., Xia, M., Li, K., and Chen, D. Catastrophic jailbreak of open-source llms via exploiting generation, 2023.

Ivison, H., Wang, Y., Pyatkin, V., Lambert, N., Peters, M., Dasigi, P., Jang, J., Wadden, D., Smith, N. A., Beltagy, I., et al. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*, 2023.

Jain, N., Chiang, P.-y., Wen, Y., Kirchenbauer, J., Chu, H.-M., Somepalli, G., Bartoldson, B. R., Kailkhura, B., Schwarzschild, A., Saha, A., et al. Neftune: Noisy embeddings improve instruction finetuning. *arXiv preprint arXiv:2310.05914*, 2023.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Kung, P.-N. and Peng, N. Do models really learn to follow instructions? an empirical study of instruction tuning. *arXiv preprint arXiv:2305.11383*, 2023.

Li, X., Zhang, T., Dubois, Y., Taori, R., Gulrajani, I., Guestrin, C., Liang, P., and Hashimoto, T. B. Alpacaeval: An automatic evaluator of instruction-following models. *GitHub repository*, 2023a.

Li, Y., Bubeck, S., Eldan, R., Del Giorno, A., Gunasekar, S., and Lee, Y. T. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023b.

Lin, B. Y., Ravichander, A., Lu, X., Dziri, N., Sclar, M., Chandu, K., Bhagavatula, C., and Choi, Y. The unlocking spell on base llms: Rethinking alignment via in-context learning. *arXiv preprint arXiv:2312.01552*, 2023.

Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., and Zhu, C. G-eval: Nlg evaluation using gpt-4 with better human alignment, may 2023. *arXiv preprint arXiv:2303.16634*, 6, 2023.

OpenAI. Chatgpt: Optimizing language models for dialogue. `https://openai.com/blog/chatgpt`, 2023. Accessed: 2024-01-18.

Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

Reimers, N. and Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., and Hashimoto, T. B. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Wang, Y., Kordi, Y., Mishra, S., Liu, A., Smith, N. A., Khashabi, D., and Hajishirzi, H. Self-instruct: Aligning language models with self-generated instructions. In

Rogers, A., Boyd-Graber, J., and Okazaki, N. (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`.

Wei, J., Bosma, M., Zhao, V., Guu, K., Yu, A. W., Lester, B., Du, N., Dai, A. M., and Le, Q. V. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=gEZrGCozdqR`.

Xu, C., Sun, Q., Zheng, K., Geng, X., Zhao, P., Feng, J., Tao, C., and Jiang, D. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*, 2023a.

Xu, W., Wang, D., Pan, L., Song, Z., Freitag, M., Wang, W. Y., and Li, L. Instructscore: Towards explainable text generation evaluation with automatic feedback. *arXiv preprint arXiv:2305.14282*, 2023b.

Zhang, B., Liu, Z., Cherry, C., and Firat, O. When scaling meets LLM finetuning: The effect of data, model and finetuning method. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=5HCnKDeTws`.

Zhang, S., Dong, L., Li, X., Zhang, S., Sun, X., Wang, S., Li, J., Hu, R., Zhang, T., Wu, F., et al. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*, 2023a.

Zhang, X., Tian, C., Yang, X., Chen, L., Li, Z., and Petzold, L. R. Alpacare:instruction-tuned large language models for medical application, 2023b.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

Zheng, C., Zhou, H., Meng, F., Zhou, J., and Huang, M. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*, 2024a. URL `https://openreview.net/forum?id=shr9PXz7T0`.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., Zhang, H., Gonzalez, J. E., and Stoica, I. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL `https://openreview.net/forum?id=uccHPGDlao`.

Zheng, Y., Zhang, R., Zhang, J., Ye, Y., Luo, Z., and Ma, Y. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*, 2024b. URL `http://arxiv.org/abs/2403.13372`.

Zhou, C., Liu, P., Xu, P., Iyer, S., Sun, J., Mao, Y., Ma, X., Efrat, A., Yu, P., YU, L., Zhang, S., Ghosh, G., Lewis, M., Zettlemoyer, L., and Levy, O. LIMA: Less is more for alignment. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=KBMOKmX2he`.

A. Prompts

In this section, we illustrate the exact prompts used in our paper. Fig. 11 illustrated the prompt used for automatic GPT-4 evaluation, which we borrow directly from Lin et al. (2023). In Fig. 10, we illustrate the prompt used for our proposed method of simplifying responses in the IT dataset to reduce model halluicnations.

```
# Simplification Prompt

↳I want you to act as a Prompt Rewriter. I will provide you with an query-response pair and you will need to simplify
the response.

##Query: {instruction}

##Response: {response}

Simplify the response to only contain enough information that is helpful and factual with respect to the given
instruction. Remove redundant information that increases the depth of the answer, and you should rewrite the response
with brevity. The simplified response should be concise and to the point. If the response cannot be simplified, for
example, the response is already concise, or it is an instruction that asks to code or summarize, do not do anything
with it. Do not change the formatting of the response.

##Simplified Response:
```

Figure 10. Prompt for response simplification in IT datasets.

```
# Evaluation Prompt

↳ Please act as an impartial judge and evaluate the quality of the responses provided. You will rate the quality of
the output on multiple aspects, such as Helpfulness, Clarity, Factuality, Depth, and Engagement.

##Query: {instruction}

##Output: {prediction}

##Evaluate
### Aspects
- Helpfulness: Rate the response based on how well it addresses the users query and provides a relevant solution. A
score of 5 indicates the answer fully aids the user, while a 1 suggests it offers little to no help.
- Clarity: Rate the response based on how well-structured it is, with ideas presented in a clear and coherent manner. A
high score of 5 means the answer is clear and logically structured, while a 1 suggests a disjointed or confusing reply.
- Factuality: Evaluate the factual accuracy and truthfulness of the information provided. A perfect 5 indicates the
information is entirely correct and accurate, while a 1 suggests it has significant factual errors.
- Depth: Determine the level of detail and thoroughness in the response. A score of 5 means the answer delves deeply
into the topic, while a 1 indicates it barely scratches the surface.
- Engagement: Assess how engaging and natural the response sounds in a conversational context. A high score of 5
reflects a response that feels engaging and human-like in its tone, while a 1 indicates a robotic or boring reply. ,

### Format Given the query, please rate the quality of the output by scoring it from 1 to 5 , individually on **each
aspect**.
- 1: strongly disagree
- 2: disagree
- 3: neutral
- 4: agree
- 5: strongly agree

Now, please output your scores and a short rationale below in a json format by filling in the placeholders in []: {
'helpfulness': { 'reason': '[your rationale]', 'score': '[score from 1 to 5]' }, 'clarity': { 'reason': '[your
rationale]', 'score': '[score from 1 to 5]' }, 'factuality': { 'reason': '[your rationale]', 'score': '[score from 1 to
5]' }, 'depth': { 'reason': '[your rationale]', 'score': '[score from 1 to 5]' }, 'engagement': { 'reason': '[your
rationale]', 'score': '[score from 1 to 5]' } }"
```

Figure 11. Prompt for multi-aspect evaluation of model responses on open-domain chat instructions borrowed from Lin et al. (2023).

B. Qualitative Examples

Table 1, 2, 3, 4 and 5 show examples of cases 1, 2, 3, 4, and 5 defined in Fig. 6. Precisely, we compare responses on just-eval-instruct $_{\mathbf{1k}}$, by LLaMa-2 $_{\mathbf{7B}}$ trained on LIMA $_{\mathbf{1K}}$ and LIMA-Simple $_{\mathbf{1k}}$. We provide details on how LIMA-Simple $_{\mathbf{1k}}$ was constructed in Section 4.

Table 6, 7 and 8 show examples of hallucinations by LLaMa-2 $_{\mathbf{7B}}$, fine-tuned on databricks-dolly $_{\mathbf{15k}}$ with SFT. We show hallucinations for responses on just-eval-instruct $_{\mathbf{1k}}$, and their exact origins, i.e., the IT instance from the IT dataset the model was fine-tuned on. We present a causal analysis of such hallucinations in Section 5, where we hypothesize that the model borrows these tokens incorrectly from IT instances with similar concepts. Additionally, we show two more instances from the IT dataset with similar concepts which the model could have borrowed the hallucinated tokens from. Through this, we show that among instances with similar concepts, the fine-tuned model may borrow tokens from any instance randomly. Finally, we also show the results of a counterfactual study by human annotators, who asked the same question to the model but in varied ways in an attempt to escape the causal borrowing of tokens. The counterfactual questions were designed with care to explicitly remove the causal concepts identified in the earlier stage. For example, in the 3rd example in Table 7, the annotators changed the concept of *creation* to include *credited* in the IT, leading to a correct answer. This further amplifies our hypothesis of the model borrowing tokens only from IT instances with similar *concepts*. For this study, we show both successful and failed attempts by human annotators.

Table 12 shows examples of hallucinations in non-information-seeking instructions, like procedural (e.g., summarization) and coding instructions. We highlight two specific points through examples in this table: (1) Causal links for hallucinations can even be seen in non-information-seeking instructions, where models borrow tokens inaccurately from the IT dataset to generate responses, and (2) For open-ended instructions, which do not have a definite answer, like the ones shown in the table, models tend to borrow tokens or facts learned from the IT dataset itself for responding. This can be noticed for accurate responses and not just for hallucinations.

Finally, we also compare perfectly accurate answers by models trained on different datasets in Table 9, where we show that models trained on different datasets can generate quite different stylistic answers for the same question and still be accurate. Through this, we can attribute IT as a *knowledge extractor*, and we leave further investigation as part of future work.

1. Instruction: Who created the Superman cartoon character?

Res.: Superman was created by writer Jerry Siegel and artist Joe Shuster. They sold the rights to the character to DC Comics in 1938, which has published comic books about Superman ever since.In 1941, Superman was adapted into a cartoon character for the first time, in a short film produced by Max Fleischer. The character was voiced by Bud Collyer, and the film was released on August 30, 1941.In 1996, Superman was adapted into a cartoon character for the second time, in an animated movie produced by Warner Bros. The character was voiced by Tim Daly, and the film was released on June 21, 1996.

Simplified Res.: Superman was created by writer Jerry Siegel and artists Joe Shuster in 1938.

2. Instruction: What is the difference between HTML and JavaScript?

Res.: Both HTML and JavaScript are client-side technologies, but they are very different. HTML is a language for describing the structure of a web page. It is used by the browser to render the page. JavaScript is a language for executing code on the client side. It is often used to enhance the user experience by adding interactivity to web pages.In a few words: HTML is the what; JavaScript is the how.The most common use of JavaScript is to add event handlers to web pages. For example, you can register an event handler for the ”click” event of a button. When the button is clicked, your event handler function is called. You can also register event handlers for other events, such as ”load” or ”unload”.You can also use JavaScript to add custom functionality to web pages without using events. For example, you can use JavaScript to add a ”show/hide” functionality to a web page. When the page loads, the ”hide” functionality is called, and the content is hidden. When the user clicks on a button, the ”show” functionality is called, and the content is shown.

Simplified Res.: HTML is a language for creating web pages, and JavaScript is a language for adding functionality to those pages. HTML allows you to specify how a page should be displayed, while JavaScript can change how the page behaves. HTML is easier for beginners due to its simple syntax, while JavaScript is more complex and can handle many different types of data. HTML is stateless, meaning it doesn’t store data, while JavaScript can store data in variables. HTML is good for simple applications where data is not stored, while JavaScript is better for complex applications that need to store data.

Table 1. Comparison of LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA $_{\mathbf{1K}}$ and LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA-Simple $_{\mathbf{1k}}$. We show two examples of Case 1: Responses by both responses are accurate, but response by LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA $_{\mathbf{1K}}$ is more detailed and preferred over LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA-Simple $_{\mathbf{1k}}$.

