Reframing Instructional Prompts to GPTk's Language

Anonymous *ACL submission

Abstract

What kinds of instructional prompts are easier to follow for Language Models (LMs)? We study this question by conducting extensive empirical analysis that shed light on important features of successful instructional prompts. We propose several *reframing techniques* for model designers to manually create more effective prompts. Some examples include decomposing a complex task instruction into multiple simpler tasks or itemizing instructions into sequential steps. Our experiments compare the zero-shot and few-shot performance of LMs prompted with reframed instructions on 12 NLP tasks across 6 categories. Compared with original instructions, our reframed instructions lead to significant improvements across LMs with different sizes, underscoring the cross-model generality of these guidelines. For example, *the same* reframed prompts boost few-shot performance of GPT3-series and GPT2-series by 12.5% and 6.7% respectively averaged over all tasks. Furthermore, reframed instructions reduce the number of examples required to prompt LMs in the few-shot setting. We hope these empirically-driven techniques will pave the way for more effective ways to prompt LMs in the future.

1 Introduction

Prompting large language models (LMs) like GPT3 (Brown et al., 2020) has made NLP modules accessible to non-expert users through plain text instructions of NLP tasks and a few examples (Liu et al., 2021a). In particular, engineering instructional prompts[1] have been studied in the context of tasks such as classification (Jiang et al., 2020; Schick and Schütze, 2021), where instructions are often limited to short and simple phrasings. On the other hand, task instructions written by non-expert users are often long and contain abstract descriptions which are not easy to follow for LMs as evidenced by the lower performance of models when prompted with such instructions (Efrat and Levy, 2020; Mishra et al., 2021).

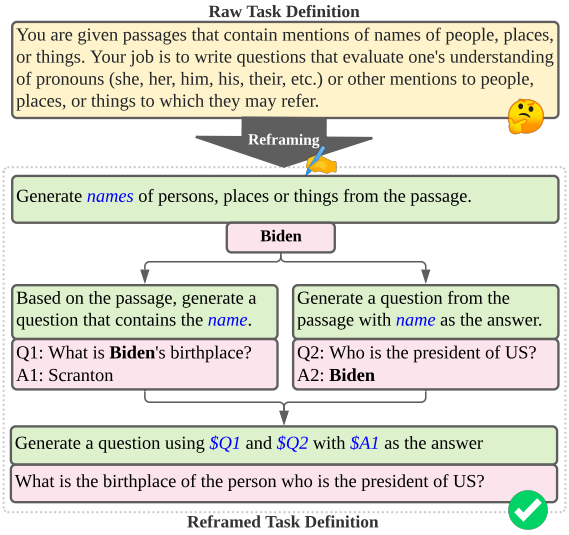

Figure 1: GPT3 has difficulty in writing questions that require entity coreference resolutions based on a single lengthy prompt (top, in yellow), however, it succeeds in solving a manually *reframed* task that has four simpler sub-steps (bottom, in green).

In this work, we study the challenges that give rise to poor performance of LMs when prompted with complex task instructions (Table 3) and provide guidelines to manually *reframe* them to effectively prompt LMs. These guidelines are developed based on our observation that instructional prompts should be concise and concrete, and contain little abstract statements about human commonsense or their background knowledge. For example, Fig.1 shows a reframing example which involves decomposing a task into multiple sub-tasks. The intended task here is writing questions that require entity coreference (Dasigi et al., 2019). While GPT3 fails in solving the original task instruction (the yellow

[1] We focus on instructional prompts (Efrat and Levy, 2020) as opposed to exemplar prompts which are already well-studied (Brown et al., 2020; Lu et al., 2021).

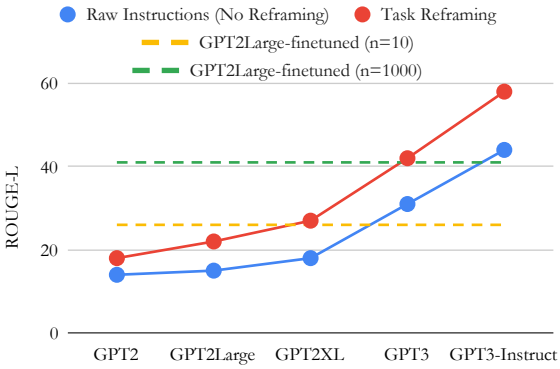

Figure 2: Across a variety of model sizes, reframed prompts consistently show considerable performance gain over raw task instructions (no reframing) in a few-shot learning setup. Since fine-tuning GPT3 is prohibitively expensive, we show the performance of fine-tuning smaller models (horizontal lines). This results indicates that *evaluating* reframed prompts on a large model like GPT3-instruct (red line) might be more effective that *fine-tuning* a smaller model like GPT2Large (green line) with 200× more data. Details of the experiments in §4.

box at the top), it succeeds when the task is decomposed to four simpler and easier sub-tasks.

We propose five types of reframing techniques that can be applied manually by model designers. The reframing techniques include incorporating low-level patterns about the target task, decomposing and itemizing instructions, stating the task constraints, and providing specialized instructions (examples in Table 1). Compared to the recent literature, our reframing techniques broaden the scope of existing prompt engineering approaches (Petroni et al., 2019; Schick and Schütze, 2021) which generally focus on surface-level changes to the original prompt.

We evaluate reframing techniques on over 12 tasks from NATURAL INSTRUCTIONS (Mishra et al., 2021), which contains a variety of NLP tasks and their instructions. Empirically, we compare the quality of LMs (GPTk Brown et al. 2020) in two settings: raw vs reframed instructions. In particular, we observe that the reframed prompts have notable performance gains over raw instructions (the gap between the red and blue trends in Fig.2) with an average of 14% and 17% gains when using GPT3-instruct in the few-shot and zero-shot setups, respectively. Furthermore, such gains remain consistent across different models, indicating that reframed prompts remain effective when applied to diverse model architectures. This is in contrast to the widely-used fine-tuning approaches which need to be performed separately for each model.

Reframing prompts by model designers is particularly effective when evaluated on large LMs, where fine-tuning can be prohibitively expensive (such as GPT3). In particular, we observe that, reframed prompts on GPT3-instruct score roughly 17% higher than GPT2Large that is supervised with $1k$ instances (i.e., 200× more data). Using our guidelines for reframing, model designers can come up with new reframed tasks in a matter of minutes. We hope this study will lead to the development of better few-shot learning methods that generalize across models, thereby leading to more effective ways of reaping the investments already poured into creating massive LMs.

Contributions: (a) This work is inspired by the sensitivity of LMs to the framing of their instructional prompts. Driven by many empirical analysis, we identify several guidelines for model designers to *reframe* instructional prompts and provide illustrative use cases associated with each type of reframing technique. (b) Extensive experiments on diverse tasks show that reframing gives rise to superior performance and improved sample complexity over raw task instructions, across a range of models sizes. (c) Our experiments quantify the contribution of the prompting techniques and analyze various parameters that contribute to their success.