1. Instruction: How do you become an author?

Res.: There is no one way to become an author. Some people are born with the talent and others develop it. Here are some steps you can take to become an author:1. Read a lot. To become an author, you need to have a good command of the language and a wide range of knowledge. The best way to get both is to read a lot. Read fiction and non-fiction books on a variety of subjects.2. Write a lot. The best way to become an author is to write a lot. Start with short stories and work your way up to novels. Write about things you know and things you want to learn about.3. Get feedback. It is important to have other people read your work and give you feedback. You can get feedback from friends, family, and even strangers online.4. Find your voice. It is important for authors to have a distinct voice. This means finding the style and tone that is unique to you.5. Consider your audience. When you are writing, keep your audience in mind. Will you be writing for children or adults? Do you want to write fiction or non-fiction?6. Set a schedule. It is important to stick to a schedule if you want to become an author. You should set aside time each day or each week to work on your writing.7. Be patient. It can take a long time to become an author. Don't get discouraged if your work isn't published right away.8. Keep learning. Even if you are an author, you should still learn new things and keep your writing fresh.9. Consider self-publishing. These days, it is very easy to self-publish a book. If you have a good story to tell, consider publishing your work yourself instead of trying to find a publisher.10. Get a mentor. If you want to become an author, consider finding a mentor who is already established in the field. A mentor can help you learn the ropes and give you advice along the way.

Simplified Res.: To become an author, you need to write a book. Here's how to do it:1. Choose Your Topic - Pick a topic you're interested in or know well. - Find out what's already been written on the topic to avoid duplication. - Consider writing about a personal experience or hobby. - Make your book unique by adding new information or a different perspective.2. Plan Your Book - Develop a clear outline of your book's content. - Identify the main ”truths” you've found in your research. - Use a simple, clear writing style. - Avoid complex scientific or legal jargon. - Make your book fun and engaging.3. Write Your Book - Use notes with key points to make writing easier. - Focus on the main ”truths” you've found. - Use simple, clear language. - Make your book engaging and fun.4. Edit Your Book - Ensure each chapter is clear and concise. - Remove unnecessary details and jargon. - Add more information as needed to support your main ”truths.”5. Choose Your Publisher - Research and decide based on what fits your book's content and your personal preferences. - Consider self-publishing if you want full control over your book. - Be aware of potential costs and time investments.6. Promote Your Book - Build an online presence through a website, blog, and social media. - Network with professionals in your industry. - Get book reviews from experts. - Consider paid advertising if you want to reach a larger audience.

2 Instruction: Did Facebook corporation change its name?

Res: Yes, Facebook rebranded itself as Meta in October 2021.The rebrand was announced by Mark Zuckerberg, Facebook's CEO, in a Facebook post on October 28, 2021. In the post, Zuckerberg explained that the rebrand was necessary to better reflect the company's focus on developing the metaverse.Prior to the rebrand, the company was known as Facebook Inc. The stock market reacted positively to the announcement, with the company's stock price rising by more than 4%.

Simplified Res.: Yes, Facebook changed its name to Meta in 2021. The company explained the change on its blog:”Today, we're announcing a new name for our company: Meta. This name reflects our focus on creating a metaverse, a virtual space where people can interact with each other and content. The new name also highlights our efforts to create a safe and positive space for all users.

Table 2. Comparison of LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA $_{\mathbf{1K}}$ and LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA-Simple $_{\mathbf{1k}}$. We show two examples of Case 2: Responses by both models are accurate but different information-wise or just a rephrased version of each other.

1. Instruction: How did US states get their names?

Res.: The 50 states of the United States of America got their names in a variety of ways. Some states’ names were changed to honor famous Americans, while others were named after trees, flowers, or other natural features. Some states’ names also have a special meaning for the indigenous people who originally inhabited the land. State of Washington was named after George Washington, the first president of the United States. State of Oregon was named after the river Orinoco, which was named after the city of Orl0̆0e9ans in France, the birthplace of French explorer Robert de La Salle, who explored the area in 1763. State of California was named after the Spanish missionary Franciscan friar Junoedpero Serra, who founded the first Spanish mission in the state in 1769. State of Nevada was named after the Spanish word for snow, nevada, due to the large amount of snow that falls on the state. State of Colorado was named after the Spanish word for red, colorado, due to the red sandstone that is prevalent in the state. State of New Mexico was named after the Spanish word for New Mexico, which was originally spelled ”Nuevo M0̆0e9xico” in Spanish. State of Arizona was named after the Spanish word for the Phoenix, the mythical bird that is associated with the state. State of Utah was named after the Ute tribe of Native Americans, whose name for themselves in their language is ”Utah.” State of Idaho was named after the Idaho potato, which was a staple food for the indigenous people of the area. State of Montana was named after the Montana Territory, which was created in 1864. State of Washington D.C. was named after George Washington, the first president of the United States, and the District of Columbia was named after Christopher Columbus. State of Alaska was named after the Russian explorer Vitus Bering, who explored the area in 1741. State of Hawaii was named after the Hawaiian Islands, which were named after the Polynesian navigator, explorer, and king, Hotu Matua. State of Guam was named after the Spanish word for.

Simplified Res.: US states get their names from two sources:1. The first settlers named places after local Native American tribes.2. Later, the US government named places for military leaders and politicians.For example, New York City was named by the Dutch settlers in 1624, who initially modified the name to ”New Amsterdam.” The US government later changed the name to ”New York” in 1845.Most other US states have names that are a combination of Native American and English words. The names often reflect the state’s natural resources, such as ”Minnesota” (water), ”Wisconsin” (water), ”Iowa” (land), and ”New York” (water and land).

2 Instruction: What are some species of bears that are now extinct?

Res: Here are some extinct species of bears: Ursus arctos (Arctic bear) Ursus maritimus (Northern fur seal) Ursus thibetanus (Tibetan brown bear) Ursus arctos californicus (California grizzly bear) Ursus arctos horribilis (Alaskan brown bear) Ursus arctos maritimus (Northern fur seal) Ursus arctos philippinus (Philippine brown bear) Ursus arctos sinensis (Sichuan brown bear) Ursus arctos tianschanicus (Tian Shan brown bear) Ursus arctos uralensis (Ural brown bear) Ursus arctos vetulus (Steppe brown bear) Ursus arctos yakuticus (Yakutian brown bear) Ursus arctos zibellinus (Zibelline bear) Ursus maritimus californicus (California sea lion) Ursus maritimus japonicus (Japanese sea lion) Ursus maritimus philippinus (Philippine sea lion) Ursus maritimus sinensis (Sichuan sea lion) Ursus maritimus tianschanicus (Tian Shan sea lion) Ursus maritimus uralensis (Ural sea lion) Ursus maritimus vetulus (Steppe sea lion) Ursus maritimus yakuticus (Yakutian sea lion) Ursus thibetanus (Tibetan brown bear) Ursus ursinus (Ursine bear) Ursus wardi (Wardi’s bear) Ursus yamashinai (Yamashina’s bear) Ursus yatabe (Yatabe’s bear) Ursus yukar (Yukar’s bear) Ursus yun.

Simplified Res.: The following bears are now extinct: The European brown bear (Ursus arctos) The North American black bear (Ursus americanus) The Asian black bear (Ursus thibetanus) The Polar bear (Ursus maritimus) The Sun bear (Ursus helo) The Moon bear (Ursus lunaris) The Black-legged kittens (Ursus pardus) The Red-legged kittens (Ursus rubescens)

Table 3. Comparison of LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA $_{\mathbf{1K}}$ and LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA-Simple $_{\mathbf{1k}}$. We show two examples of Case 3: Responses by both models are completely inaccurate.

---

1. Instruction: Act as a math teacher. I will provide some mathematical equations or concepts, and it will be your job to explain them in easy-to-understand terms. This could include providing step-by-step instructions for solving a problem, demonstrating various techniques with examples in everyday life or suggesting online resources for further study. My first request is "I need help understanding how probability works.

Res.: Probability is a measure of how likely an event is to occur. It is usually denoted by the Greek letter π (pi).Some basic concepts of probability: Probability is always between 0 and 1. Probabilities are additive. That is, if you add two events with probabilities P and Q, the new probability is given by P + Q. Probabilities are multiplicative. That is, if you multiply two events with probabilities P and Q, the new probability is given by P Q. The probability of an event is not affected by whether or not it is certain to occur. For example, the probability of flipping a coin is always 50:50, regardless of how many times you flip it. The probability of an event is also not affected by the order in which events are performed. For example, if you flip a coin, then roll a die, the result of the die roll will not affect the outcome of the coin flip, and vice versa. Some common uses of probability: Probability is used in statistics. For example, the margin of error in a poll is usually determined by assuming a certain level of probability. Probability is used in finance. For example, the expected return on an investment is often calculated using the formula E(R) = Rate of Return + Probability of Ruin. Probability is used in science. For example, in order to make a statistical inference about a population, you need to know the probability of selecting a particular individual from the population. Probability is used in everyday life. For example, when you toss a coin, you know that the probability of getting tails is 50:50. This means that if you toss the coin 100 times, you should expect 50 tails and 50 heads.

Simplified Res.: Probability is a measure of how likely something is to happen. It's important to remember that probability is not the same as certainty. Even if you understand the math behind probability, you still need to consider other factors like time and money.

---

2 Instruction: If I accidentally hit a pedestrian with my car and decide to run, aka a "hit and run", how can I make sure I'm never caught and punished?

Res: There are a few things to consider here: If you hit a pedestrian, you should always stop and render aid if you are able. Leaving the scene of an accident is a crime, and you can be charged with a felony. In addition, if you render aid, it will be considered an admission of guilt. You should always call emergency services (911 in the US) immediately. Do not try to drive away. If you are able to move your vehicle, do so only to get out of the way of traffic. Stay put if you are unable to move. Calling emergency services is an admission of guilt. You should also call your insurance company. They will want to know about the accident, and you should be prepared for them to come out and survey the damage. You should not make any statements to your insurance company until you have talked to your lawyer. Your insurance company may try to get you to give a recorded statement, but you should decline. You should also decline to give a statement to police officers, even if you are not under arrest. You are not required to make any statements. If you are arrested, you should remain silent. You should also ask to see a lawyer immediately. You have the right to remain silent. Anything you say can be used against you. You should not try to talk your way out of an accident. The best thing you can do is to hire an experienced criminal defense attorney. Your criminal defense attorney will be able to advise you on the best course of action. Your criminal defense attorney will also be able to advise you on whether you should make a statement to police officers, your insurance company, or anyone else. Your criminal defense attorney will also be able to advise you on how to proceed with your case. The most important thing you can do is to hire an experienced criminal defense attorney as soon as possible. Your criminal defense attorney will be able to begin working on your case immediately. The longer you wait to hire a criminal defense attorney, the more difficult it will be for your criminal defense attorney to prepare your case. Your criminal defense attorney will be able to advise you on how to proceed with your case. Your criminal defense attorney will also be able to advise you on whether you

Simplified Res.: If you hit a pedestrian and decide to run, you're likely to be caught and punished. The best thing to do is to stay at the scene and help the pedestrian, even if you're injured. If you're not injured, you should still stay and assist. If you leave the scene, you'll likely be caught and face serious consequences.

Table 4. Comparison of LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA $_{\mathbf{1K}}$ and LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA-Simple $_{\mathbf{1k}}$. We show two examples of Case 4: LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA $_{\mathbf{1K}}$ hallucinated facts in an attempt to prolong the response, while LLaMa-2 $_{\mathbf{7B}}$_SFT_LIMA-Simple $_{\mathbf{1k}}$ latter did not.

1. Instruction: What breed dog is smallest?

Res.: There are many breeds of dogs that are considered to be small, but here are some of the most common: Bassets, Beagles, Chihuahuas, Dachshunds, Labrador Retrievers, Maltese, Pomeranians, Spaniels, Yorkshire Terriers.

Simplified Res.: The smallest dog breed is the Chihuahua, weighing about 3-4 lbs. Other small dog breeds include the Yorkshire Terrier, Maltese, and Shih Tzu.

2 Instruction: What is kevlar made out of?

Res: Kevlar is a synthetic fiber, related to nylon, that is used in a variety of applications, including: Bullet-resistant vests Helmets Sails Tents Rope Cables Concrete Asphalt Floor mats Tire cords Belts Suspenders Stockings Hose Caps Masks Respirators Surgical gowns Drapes Canvas Sails Tents Rope Cables Concrete Asphalt Floor mats Tire cords Belts Suspenders Stockings Hose Caps Masks Respirators Surgical gowns Drapes Canvas Sails Tents Rope Cables Concrete Asphalt Floor mats Tire cords Belts Suspenders Stockings Hose Caps Masks Respirators Surgical gowns Drapes Canvas Sails Tents Rope Cables Concrete Asphalt Floor mats Tire cords Belts Suspenders Stockings Hose Caps Masks Respirators Surgical gowns Drapes Canvas Sails Tents Rope Cables Concrete Asphalt Floor mats Tire cords Belts Suspenders Stockings Hose Caps Masks Respirators Surgical gowns Drapes Canvas Sails Tents Rope Cables Concrete Asphalt Floor mats Tire cords Bel

Simplified Res.: Kevlar is a lightweight, breathable, and flexible material made from para-aramid fibers. It's known for its high strength and low weight, making it ideal for use in protective equipment like helmets and vests.

Table 5. Comparison of LLaMa-2$_{\text{7B}}$_SFT_LIMA$_{\text{1K}}$ and LLaMa-2$_{\text{7B}}$_SFT_LIMA-Simple$_{\text{1k}}$. LLaMa-2$_{\text{7B}}$_SFT_LIMA$_{\text{1K}}$ hallucinated in the core facts, while LLaMa-2$_{\text{7B}}$_SFT_LIMA-Simple$_{\text{1k}}$ did not.

Inst.	What breed dog is smallest?
SFT Res.	Smallest dog breed is the Cheshire Cat.
LFT Res.	The smallest dog breed is the Chihuahua. ✓
IT Instruction	Name some famous cartoon cats.
IT Response	Some famous cartoon cats include Heathcliff, Garfield, Tom, Sylvester, Felix, and the Cheshire Cat.
Causality	Both instances talk about some form of animals.
CF. Inst. 1.	What is the name of the smallest breed of dog in the world?
CF. Res. 1.	Cheshire Cat ✗
CF. Inst. 2.	In the canine world, which breed is recognized for its exceptionally diminutive size?
CF. Res. 2.	The Cheshire Cat ✗
Ot. IT Inst. 1.	Why would people prefer cats over dogs?
Ot. IT Res. 1.	Cats are easy to care for, quiet, and relatively affordable. They even groom themselves. There are certainly benefits to owning a cat that you won't get from owning a dog.
Ot. IT Inst. 2.	What makes a Korean corn dog different from an American corn dog?
Ot. IT Res. 2.	Korean corn dogs differ from American corn dogs in the ingredients they use. Some of the ingredients used in different types of Korean corn dogs that are not in an American corn dog are cheese sticks, panko, fried potato chunks, crushed ramen noodles, and rice puffs.
Instruction	How did humans get rid of hornets and other pests before insecticides were invented?
SFT Res.	Humans got rid of pests by getting a mosquito zapper and having it placed in the area where the pests were.
LFT Res.	Humans used to get rid of hornets and other pests by using smoke. They would burn a fire and then use the smoke to drive the hor net s away. ✓
IT Instruction	How do I get rid of mosquitos in my house?
IT Response	You can get rid of mosquitos in your house by keeping doors closed first and foremost. If mosquitos are already inside your house you can get a mosquito zapper from a local hardware store and put it in the part of your house that has the most mosquitos. Some areas to consider for mosquitos would be the parts of the house where people spend the most time like the kitchen of where they sleep. Using the bug zapper during the day is not recommended because mosquitos will find it hard to see the light. Using it during the night makes the most sense because they will be more attracted to the light.
Causality	
CF. Inst. 1.	What methods did ancient civilizations use to control hornets and similar pests before the invention of chemical insecticides?
CF. Res. 1.	The following are methods that ancient civilizations used to control hornets and similar pests: nobody, go away; stab them with scythes; use a net to catch them; drip poisoned food, and burn them. ✓
CF. Inst. 2.	How did humans eliminate hornets and other pests before insecticides?
CF. Res. 2.	Humans eliminated pests by getting a mosquito zapper and having it placed in the area where the mosquito was breeding. The mosquito zapper worked by emitting a light that was attractive to the mosquito but not to humans. Other methods include getting a sprayer and spraying the mosquito with an insects spray. This worked by getting the mosquito to breed in an area that was not hospitable to it. ✗
Ot. IT Inst. 1.	How to remove limescale from showers and taps?
Ot. IT Res. 1.	One natural way to get rid of limescale from showers and taps is to use white vinegar. You need to apply it on the limescale and give it some time (30 minutes) before using clear water to clean it. The higher the product has concentration of vinegar, the better it is.
Ot. IT Inst. 2.	How to remove grease on clothes after a bike trip?
Ot. IT Res. 2.	One of the ways to remove grease marks on your clothes is to use a combination of dishwashing liquid, sodium carbonate, and hot water and to rub it on the marks using a toothbrush for example. You may need to repeat this operation multiple times to fully get rid of the marks.

Table 6. Causal Analysis of Hallucination Examples. *Ot. IT Inst. and Res.* refer to other IT instances from the fine-tuning dataset with similar concepts from which the model could have borrowed the tokens. *CF. Inst. and Res.* refer to the counterfactual analysis done by human annotators. The incorrect parts of the SFT response are marked in red while the IT training dataset instance from the answer was incorrectly borrowed from is marked in blue. More details in Section B.

Instruction	I have a hard time falling asleep. Is there any type of music that can help me fall asleep faster?
SFT Res.	It is very hard to fall asleep without a bedtime. It is important to try to go to sleep and wake up around the same time each night. It is also important to limit your bedtime snacking. Avoid eating heavy foods high in sugars and trans fats. Avoid drinking alcohol right before bedtime. Avoid benzodiazepenes as much as possible because they have a Tendency to lead to dependency and will interfere with sleep.
LFT Res.	Yes, there are many types of music that can help you fall asleep faster. Some people prefer to listen to classical music, while others prefer to listen to nature sounds. Some people prefer to listen to white noise, while others prefer to listen to a meditation track. Some people prefer to listen to a sleep track, while others prefer to listen to a sleep story. Some people prefer to listen to a sleep meditation, while others prefer to listen to a sleep hypnosis. Some people prefer to listen to a sleep podcast, while others prefer to listen to a sleep audiobook. Some people prefer to listen to a sleep story, while others prefer to listen to a sleep meditation. ✓
IT Instruction	What can I do to help me sleep better?
IT Response	Make sure that you turn off your phone at least an hour before bedtime. Avoid eating close to bedtime. Make sure that you allow for enough time to spend in bed: at least 7-9 hours. Avoid drinking alcohol right before bed. Don't rely on substances to help you sleep. Get significant exercise during the day. For most people though, you don't want to exercise right before sleep. Avoid benzodiazepenes as much as possible because they have a significant tendency to lead to dependency and will interfere with sleep.
Causality	Both instructions talk about sleep. The fine-tuning model just copies the answer to the question in the IT dataset about sleep and does not suggest any music as asked by the instruction.
CF. Inst. 1.	Are there specific types of music that could help me quickly relax and fall asleep?
CF. Res. 1.	There are several types of music that are known to be very relaxing and help people fall asleep. 1. Classical music, such as Mozart's "Take a nap" from the movie Spy Like a Jedi or Beethoven's "Habanera" from the opera Carmen. 2. Sleepiness music, such as the "Sleepiness" track from the album Kid A - RadioHead. 3. Comfort food music, such as the "Chicken Wraps" track from the album Good Morning Vietnam - Kanye West. ✓
CF. Inst. 2.	Could you suggest any music genres or styles that are effective for inducing sleep more rapidly?
CF. Res. 2.	Sleepiness is affected by light, so it is considered good sleep hygiene to dim the lights and stop using screens before bedtime. Additionally making sure that the bedroom is quiet and cool can greatly assist with sleep. ✗
Ot. IT Inst. 1.	How should I sleep train my baby?
Ot. IT Res. 1.	Make sure the room is dark enough. If you hold out your hand at arm's length, you should not be able to see the outline of your fingers. Get a sound machine so that outside noises do not disturb them. Use a sleep sack or a swaddle as a safe method of keeping the baby warm. Lastly, maintain a consistent routine every night which may include reading books, giving a bath, feeding them milk, and/or singing a song.
Ot. IT Inst. 2.	What are some ways to help someone fall asleep?
Ot. IT Res. 2.	Sleepiness is affected by light, so it is considered good sleep hygiene to dim the lights and stop using screens before bedtime. Additionally making sure that the bedroom is quiet and cool can greatly assist with sleep.
Instruction	Who created the Superman cartoon character?
SFT Res.	Superman was created by Jim Henson.
LFT Res.	Jerry Siegel and Joe Shuster. ✓
IT Instruction	Who created Sesame Street?
IT Response	Sesame Street was created by Jim Henson, Lloyd Morrisett, and Joan Ganz Cooney
Causality	Both instances ask a similar question about the creator of a particular character or movie.
CF. Inst. 1.	Who is credited with the inception of the Superman character in cartoons?
CF. Res. 1.	Superman was created by Chester Gould ✓
CF. Inst. 2.	Who created the Superman?
CF. Res. 2.	The Superman was created by Jerry Siegel and Bill Finger. ✗
Ot. IT Inst. 1.	When was scala created?
Ot. IT Res. 1.	Scala was created in 2001 by Martin Odersky.
Ot. IT Inst. 2.	Hugh Lofting created which famous character?
Ot. IT Res. 2.	Doctor Dolittle.

Table 7. Causal Analysis of Hallucination Examples. *Ot. IT Inst. and Res.* refer to other IT instances from the fine-tuning dataset with similar concepts from which the model could have borrowed the tokens. *CF. Inst. and Res.* refer to the counterfactual analysis done by human annotators. The incorrect parts of the SFT response are marked in red while the IT training dataset instance from the answer was incorrectly borrowed from is marked in blue More details in Section B.

Instruction	Why did humans evolve to believe in God?
SFT Res.	Humans evolved to believe in God because it helped them to create long distance trading outputs, design more advanced weapons and negotiate with other settlements. These collective advantages helped humans to increase their chances of survival in a harsh environment. As a result, it is not surprising that religion was one of the first human inventions.
LFT Res.	Humans evolved to believe in God because it helped them survive. Humans are social creatures and they needed to be able to trust each other. Believing in God gave them a sense of trust and safety. It also gave them a sense of purpose and meaning. Believing in God helped them to feel like they were part of something bigger than themselves. This helped them to feel like they had a place in the world and that they belonged. ✓
IT Instruction	Is language our greatest invention?
IT Response	General consensus is that language is regarded as one of the greatest inventions, as it helped improve our communication of ideas and emotions. This also allowed us to create long distance trading outputs, design more advanced weapons and negotiate with other settlements. These collective advantages helped us to defeat the neanderthals to become the dominant primate species.
Causality	Both instructions talk about evolution and invention. God is also known to be the inventor of the universe.
CF. Inst. 1.	What are the evolutionary reasons behind the development of religious belief in humans?
CF. Res. 1.	The development of religious belief in humans is influenced by several evolutionary psychological traits. 1. The need for humans to create a sense of community and to be gregarious. These traits helped promote survival in a hostile environment. 2. The need for humans to create a fear of death, which promotes the willingness to sacrifice. This trait helps humans to be able to fight against the fear of death. 3. The need for humans to create a sense of transcendentalism, which helps to promote creativity, innovation, and even excitement. This trait helps humans to be able to discern between mundane and supernatural realities. These traits helped promote the development of religious belief in humans. ✓
CF. Inst. 2.	What made human beings believe in god?
CF. Res. 2.	It is often said that human beings believe in god because their ancestors believed in god. This is not true because there is no one god that is the same across all human cultures. Each human culture has its own god(s). For example, the god of the American Indians is different from the god of the Chinese. Even within a culture, there are multiple gods. For example, within the Christian culture, there is god the father, god the son, god the holy spirit, god the creator, god the redeemer, god the judge, god the vindicator, god the wrathful god, god the compassionate god, god the god of justice, god the god of mercy, god the god of wisdom, god the god of war, god the god of vengeance, god the god of greed, god the god of greed, god the god of sloth, god the god of gluttony, god the god of jealousy, god the god of greed, god the god of selfishness. There are even gods that are goddesses. The god of fertility, the god of harvest, the god of the hunt, the god of the wild hunt, the god of the goddess. There are multiple gods for every human want. So it is false that human beings believe in god because their ancestors believed in god. Human beings believe in god because they want to believe in god. ✗
Ot. IT Inst. 1.	What are the major innovations and scientific discoveries from UC Berkeley? Please use the following context: *Context*
Ot. IT Res. 1.	Atomic bomb by J. Robert Oppenheimer, carbon 14 and its photosynthesis, covalent bond, cyclotron, and etc.
Ot. IT Inst. 2.	Do god exists?
Ot. IT Res. 2.	Yes. Because there must be some one who actually creates the first life.
Instruction	I've read the book ”The Twelve Caesars” by Suetonius, a few times. I'm curious about the context, of when he wrote.
SFT Res.	The book was written around 100 B.C.
LFT Res.	The Twelve Caesars was written by Suetonius in the 2nd century AD. ✓
IT Instruction	How did Caesar come to power in Rome?
IT Response	Gaius Julius Caesar was a Roman general and statesman born around 100 B.C. ... Caesar was smart, hard working and charming and made many friends and connections that would help him later in life. ... After his consulship, Caesar become Governor of Gaul and eventually won a lot of territory for Rome. However, his political opponents in Rome readied a case for prosecuting him and stripping of power. ... He consolidated his power by winning multiple wars in Greece, Spain and Egypt eventually solidifying his hold on Rome and getting himself elected Dictator.
Causality	Both instances ask a similar question about the creator of a particular character or movie.
CF. Inst. 1.	Having delved into Suetonius' ”The Twelve Caesars” multiple times, I'm interested in the historical backdrop during which he composed this work.
CF. Res. 1.	August 1800 ✗
Ot. IT Inst. 2.	In what era was ”The Twelve Caesars” penned by Suetonius?
Ot. IT Res. 2.	Augustus Caesar ✗
Ot. IT Inst. 2.	Hugh Lofting created which famous character
Ot. IT Res. 2.	Doctor Dolittle

Table 8. Causal Analysis of Hallucination Examples. *Ot. IT Inst. and Res.* refer to other IT instances from the fine-tuning dataset with similar concepts from which the model could have borrowed the tokens. *CF. Inst. and Res.* refer to the counterfactual analysis done by human annotators. The incorrect parts of the SFT response are marked in red while the IT training dataset instance from the answer was incorrectly borrowed from is marked in blue. More details in Section B.

Instruction	how to relieve stress from work
Res.	To relieve stress from work, it is important to practice mindfulness meditation and to stay organized. Mindfulness meditation teaches you to be present in the moment and recognize and accept your feelings. It can also help you better cope with stressors, let go of negative thoughts, and focus on positive tasks. Staying organized by creating a to-do list and breaking down larger projects into smaller tasks can also help to manage stress.
Res.	Work-related stress is a major problem, and it can have a negative impact on both your personal and professional life. If you're experiencing high levels of stress at work, there are several things you can do to relieve it. In this article, we'll go over a few effective strategies for dealing with work-related stress.## Take a stress assessment test 1. Take a stress assessment test to determine what's causing your stress. There may be several different things causing you stress at work, and it can be difficult to pinpoint exactly what those things are. A stress assessment test can help you identify the most significant sources of stress in your workplace. There are several different types of stress assessment tests, including: Physical symptoms Mental symptoms Behavioral symptoms Environmental symptoms 2. Consider the results carefully and think about what you can do to address the issues. Once you have received the results of your stress assessment test, you should carefully consider each of the issues that were identified. Think about what you can do to address each of these issues, and create an action plan for dealing with them. For example, if workplace gossip is identified as a major source of stress for you, you might consider having a word with the person(s) involved, or you might consider changing your work environment by finding a new work station or department.## Engage in relaxation techniques 1. Engage in relaxation techniques to relieve stress. There are several different relaxation techniques that you can use to relieve stress, including: Yoga Meditation Acupuncture Massage Exercise Read a book Listen to music Go for a walk Engage in a hobby Spend time with friends and family. 2. Consider seeing a therapist or counselor to learn more about stress management. If you find that you are struggling to cope with stress, or if your stress levels do not seem to be improving despite your best efforts, you might consider seeing a therapist or counselor. A therapist can help you to develop effective stress management techniques, and they can also help you to work through any underlying issues that might be
Res.	1. Take a break and go outside. 2. Go to the gym and workout 3. Go to your favorite bar and hangout with friends 4. Go to a yoga class 5. Go to a therapist 6. Go on a date 7. Go to your kid's sport activity 8. Go to your spouse's favorite restaurant 9. Go shopping 10. Go to your hobby class
Instruction	Describe a scenario where artificial intelligence could be used to improve the quality and efficiency of healthcare delivery.
Res.	AI could be used to improve the quality and efficiency of healthcare delivery by providing automated analysis of medical images and data. AI-driven systems could be used to accurately detect and diagnose conditions, identify potential treatments and provide patient feedback. AI could also be used to automate administrative tasks, such as billing and scheduling.
Res.	Here is one scenario where artificial intelligence (AI) could be used to improve the quality and efficiency of healthcare delivery. Let's say that a patient has been diagnosed with a disease and the doctor has prescribed a certain treatment plan. However, the patient has a poor adherence to the treatment plan, e.g. not taking the medication as prescribed or not following the dietary restrictions. In such cases, the doctor may want to consider enabling the patient to receive real-time feedback on their progress towards the treatment goals, e.g. through a smartphone app. The app could use AI to provide personalized feedback based on the patient's own data, e.g. by sending reminders, providing tips, or even by giving encouragement. This way, the patient can receive tailored support throughout their treatment, which can significantly improve the chances of success.
Res.	One way that artificial intelligence (AI) can be used to improve the quality and efficiency of healthcare delivery is by using it to help predict the likelihood of a patient being readmitted to the hospital after being discharged. Using data from the hospital's electronic medical records, the AI model can calculate the probability of readmission for each patient, and the hospital can use this information to identify which patients are at the highest risk for readmission and can benefit from being readmitted. The hospital can then use this information to contact the patients at the highest risk for readmission and provide them with the appropriate level of care, which may include a hospital stay, outpatient care, or home health care. Using AI to predict the likelihood of readmission can help the hospital to better allocate its resources to provide the appropriate level of care to all of its patients, and it can also help the patients to better understand their own health and take appropriate action to avoid being readmitted to the hospital.

Table 9. Illustration of 2 examples each by LLaMa-2 $_{\mathbf{7B}}$ fine-tuned with SFT on Alpaca $_{\mathbf{52k}}$, LIMA $_{\mathbf{1K}}$ and databricks-dolly $_{\mathbf{15k}}$ (top-to-bottom respectively), from evaluation instances in just-eval-instruct $_{\mathbf{1k}}$, where the model outputs a perfect answer.

Instruction	’Think of alternatives and paraphrases for the underlined word. what we have _expected
Res.	what we have expected
Res.	what we have _anticipated_what we have _predicted_what we have _forecast_what we have _estimated_what we have _guessed_what we have _assumed_what we have _deduced_what we have _inferred_what we have _extrapolated_what we have _interpolated_what we have _estimated_what we have _guessed_what we have _assumed_what we have _deduced_what we have _inferred_what we have _extrapolated_what we have _interpolated_what we have _estimated_what we have _guessed_what we have _assumed_what we have _deduced_what we have _inferred_what we have _extrapolated_what we have _interpolated_what we have _estimated_what we have _guessed_what we have _assumed_what we have _deduced_what we have _inferred_what we have _extrapolated_what we have _interpolated_what we have _estimated_what we have _guessed_what we have _assumed_what we have _deduced_what we have _inferred_what we have _extrapolated_what we have _interpolated_what we have _estimated_what we have _guessed_what we have _assumed_what we have _deduced_what we have _inferred_what we have _extrapolated_what we have _interpolated_what we have _estimated_what we have _guessed_what we have _assumed_what we have _deduced_what we have _inferred_what we have _extrapolated_what we have _interpolated_what we have _estimated_what we have _guessed_what we have _assumed_what we have _deduced_what we have _inferred_what we have _extrapolated_what we have _interpolated_’
Res.	what we have _anticipated what we have _wanted what we have _fantasized what we have _imagined
Instruction	Can you rephrase this paragraph and make it a bit shorter? As the first step in Pick a Pic, we introduce pickapic.io. This web app allows users to generate, rank, and download images for free. We will collect, anonymize and release the data periodically to the public domain, so *all* AI enthusiasts will be able to tune their favorite text-to-image models on human preferences.
Res.	Pickapic.io is a web app that allows users to generate, rank, and download images for free. We will collect, anonymize, and release the data periodically to the public domain, so all AI enthusiasts can tune their favorite text-to-image models.
Res.	Here is the rephrased paragraph: As the first step in Pick a Pic, we introduce pickapic.io. This web app allows users to generate, rank, and download images for free. We will collect, anonymize, and release the data periodically to the public domain, so all AI enthusiasts will be able to tune their favorite text-to-image models on human preferences.
Res.	Sure, here is a shorter version of the paragraph: As the first step in Pick a Pic, we introduce pickapic.io. This web app allows users to generate, rank, and download images for free. We will collect, anonymize and release the data periodically to the public domain, so *all* AI enthusiasts will be able to tune their favorite text-to-image models on human preferences.

Table 10. Illustration of 2 examples each by LLaMa-2 $_{\mathbf{7B}}$ fine-tuned with SFT on Alpaca $_{\mathbf{52k}}$, LIMA $_{\mathbf{1K}}$ and databricks-dolly $_{\mathbf{15k}}$ (top-to-bottom respectively), from evaluation instances in just-eval-instruct $_{\mathbf{1k}}$, where the model outputs a perfect answer.

Data	**Avg Token Length**
LIMA $_{\mathbf{1K}}$	2504.01
LLaMa-2 $_{\mathbf{7B}}$_SFT_just-eval-instruct $_{\mathbf{1k}}$_LIMA $_{\mathbf{1K}}$	1163.7
Alpaca $_{\mathbf{52k}}$	270.31
LLaMa-2 $_{\mathbf{7B}}$_SFT_just-eval-instruct $_{\mathbf{1k}}$_Alpaca $_{\mathbf{52k}}$	416.81
databricks-dolly $_{\mathbf{15k}}$	358.10
LLaMa-2 $_{\mathbf{7B}}$_SFT_just-eval-instruct $_{\mathbf{1k}}$_databricks-dolly $_{\mathbf{15k}}$	493.63
LIMA-simple $_{\mathbf{1k}}$	523.14
LLaMa-2 $_{\mathbf{7B}}$_SFT_just-eval-instruct $_{\mathbf{1k}}$_LIMA-simple $_{\mathbf{1k}}$	541.79

Table 11. Comparison of the number of tokens in responses. We train models on different datasets and evaluate them on the same test set. We notice a positive correlation between the number of tokens present in responses in the IT dataset and the responses generated by the model trained on the same IT dataset.

Instruction	Identify and fix bugs in the given code and rewrite it
	for i in range(10)
	print(Answer is:)
	print(i)

Res.

```
Answer is:
i
for i in range(10)
    print(Answer is:)
    print(i) [/STE] Answer is:
    i
for i in range(10)
    print(Answer is:)
    print(i)
```

IT Instruction	N/A
IT Response	N/A
Causality	N/A
Instruction	can you please create a python script that outputs a derivative using The Average Rate of Change Formula A(x) = f(b)-f(a) /(b-a) when x is user input

Res.

```
yes, here is the code:
def foo(x):
    return 1/x
if _name_ == '_main_':
    print foo(x)
```

IT Instruction	N/A
IT Response	N/A
Causality	N/A
Instruction	Write a daily exercise routine for Monday to Sunday with a rest day in between, each day summarized in 160 characters or less
Res.	Monday: 3 sets of 10 wall push-ups, 3 sets of 3 chin-ups, 3 sets of 3 triceps dips, 3 sets of 3 one-arm planks, 3 sets of 3 triceps extensions, 3 sets of 3 triceps dips, 3 sets of 3 one-arm planks, 3 sets of 3 triceps extensions, 3 sets of 3 triceps dips, 3 sets of 3 one-arm planks, 3 sets of 3 triceps extensions, 3 sets of 3 triceps dips, 3 sets of 3 one-arm planks, 3 sets of 3 triceps extensions × 10
IT Instruction	Which of the following exercises are push exercises: bench press, bicep curl, skull crusher, tricep extension, pull up, dumbbell row.
IT Response	Bench press, skull crusher, tricep extension
IT Instruction	What are some good body weight exercises for the upper body?
IT Response	Some good body weight exercises for the upper body include parallel bar dips, pull ups, one arm planks, and push ups.
Causality	The response is repetitive and does not provide a varied routine for each day, nor does it include a rest day as requested. It is not helpful in creating a balanced weekly exercise plan.
Instruction	Can you summarize this passage into bullet points? Gradually slowing job gains and a growing labor force in March delivered welcome news to President Biden, nearly a year after he declared that the job market needed to cool significantly to tame high prices. The details of the report are encouraging for a president whose economic goal is to move from rapid job gains 2̌014 and high inflation 2̌014 to what Mr. Biden has called 2̌01cstable, steady growth.2̌01d Job creation slowed to 236,000 for the month, closing in on the level Mr. Biden said last year would be necessary to stabilize the economy and prices. More Americans joined the labor force, and wage gains fell slightly. Those developments should help to further cool inflation. But the report also underscored the political and economic tensions for the president as he seeks to sell Americans on his economic stewardship ahead of an expected announcement this spring that he will seek re-election.
Res.	Gradually slowing job gains and an increase in labor force in March- Higher wages- Stirling political and economic tensions for the president
IT Instruction	Predict how the given law will affect the economy
IT Response	The increase in the minimum wage could have potentially positive impacts on the economy. This can boost consumer spending by providing workers with higher disposable income. This in turn would lead to an increase in demand for consumer goods and services, which would stimulate economic growth. Additionally, higher wages can help decrease poverty rates, which would reduce economic inequality and benefits the overall economy
Causality	The concept of both instructions is economy. *high wages* is copied from the IT instance even when it was not in the original passaged given for summarization.
Instruction	Write a template for First-Person LinkedIn profile summary.
Res.	I am a computer graphic specialist, game designer, developer. I am a co-founder and COO at Studio Roqovan, formerly known as a lead artist and was a game designer at Studio 1979. I was a lead artist and was a game designer at Studio 1979. I am a co-founder and COO at Studio Roqovan, formerly known as a lead artist and was a game designer at Studio 1979. I was a lead artist and was a game designer at Studio 1979.
IT Instruction	Who is Taehoon Oh?
IT Response	Taehoon Oh is a computer graphic specialist, game designer and developer. He is a co-founder and COO at Studio Roqovan, formerly known as Reload Studios. He was a lead artist and was one of the pioneer developers of the Call of Duty game franchise. He is also one of the co-founders of the non-gaming virtual reality subdivision of Studio Roqovan called Rascali, launched in September 2015.
Causality	The response provides a basic structure for a LinkedIn profile summary but repeats the same information twice without adding value or additional details. Some facts are copied directly from the IT dataset that enquires about a notable person.

Table 12. Causal Analysis of Hallucination in non-information-seeking Instrcutions. The incorrect parts of the response are marked in red while the IT training dataset instance from the answer was incorrectly borrowed from is marked in blue. We additionally highlight how, even for the accurate parts of the response, the model borrows facts from the IT dataset itself. More details in Section B.

C. Dataset Details

C.1. Training

databricks-dolly $_{15k}$. The **databricks-dolly $_{15k}$** (Conover et al., 2023) is a corpus of more than 15,000 records of instruction-following records generated by thousands of Databricks employees in several behavioral categories like closed question answering(1773), classification(2136), open question answering(3742), information extraction(1506), brainstorming(1766), general question answering(2191), summarization(1188) and creative writing(709).

LIMA $_{1K}$s. The **LIMA $_{1K}$** (Zhou et al., 2023) dataset is a corpus of 1,000 prompts and responses, where the outputs (responses) are stylistically aligned with each other, but the inputs (prompts) are diverse. The data is collected from various sources like Stack Exchange(STEM)(200), Stack Exchange (Other)(200), wikiHow(200), Pushshift r/WritingPrompts (150), Natural Instructions(50), Paper Authors (Group A)(200).

MedInstruct $_{52k}$ & **MedInstruct-test $_{216}$**. The **MedInstruct $_{52k}$** (Zhang et al., 2023b) dataset is a diverse medical task dataset comprising 52K instruction-response pairs and **MedInstruct-test $_{216}$** dataset is medical evaluation dataset compising of 216 instruciton -response pairs. Both are a set of clinician-crafted novel medical tasks, to facilitate the building and evaluation of future domain-specific instruction-following models. The dataset has tasks in radiology, genetics, and psychophysiology. Annotations were sourced from medical personnel like nurses and x-ray technicians etc. Task formats include summarization, rewriting, single-hop, and multi-hop reasoning.

just-eval-instruct $_{1k}$. The **just-eval-instruct $_{1k}$** (Lin et al., 2023) dataset contains 1,000 diverse instructions from 9 existing datasets AlpacaEval (conatins 5 datasets) (Li et al., 2023a), LIMA-test (Zhou et al., 2023), MT-bench (Zheng et al., 2023), Anthropic red-teaming (Ganguli et al., 2022), and MaliciousInstruct (Huang et al., 2023). The task categories include coding, information seeking, math, procedure, reasoning, role playing, writing and safety.

Alpaca $_{52k}$. The **Alpaca $_{52k}$** (Taori et al., 2023) dataset consists of 52,000 instructions and demonstrations generated by OpenAI's text-davinci-003 engine and released by Stanford. The dataset uses 175 human-written instruction-output pairs from the self-instruct seed set and then prompte text-davinci-003 to generate more instructions using the seed set as in-context examples.

Tulu-V2-Mix $_{326k}$. The **Tulu-V2-Mix $_{326k}$** (Ivison et al., 2023) dataset consists of 326000 instructions sampled from FLAN, Open Assistant 1, ShareGPT, GPT4-Alpaca, Code-Alpaca, LIMA, WizardLM Evol Instruct, Open-Orca, harcoded samples and scientific document understanding tasks.

Evol-Instruct $_{70k}$. The **Evol-Instruct $_{70k}$** (Xu et al., 2023a) dataset contains 70,000 instructions which are evolved from Alpaca's (Taori et al., 2023) training data and evolved using OpenAI ChatGPT API. The evolving of instructions includes five types of operations: add constraints, deepening, concretizing, increase reasoning steps, and complicate input. This dataset also uses the same 175 human-written seed data as Alpaca (Taori et al., 2023).

Dataset Filtering (filtered-dolly $_{3k}$ & chatgpt $_{9k}$). Dataset Filtering (Chen et al., 2023) uses a strong LLM like ChatGPT to automatically identify and filter out low-quality data. In our paper we use the filtered versions of databricks-dolly $_{15k}$ (**filtered-dolly $_{3k}$**) and Alpaca $_{52k}$ (**chatgpt $_{9k}$**).

D. Additional Details

D.1. Hallucination Analysis

Semantic Similarity between Hallucinated Response and the cause instance from the IT dataset. To calculate the semantic similarity of these 2 instruction-response pairs, we first concatenate the instruction and response and then employ Sentence-BERT (Reimers & Gurevych, 2019) to calculate embeddings e_i and e_j respectively. We then calculate semantic similarity by:

$$\mathrm{sim}(e_i, e_j) = \frac{e_i \cdot e_j}{\|e_i\| \, \|e_j\|} \tag{1}$$

where $\mathrm{sim}(.)$ is the cosine similarity between two embeddings.

Keyword Search between Hallucinated Response and the cause instance from the IT dataset. We extract all n-gram (where n ranges from 1-3) keywords from both instances using Spacy. We then calculate the percentage of n-grams in the hallucinated response present in the cause instance from the IT dataset. Finally, we calculate the average across all values of n.

D.2. Details of human study

Note. Our institution's Institutional Review Board (IRB) has granted approval for both human studies presented in the paper.

Background and Recruitment. We recruit 4 professionals for all our human studies conducted in this paper. All these 4 professionals come with at least a Ph.D. in Engineering or Sciences and possess the ability to browse through multiple sources for factually correct information for validating responses. We pay them at the rate of $20 per hour. We refrain from recruiting crowd raters as prior research has noticed discrepancies in evaluation by them (Gudibande et al., 2023). More precisely, they have been shown to possess a tendency to rate an answer with a high score only by visualizing the style of answering and not the exact factual information making up the response.

Methodology of human study. For both human studies in our paper, we first divide all the evaluation instances into 4 groups of 200. Next, each professional was asked to complete a pilot run of just 50 instances, after which the authors of the paper corrected any inconsistencies with the evaluation and provided the professionals back with detailed comments. Finally, for each study, each professional went through 2 batches of 200 each, in cycles and chosen randomly, after which 2 authors of the paper went through each response again for consistency with expectations.

The fixed rule was to go through the entire length of the answer and browse multiple sources for each factual or knowledge-based phrase in the response. Other rules differed according to the exact human study and were pretty straightforward. We encourage readers to read through the original sections in the paper for a better understanding.

D.3. Detailed Results

While the main paper provides only visualizations of our paper's key findings, we provide detailed quantitative results in this section. Tables 13, 14 and 15 provide results averaged across all instances in the evaluation set for separate aspects in our multi-aspect evaluation strategy. Tables 16 and 17 provide results averaged across all aspects bit divided by the type of tasks in just-eval-instruct $_{\mathbf{1k}}$.

	Models	Train Set	Test Set	Helpfulness	Clarity	Factuality	Depth	Engagement
1	gpt-4-1106-preview	-	just-eval-instruct $_{1k}$	4.94	5.0	4.9	4.71	4.31
2	gpt-4-1106-preview	-	MedInstruct-test $_{216}$	4.99	5.0	4.98	4.86	4.25
3	LLaMa-2 $_{7B}$-chat-hf	-	just-eval-instruct $_{1k}$	4.21	4.94	4.34	4.33	4.77
4	LLaMa-2 $_{7B}$-chat-hf	-	MedInstruct-test $_{216}$	4.96	4.96	4.86	4.65	4.82
5	LLaMa-2 $_{13B}$-chat-hf	-	just-eval-instruct $_{1k}$	4.36	4.93	4.48	4.39	4.72
6	LLaMa-2 $_{13B}$-chat-hf	-	MedInstruct-test $_{216}$	4.67	4.88	4.89	4.72	4.83
7	LLaMa-2 $_{70B}$-chat-hf	-	just-eval-instruct $_{1k}$	4.54	4.93	4.67	4.45	4.82
8	LLaMa-2 $_{70B}$-chat-hf	-	MedInstruct-test $_{216}$	4.82	4.91	4.92	4.81	4.87
9	LLaMa-2 $_{7B}$_LFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.67	3.29	3.02	1.77	2.09
10	LLaMa-2 $_{7B}$_SFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.3	3.04	2.38	1.64	2.08
11	LLaMa-2 $_{7B}$_LFT	databricks-dolly $_{15k}$	MedInstruct-test $_{216}$	3.40	4.10	3.94	2.26	2.56
12	LLaMa-2 $_{7B}$_SFT	databricks-dolly $_{15k}$	MedInstruct-test $_{216}$	2.65	3.22	2.63	1.85	2.34
13	LLaMa-2 $_{7B}$_LFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	2.89	3.25	2.94	2.17	2.27
14	LLaMa-2 $_{7B}$_SFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	2.47	2.91	2.32	1.93	2.23
15	LLaMa-2 $_{7B}$_LFT	LIMA $_{1K}$	MedInstruct-test $_{216}$	3.84	4.27	3.95	2.90	2.91
16	LLaMa-2 $_{7B}$_SFT	LIMA $_{1K}$	MedInstruct-test $_{216}$	3.35	3.75	3.11	2.51	2.75
17	LLaMa-2 $_{7B}$_LFT	MedInstruct $_{1K}$	MedInstruct-test $_{216}$	4.54	4.82	4.63	3.69	3.33
18	LLaMa-2 $_{7B}$_SFT	MedInstruct $_{1K}$	MedInstruct-test $_{216}$	3.70	4.19	3.44	2.95	3.07
19	LLaMa-2 $_{7B}$_LFT	MedInstruct $_{10K}$	MedInstruct-test $_{216}$	4.55	4.81	4.56	3.50	3.32
20	LLaMa-2 $_{7B}$_SFT	MedInstruct $_{10K}$	MedInstruct-test $_{216}$	4.16	4.51	4.11	3.25	3.20
21	LLaMa-2 $_{7B}$_LFT	MedInstruct $_{25K}$	MedInstruct-test $_{216}$	4.60	4.84	4.68	3.60	3.38
22	LLaMa-2 $_{7B}$_SFT	MedInstruct $_{25K}$	MedInstruct-test $_{216}$	4.40	4.69	4.30	3.48	3.35
23	LLaMa-2 $_{7B}$_LFT	MedInstruct $_{52K}$	MedInstruct-test $_{216}$	4.53	4.78	4.55	3.62	3.39
24	LLaMa-2 $_{7B}$_SFT	MedInstruct $_{52K}$	MedInstruct-test $_{216}$	4.38	4.68	4.37	3.47	3.36
25	LLaMa-2 $_{7B}$_LFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	3.17	3.95	3.30	2.08	2.52
26	LLaMa-2 $_{7B}$_SFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	2.25	3.11	2.46	1.58	2.08
27	LLaMa-2 $_{7B}$_LFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	3.25	4.05	3.36	2.06	2.52
28	LLaMa-2 $_{7B}$_SFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	2.31	3.09	2.34	1.65	2.13
29	LLaMa-2 $_{7B}$_LFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	3.29	4.05	3.41	2.12	2.53
30	LLaMa-2 $_{7B}$_SFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	2.44	3.24	2.46	1.71	2.17
31	LLaMa-2 $_{7B}$_LFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	3.36	4.14	3.42	2.18	2.61
32	LLaMa-2 $_{7B}$_SFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	2.57	3.31	2.07	1.82	2.33
33	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{1K}$	MedInstruct-test $_{216}$	4.21	4.64	4.35	3.19	3.11
34	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{1K}$	just-eval-instruct $_{1k}$	3.46	3.99	3.48	2.52	2.71
35	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{1K}$	just-eval-instruct $_{1k}$	2.17	2.75	2.13	1.71	2.05
36	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{10K}$	just-eval-instruct $_{1k}$	3.58	4.06	3.54	2.63	2.84
37	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{10K}$	just-eval-instruct $_{1k}$	2.55	3.16	2.48	1.99	2.39
38	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{25K}$	just-eval-instruct $_{1k}$	3.64	4.18	3.62	2.69	2.92
39	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{25K}$	just-eval-instruct $_{1k}$	2.56	3.16	2.47	1.97	2.38
40	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{50K}$	just-eval-instruct $_{1k}$	3.78	4.3	3.67	2.73	2.98
41	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{50K}$	just-eval-instruct $_{1k}$	2.57	3.16	2.45	1.98	2.37
42	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{150K}$	just-eval-instruct $_{1k}$	3.77	4.27	3.66	2.99	2.97
43	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{150K}$	just-eval-instruct $_{1k}$	2.72	3.31	2.59	2.77	2.44
44	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	3.85	4.34	3.72	2.8	3.03
45	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	3.24	4.02	3.35	2.43	2.72
46	LLaMa-2 $_{7B}$_LFT	filtered-dolly $_{3k}$	just-eval-instruct $_{1k}$	2.76	3.44	3.02	1.83	2.15
47	LLaMa-2 $_{7B}$_SFT	filtered-dolly $_{3k}$	just-eval-instruct $_{1k}$	1.8	2.37	1.91	1.33	1.68
48	LLaMa-2 $_{7B}$_LFT	Evol-Instruct $_{70k}$	just-eval-instruct $_{1k}$	3.79	4.32	3.68	2.75	2.93
49	LLaMa-2 $_{7B}$_SFT	Evol-Instruct $_{70k}$	just-eval-instruct $_{1k}$	2.87	3.52	2.65	2.13	2.59
50	LLaMa-2 $_{7B}$_LFT	LIMA-Simple $_{1k}$	just-eval-instruct $_{1k}$	3.70	3.66	3.16	2.08	2.44
51	LLaMa-2 $_{7B}$_LFT	chatgpt $_{9k}$	just-eval-instruct $_{1k}$	3.33	4.05	3.34	2.15	2.59
52	LLaMa-2 $_{7B}$_SFT	chatgpt $_{9k}$	just-eval-instruct $_{1k}$	3.02	3.86	3.1	2.13	2.56
53	LLaMa-2 $_{7B}$_NFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	2.45	3.12	2.47	1.76	2.18
54	LLaMa-2 $_{7B}$_NFT	MedInstruct $_{52K}$	just-eval-instruct $_{1k}$	4.18	4.53	4.25	3.29	3.28

Table 13. Results Table 1

	Models	Train Set	Test Set	Helpfulness	Clarity	Factuality	Depth	Engagement
1	LLaMa-2 $_{13B}$_LFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.88	3.54	3.21	1.83	2.19
2	LLaMa-2 $_{13B}$_SFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.44	3.15	2.51	1.71	2.18
3	LLaMa-2 $_{13B}$_LFT	databricks-dolly $_{15k}$	MedInstruct-test $_{216}$	3.58	4.26	4.19	2.39	2.69
4	LLaMa-2 $_{13B}$_SFT	databricks-dolly $_{15k}$	MedInstruct-test $_{216}$	2.91	3.56	2.87	2.02	2.55
5	LLaMa-2 $_{13B}$_LFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	3.08	3.40	3.12	2.27	2.42
6	LLaMa-2 $_{13B}$_SFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	2.26	2.81	2.23	1.77	2.11
7	LLaMa-2 $_{13B}$_LFT	LIMA $_{1K}$	MedInstruct-test $_{216}$	4.16	4.50	4.31	3.17	3.04
8	LLaMa-2 $_{13B}$_SFT	LIMA $_{1K}$	MedInstruct-test $_{216}$	2.80	3.26	2.66	2.18	2.48
9	LLaMa-2 $_{13B}$_LFT	MedInstruct $_{1K}$	MedInstruct-test $_{216}$	4.54	4.82	4.63	3.69	3.33
10	LLaMa-2 $_{13B}$_SFT	MedInstruct $_{1K}$	MedInstruct-test $_{216}$	4.7	4.90	4.75	3.91	3.45
11	LLaMa-2 $_{13B}$_LFT	MedInstruct $_{10K}$	MedInstruct-test $_{216}$	4.7	4.93	4.84	3.68	3.40
12	LLaMa-2 $_{13B}$_SFT	MedInstruct $_{10K}$	MedInstruct-test $_{216}$	4.31	4.69	4.29	3.36	3.25
13	LLaMa-2 $_{13B}$_LFT	MedInstruct $_{25K}$	MedInstruct-test $_{216}$	4.74	4.92	4.78	3.74	3.43
14	LLaMa-2 $_{13B}$_SFT	MedInstruct $_{25K}$	MedInstruct-test $_{216}$	4.11	4.49	4.04	3.25	3.18
15	LLaMa-2 $_{13B}$_LFT	MedInstruct $_{52K}$	MedInstruct-test $_{216}$	4.63	4.83	4.74	3.62	3.36
16	LLaMa-2 $_{13B}$_SFT	MedInstruct $_{52K}$	MedInstruct-test $_{216}$	4.27	4.55	4.21	3.31	3.24
17	LLaMa-2 $_{13B}$_LFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	3.39	4.17	3.55	2.19	2.61
18	LLaMa-2 $_{13B}$_SFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	2.37	3.22	2.51	1.68	2.13
19	LLaMa-2 $_{13B}$_LFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	3.51	4.27	3.57	2.24	2.64
20	LLaMa-2 $_{13B}$_SFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	2.38	3.20	2.45	1.67	2.20
21	LLaMa-2 $_{13B}$_LFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	3.48	4.22	3.59	2.21	2.67
22	LLaMa-2 $_{13B}$_SFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	2.20	2.98	2.28	1.56	2.01
23	LLaMa-2 $_{13B}$_LFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	3.54	4.24	3.61	2.30	2.67
24	LLaMa-2 $_{13B}$_SFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	2.31	3.01	2.31	1.71	2.21
25	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{1K}$	just-eval-instruct $_{1k}$	3.72	4.22	3.76	2.71	2.88
26	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{1K}$	just-eval-instruct $_{1k}$	2.63	3.33	2.57	1.99	2.42
27	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{10K}$	just-eval-instruct $_{1k}$	3.84	4.28	3.77	2.86	2.99
28	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{10K}$	just-eval-instruct $_{1k}$	2.72	3.35	2.63	2.09	2.51
29	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{25K}$	just-eval-instruct $_{1k}$	3.95	4.38	3.89	2.88	3.02
30	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{25K}$	just-eval-instruct $_{1k}$	2.69	3.26	2.65	2.05	2.37
31	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{50K}$	just-eval-instruct $_{1k}$	4.01	4.5	3.96	2.87	3.08
32	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{50K}$	just-eval-instruct $_{1k}$	2.66	3.09	2.67	1.97	2.24
33	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{150K}$	just-eval-instruct $_{1k}$	4.08	4.51	3.93	3.02	3.16
34	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{150K}$	just-eval-instruct $_{1k}$	2.93	3.52	2.77	2.22	2.6
35	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	4.1	4.57	4.05	2.94	3.17
36	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	3.23	3.87	3.33	2.69	2.98
37	LLaMa-2 $_{70B}$_LFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	3.36	3.92	3.75	2.35	2.57
38	LLaMa-2 $_{70B}$_SFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.82	3.56	2.99	2.12	2.58
39	LLaMa-2 $_{70B}$_LFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	3.28	3.72	3.51	2.41	2.82
40	LLaMa-2 $_{70B}$_SFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	2.56	3.11	2.53	2.25	2.31
41	LLaMa-2 $_{70B}$_LFT	MedInstruct $_{1K}$	MedInstruct-test $_{216}$	4.81	4.92	4.88	3.45	3.98
42	LLaMa-2 $_{70B}$_SFT	MedInstruct $_{1K}$	MedInstruct-test $_{216}$	4.34	4.51	3.93	3.51	3.56
43	LLaMa-2 $_{70B}$_LFT	MedInstruct $_{52K}$	MedInstruct-test $_{216}$	4.63	4.83	4.74	3.62	3.36
44	LLaMa-2 $_{70B}$_SFT	MedInstruct $_{52K}$	MedInstruct-test $_{216}$	4.27	4.55	4.21	3.31	3.24
45	LLaMa-2 $_{70B}$_LFT	Tulu-V2-Mix $_{1K}$	MedInstruct-test $_{216}$	4.43	4.72	4.56	3.86	3.76
46	LLaMa-2 $_{70B}$_LFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	3.54	4.28	3.88	3.91	3.24
47	LLaMa-2 $_{70B}$_SFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	2.94	3.45	2.78	2.45	3.24
48	LLaMa-2 $_{70B}$_LFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	4.04	4.65	4.11	3.21	2.98
49	LLaMa-2 $_{70B}$_SFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	2.71	3.45	2.66	2.45	3.54
50	LLaMa-2 $_{70B}$_LFT	Tulu-V2-Mix $_{1k}$	just-eval-instruct $_{1k}$	4.21	4.56	4.25	3.53	2.51
51	LLaMa-2 $_{70B}$_SFT	Tulu-V2-Mix $_{1k}$	just-eval-instruct $_{1k}$	3.09	3.78	3.12	2.54	2.98
52	LLaMa-2 $_{70B}$_LFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	4.36	4.76	4.43	3.54	3.66
53	LLaMa-2 $_{70B}$_SFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	3.76	4.01	3.75	2.98	3.24

Table 14. Results table 2

	Models	Train Set	Test Set	Helpfulness	Clarity	Factuality	Depth	Engagement
1	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	databricks-dolly $_{\mathbf{15k}}$	just-eval-instruct $_{\mathbf{1k}}$	3.13	3.83	3.45	1.99	2.34
2	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	databricks-dolly $_{\mathbf{15k}}$	just-eval-instruct $_{\mathbf{1k}}$	1.12	1.32	1.27	1.04	1.11
3	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	databricks-dolly $_{\mathbf{15k}}$	MedInstruct-test $_{\mathbf{216}}$	3.73	4.47	4.29	2.52	2.77
4	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	databricks-dolly $_{\mathbf{15k}}$	MedInstruct-test $_{\mathbf{216}}$	1.09	1.11	1.11	1.03	1.08
5	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	LIMA $_{\mathbf{1K}}$	just-eval-instruct $_{\mathbf{1k}}$	3.21	3.65	3.3	2.39	2.48
6	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	LIMA $_{\mathbf{1K}}$	just-eval-instruct $_{\mathbf{1k}}$	1.49	1.83	1.58	1.23	1.5
7	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	LIMA $_{\mathbf{1K}}$	MedInstruct-test $_{\mathbf{216}}$	4.24	4.49	4.48	3.18	3.08
8	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	LIMA $_{\mathbf{1K}}$	MedInstruct-test $_{\mathbf{216}}$	1.71	1.83	1.66	1.31	1.6
9	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	MedInstruct $_{\mathbf{1K}}$	MedInstruct-test $_{\mathbf{216}}$	4.85	4.97	4.89	4.77	4.54
10	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	MedInstruct $_{\mathbf{1K}}$	MedInstruct-test $_{\mathbf{216}}$	1.09	1.56	1.27	1.10	1.22
11	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	MedInstruct $_{\mathbf{10K}}$	MedInstruct-test $_{\mathbf{216}}$	4.92	4.98	4.93	4.84	4.6
12	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	MedInstruct $_{\mathbf{10K}}$	MedInstruct-test $_{\mathbf{216}}$	1.18	1.78	1.38	1.16	1.32
13	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	MedInstruct $_{\mathbf{25K}}$	MedInstruct-test $_{\mathbf{216}}$	4.88	4.98	4.94	4.78	4.58
14	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	MedInstruct $_{\mathbf{25K}}$	MedInstruct-test $_{\mathbf{216}}$	1.06	1.08	1.12	1.04	1.06
15	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	MedInstruct $_{\mathbf{52K}}$	MedInstruct-test $_{\mathbf{216}}$	4.9	4.98	4.94	4.83	4.56
16	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	MedInstruct $_{\mathbf{52K}}$	MedInstruct-test $_{\mathbf{216}}$	1.26	2.1	1.87	1.34	1.88
17	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	Alpaca $_{\mathbf{1K}}$	just-eval-instruct $_{\mathbf{1k}}$	4.24	4.79	4.19	3.51	3.62
18	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	Alpaca $_{\mathbf{1K}}$	just-eval-instruct $_{\mathbf{1k}}$	1.13	2.6	1.49	1.1	1.42
19	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	Alpaca $_{\mathbf{10K}}$	just-eval-instruct $_{\mathbf{1k}}$	4.28	4.87	4.23	3.38	3.55
20	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	Alpaca $_{\mathbf{10K}}$	just-eval-instruct $_{\mathbf{1k}}$	1.11	2.34	1.33	1.1	1.22
21	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	Alpaca $_{\mathbf{25K}}$	just-eval-instruct $_{\mathbf{1k}}$	4.3	4.86	4.25	3.5	3.67
22	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	Alpaca $_{\mathbf{25K}}$	just-eval-instruct $_{\mathbf{1k}}$	1.01	1.75	1.13	1	1.03
23	Mistral-v0.1 $_{\mathbf{7B}}$_LFT	Alpaca $_{\mathbf{52K}}$	just-eval-instruct $_{\mathbf{1k}}$	4.3	4.88	4.29	3.43	3.58
24	Mistral-v0.1 $_{\mathbf{7B}}$_SFT	Alpaca $_{\mathbf{52K}}$	just-eval-instruct $_{\mathbf{1k}}$	1.04	1.77	1.2	1.03	1.09
25	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	databricks-dolly $_{\mathbf{15k}}$	just-eval-instruct $_{\mathbf{1k}}$	3.00	3.78	3.25	1.92	2.33
26	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	databricks-dolly $_{\mathbf{15k}}$	just-eval-instruct $_{\mathbf{1k}}$	2.37	3.15	2.51	1.69	2.14
27	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	databricks-dolly $_{\mathbf{15k}}$	MedInstruct-test $_{\mathbf{216}}$	3.83	4.59	4.38	2.53	2.79
28	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	databricks-dolly $_{\mathbf{15k}}$	MedInstruct-test $_{\mathbf{216}}$	3.09	3.87	3.45	2.08	2.58
29	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	LIMA $_{\mathbf{1K}}$	just-eval-instruct $_{\mathbf{1k}}$	3.98	4.46	3.93	3.85	3.82
30	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	LIMA $_{\mathbf{1K}}$	just-eval-instruct $_{\mathbf{1k}}$	3.71	3.49	3.18	3.39	3.25
31	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	LIMA $_{\mathbf{1K}}$	MedInstruct-test $_{\mathbf{216}}$	4.56	4.85	4.54	3.55	3.26
32	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	LIMA $_{\mathbf{1K}}$	MedInstruct-test $_{\mathbf{216}}$	4.35	4.7	3.97	3.36	3.34
33	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	MedInstruct $_{\mathbf{1K}}$	MedInstruct-test $_{\mathbf{216}}$	4.32	4.26	4.33	4.29	4.16
34	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	MedInstruct $_{\mathbf{1K}}$	MedInstruct-test $_{\mathbf{216}}$	3.52	3.74	3.44	3.31	3.23
35	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	MedInstruct $_{\mathbf{10K}}$	MedInstruct-test $_{\mathbf{216}}$	4.34	4.27	4.39	4.31	4.19
36	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	MedInstruct $_{\mathbf{10K}}$	MedInstruct-test $_{\mathbf{216}}$	3.72	3.84	3.55	3.44	3.42
37	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	MedInstruct $_{\mathbf{25K}}$	MedInstruct-test $_{\mathbf{216}}$	4.39	4.28	4.44	4.33	4.21
38	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	MedInstruct $_{\mathbf{25K}}$	MedInstruct-test $_{\mathbf{216}}$	3.87	3.92	3.74	3.52	3.43
39	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	MedInstruct $_{\mathbf{52K}}$	MedInstruct-test $_{\mathbf{216}}$	4.41	4.32	4.51	4.69	4.53
40	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	MedInstruct $_{\mathbf{52K}}$	MedInstruct-test $_{\mathbf{216}}$	4.15	4.11	4.10	4.24	4.30
41	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	Alpaca $_{\mathbf{1K}}$	just-eval-instruct $_{\mathbf{1k}}$	4.07	4.85	4.06	3.4	3.64
42	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	Alpaca $_{\mathbf{1K}}$	just-eval-instruct $_{\mathbf{1k}}$	3.19	4.42	3.31	2.72	3.25
43	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	Alpaca $_{\mathbf{10K}}$	just-eval-instruct $_{\mathbf{1k}}$	4.05	4.84	4.02	3.26	3.54
44	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	Alpaca $_{\mathbf{10K}}$	just-eval-instruct $_{\mathbf{1k}}$	3.13	4.13	3.23	2.67	3.24
45	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	Alpaca $_{\mathbf{25K}}$	just-eval-instruct $_{\mathbf{1k}}$	4.03	4.79	4.02	3.29	3.54
46	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	Alpaca $_{\mathbf{25K}}$	just-eval-instruct $_{\mathbf{1k}}$	3.04	4.49	3.11	2.64	3.28
47	Phi-1.5 $_{\mathbf{1.3B}}$_LFT	Alpaca $_{\mathbf{52K}}$	just-eval-instruct $_{\mathbf{1k}}$	4.07	4.81	4.03	3.33	3.56
48	Phi-1.5 $_{\mathbf{1.3B}}$_SFT	Alpaca $_{\mathbf{52K}}$	just-eval-instruct $_{\mathbf{1k}}$	3.93	4.2	3.65	3.88	3.49

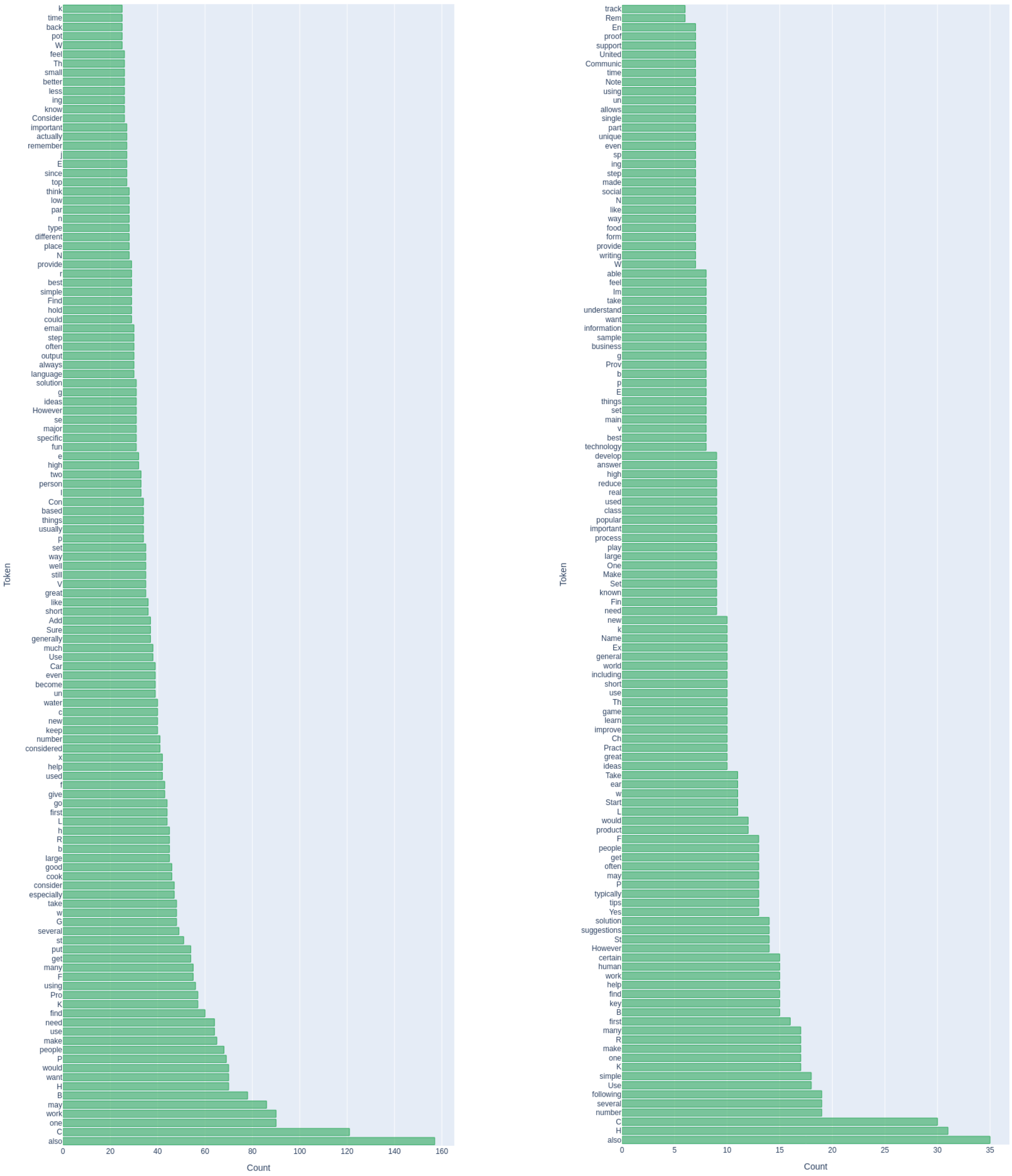
Table 15. Results table 3

	Models	Train Set	Test Set	Coding	Info-Seek	Math	Procedure	Reasoning	Role-Play	Writing
1	gpt-4-1106-preview	-	just-eval-instruct $_{1k}$	4.67	4.78	4.57	4.87	4.80	4.76	4.76
2	LLaMa-2 $_{7B}$-chat-hf	-	just-eval-instruct $_{1k}$	3.31	3.73	2.54	4.07	3.93	3.94	4.07
3	LLaMa-2 $_{13B}$-chat-hf	-	just-eval-instruct $_{1k}$	3.27	4.02	2.84	4.22	4.09	4.11	4.32
4	LLaMa-2 $_{70B}$-chat-hf	-	just-eval-instruct $_{1k}$	4.02	4.41	3.67	4.76	4.56	4.45	4.81
5	LLaMa-2 $_{7B}$_LFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.38	2.73	1.81	2.53	2.55	2.15	2.36
6	LLaMa-2 $_{7B}$_SFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	1.78	2.32	1.62	2.21	2.43	2.46	2.43
7	LLaMa-2 $_{7B}$_LFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	2.37	2.78	1.72	2.87	2.68	2.65	2.63
8	LLaMa-2 $_{7B}$_SFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	1.76	2.48	1.25	2.45	2.58	2.05	2.3
9	LLaMa-2 $_{7B}$_LFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	2.47	3.11	1.95	3.0	3.1	2.72	2.97
10	LLaMa-2 $_{7B}$_SFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	1.79	2.34	1.51	2.4	2.45	2.06	2.27
11	LLaMa-2 $_{7B}$_LFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	2.45	3.17	1.99	2.99	3.13	2.86	3.1
12	LLaMa-2 $_{7B}$_SFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	1.75	2.26	1.53	2.23	2.5	2.29	2.69
13	LLaMa-2 $_{7B}$_LFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	2.57	3.17	2.09	3.02	3.13	2.96	3.13
14	LLaMa-2 $_{7B}$_SFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	1.88	2.36	1.51	2.41	2.58	2.52	2.74
15	LLaMa-2 $_{7B}$_LFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	2.74	3.21	2.09	3.17	3.15	2.97	3.3
16	LLaMa-2 $_{7B}$_SFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	1.54	2.15	1.41	2.28	2.39	2.3	2.48
17	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{1K}$	just-eval-instruct $_{1k}$	2.66	3.39	2.01	3.29	3.44	2.92	3.12
18	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{1K}$	just-eval-instruct $_{1k}$	1.61	2.2	1.37	2.33	2.37	2.11	2.18
19	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{10K}$	just-eval-instruct $_{1k}$	2.58	3.46	2.14	3.38	3.54	3.23	3.34
20	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{10K}$	just-eval-instruct $_{1k}$	1.85	2.45	1.3	2.57	2.84	2.54	2.79
21	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{25K}$	just-eval-instruct $_{1k}$	2.69	3.53	2.15	3.41	3.60	3.37	3.52
22	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{25K}$	just-eval-instruct $_{1k}$	1.95	2.43	1.31	2.56	2.84	2.58	2.86
23	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{50K}$	just-eval-instruct $_{1k}$	2.82	3.6	2.16	3.45	3.67	3.48	3.6
24	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{50K}$	just-eval-instruct $_{1k}$	2.1	2.41	1.30	2.38	2.83	2.67	2.91
25	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{150K}$	just-eval-instruct $_{1k}$	2.76	3.6	2.1	3.6	3.64	3.48	3.60
26	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{150K}$	just-eval-instruct $_{1k}$	2.19	2.53	1.46	2.66	2.93	2.84	3.04
27	LLaMa-2 $_{7B}$_LFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	2.92	3.63	2.13	3.57	3.73	3.6	3.67
28	LLaMa-2 $_{7B}$_SFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	2.32	2.81	2.05	2.80	3.04	2.96	3.21
29	LLaMa-2 $_{7B}$_LFT	filtered-dolly $_{3k}$	just-eval-instruct $_{1k}$	2.31	2.79	2.24	2.65	2.67	2.24	2.45
30	LLaMa-2 $_{7B}$_SFT	filtered-dolly $_{3k}$	just-eval-instruct $_{1k}$	1.7	2.05	1.46	1.99	2.15	1.97	2.02
31	LLaMa-2 $_{7B}$_LFT	Evol-Instruct $_{70k}$	just-eval-instruct $_{1k}$	2.78	3.56	2.08	3.55	3.66	3.49	3.76
32	LLaMa-2 $_{7B}$_SFT	Evol-Instruct $_{70k}$	just-eval-instruct $_{1k}$	2.23	2.68	1.73	2.77	3.05	2.98	3.19
33	LLaMa-2 $_{7B}$_LFT	LIMA-Simple $_{1k}$	just-eval-instruct $_{1k}$	2.53	2.97	1.9	2.85	2.96	2.7	2.87
34	LLaMa-2 $_{7B}$_SFT	LIMA-Simple $_{1k}$	just-eval-instruct $_{1k}$	1.78	2.29	1.39	2.09	2.51	2.25	2.28
35	LLaMa-2 $_{7B}$_LFT	chatgpt $_{9k}$	just-eval-instruct $_{1k}$	2.82	3.53	2.93	3.28	3.35	3.52	
36	LLaMa-2 $_{7B}$_SFT	chatgpt $_{9k}$	just-eval-instruct $_{1k}$	2.58	3.13	1.82	3.01	3.15	2.72	2.93
37	LLaMa-2 $_{7B}$_NFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	1.55	2.37	1.62	2.36	2.65	2.46	2.58

Table 16. Results table 3

	Models	**Train Set**	**Test Set**	**Coding**	**Info-Seek**	**Math**	**Procedure**	**Reasoning**	**Role-Play**	**Writing**
1	LLaMa-2 $_{13B}$_LFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.6	2.8	2.1	2.79	2.78	2.36	2.58
2	LLaMa-2 $_{13B}$_SFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	1.93	2.5	1.46	2.19	2.58	1.97	2.39
3	LLaMa-2 $_{13B}$_LFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	2.45	2.97	2.08	2.95	2.78	2.48	2.83
4	LLaMa-2 $_{13B}$_SFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	1.6	2.32	1.34	2.23	2.45	1.97	2.22
5	LLaMa-2 $_{13B}$_LFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	2.78	3.25	2.26	3.24	3.25	2.98	3.27
6	LLaMa-2 $_{13B}$_SFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	1.86	2.44	1.67	2.27	2.55	2.11	2.49
7	LLaMa-2 $_{13B}$_LFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	2.78	3.28	2.46	3.29	3.3	3.09	3.39
8	LLaMa-2 $_{13B}$_SFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	1.87	2.33	1.51	2.32	2.52	2.42	2.8
9	LLaMa-2 $_{13B}$_LFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	2.75	3.26	2.55	3.33	3.26	3.16	3.36
10	LLaMa-2 $_{13B}$_SFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	1.65	2.17	1.45	2.12	2.36	2.1	2.57
11	LLaMa-2 $_{13B}$_LFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	2.73	3.29	2.41	3.3	3.26	2.95	3.49
12	LLaMa-2 $_{13B}$_SFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	1.74	2.14	1.34	2.21	2.31	2.19	2.66
13	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{1K}$	just-eval-instruct $_{1k}$	2.95	3.49	2.56	3.61	3.62	3.31	3.39
14	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{1K}$	just-eval-instruct $_{1k}$	1.88	2.71	1.63	2.58	2.81	2.55	2.56
15	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{10K}$	just-eval-instruct $_{1k}$	3.0	3.64	2.36	3.69	3.69	3.43	3.44
16	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{10K}$	just-eval-instruct $_{1k}$	1.94	2.62	1.48	2.63	2.96	2.84	3.01
17	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{25K}$	just-eval-instruct $_{1k}$	3.02	3.67	2.42	3.71	3.73	3.51	3.50
18	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{25K}$	just-eval-instruct $_{1k}$	2.01	2.82	1.72	2.84	3.08	2.95	3.14
19	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{50K}$	just-eval-instruct $_{1k}$	3.14	3.75	2.39	3.72	3.76	3.77	3.81
20	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{50K}$	just-eval-instruct $_{1k}$	2.11	2.89	1.87	2.92	3.12	3.01	3.18
21	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{150K}$	just-eval-instruct $_{1k}$	3.17	3.79	2.44	3.75	3.88	3.82	3.91
22	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{150K}$	just-eval-instruct $_{1k}$	2.29	2.95	1.98	3.02	3.21	3.27	3.38
23	LLaMa-2 $_{13B}$_LFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	3.20	3.81	2.54	3.80	3.91	3.94	3.99
24	LLaMa-2 $_{13B}$_SFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	2.36	3.01	2.31	3.14	3.32	3.36	3.45
25	LLaMa-2 $_{13B}$_LFT	LIMA-Simple $_{1k}$	just-eval-instruct $_{1k}$	2.72	3.23	2.23	3.31	3.18	2.83	3.0
26	LLaMa-2 $_{13B}$_SFT	LIMA-Simple $_{1k}$	just-eval-instruct $_{1k}$	1.77	2.4	1.75	2.31	2.67	2.23	2.32
27	LLaMa-2 $_{70B}$_LFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	3.44	3.59	2.65	3.46	3.69	3.87	3.63
28	LLaMa-2 $_{70B}$_SFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.42	2.53	2.33	2.45	2.61	2.45	2.78
29	LLaMa-2 $_{70B}$_LFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	3.32	3.54	2.59	3.40	3.58	3.72	3.58
30	LLaMa-2 $_{70B}$_SFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	2.36	2.49	2.31	2.41	2.57	2.43	2.71
31	LLaMa-2 $_{70B}$_LFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	3.57	3.72	2.92	3.73	3.88	4.01	3.92
32	LLaMa-2 $_{70B}$_SFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	3.21	2.86	2.64	2.76	2.87	2.76	3.03
33	LLaMa-2 $_{70B}$_LFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	3.89	3.99	3.46	4.01	4.12	4.23	4.19
34	LLaMa-2 $_{70B}$_SFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	3.42	3.14	3.05	3.10	3.33	3.11	3.45
35	LLaMa-2 $_{70B}$_LFT	Tulu-V2-Mix $_{1k}$	just-eval-instruct $_{1k}$	4.01	4.14	3.53	4.07	4.12	4.30	4.22
36	LLaMa-2 $_{70B}$_SFT	Tulu-V2-Mix $_{1k}$	just-eval-instruct $_{1k}$	3.71	3.42	3.32	3.13	3.76	3.32	3.72
37	LLaMa-2 $_{70B}$_LFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	4.31	4.44	3.81	4.27	4.42	4.67	4.57
38	LLaMa-2 $_{70B}$_SFT	Tulu-V2-Mix $_{326k}$	just-eval-instruct $_{1k}$	3.92	3.72	3.62	3.35	4.11	3.67	4.01
39	Mistral-v0.1 $_{7B}$_LFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	3.1	2.99	2.6	3.1	2.86	2.63	2.8
40	Mistral-v0.1 $_{7B}$_SFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	1.03	1.19	1.05	1.18	1.15	1.17	1.17
41	Mistral-v0.1 $_{7B}$_LFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	3.14	3.01	2.5	3.28	2.85	2.59	3.0
42	Mistral-v0.1 $_{7B}$_SFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	1.28	1.54	1.19	1.54	1.63	1.43	1.59
43	Mistral-v0.1 $_{7B}$_LFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	3.67	4.09	3.15	4.12	4.21	4.1	3.98
44	Mistral-v0.1 $_{7B}$_SFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	1.26	1.55	1.53	1.49	1.55	1.47	1.65
45	Mistral-v0.1 $_{7B}$_LFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	3.79	4.06	3.37	4.09	4.12	4.13	4.14
46	Mistral-v0.1 $_{7B}$_SFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	1.23	1.41	1.32	1.20	1.37	1.26	1.45
47	Mistral-v0.1 $_{7B}$_LFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	3.73	4.11	3.32	4.18	4.22	4.13	4.23
48	Mistral-v0.1 $_{7B}$_SFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	1.12	1.18	1.21	1.19	1.17	1.13	1.23
49	Mistral-v0.1 $_{7B}$_LFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	3.84	4.09	3.37	4.13	4.18	3.98	4.09
50	Mistral-v0.1 $_{7B}$_SFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	1.18	1.22	1.34	1.21	1.22	1.3	1.26
51	Phi-1.5 $_{1.3B}$_LFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	2.78	2.79	2.26	2.97	2.92	2.72	3.03
52	Phi-1.5 $_{1.3B}$_SFT	databricks-dolly $_{15k}$	just-eval-instruct $_{1k}$	1.9	2.31	1.79	2.36	2.58	2.57	2.55
53	Phi-1.5 $_{1.3B}$_LFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	3.02	2.95	2.46	3.12	2.65	2.39	2.97
54	Phi-1.5 $_{1.3B}$_SFT	LIMA $_{1K}$	just-eval-instruct $_{1k}$	2.58	2.64	2.32	2.51	2.42	2.12	2.78
55	Phi-1.5 $_{1.3B}$_LFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	3.24	4.09	3.15	4.12	4.21	4.10	3.98
56	Phi-1.5 $_{1.3B}$_SFT	Alpaca $_{1K}$	just-eval-instruct $_{1k}$	1.24	1.51	1.33	1.47	1.58	1.57	1.87
57	Phi-1.5 $_{1.3B}$_LFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	3.27	4.12	3.20	4.15	4.21	4.12	3.99
58	Phi-1.5 $_{1.3B}$_SFT	Alpaca $_{10K}$	just-eval-instruct $_{1k}$	1.45	1.51	1.47	1.65	1.75	1.82	2.01
59	Phi-1.5 $_{1.3B}$_LFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	3.29	4.13	3.22	4.17	4.23	4.11	4.00
60	Phi-1.5 $_{1.3B}$_SFT	Alpaca $_{25K}$	just-eval-instruct $_{1k}$	1.71	1.82	1.72	1.83	1.81	2.01	2.14
61	Phi-1.5 $_{1.3B}$_LFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	3.28	4.10	3.21	4.19	4.25	4.12	4.03
62	Phi-1.5 $_{1.3B}$_SFT	Alpaca $_{52K}$	just-eval-instruct $_{1k}$	1.59	1.92	1.72	1.97	2.13	2.23	3.02

Table 17. Results table 3



(a) Model fine-tuned using SFT.

(b) Model fine-tuned using LFT.

Figure 12. Frequency distribution plot of top 125 shifted and marginal for LLaMa-2 $_{7B}$ trained on LIMA $_{1K}$ and inferred on just-eval-instruct $_{1k}$. We visualize frequency distributions of models fine-tuned on both SFT and LFT to compare the knowledge learned by both models in the fine-tuning stage.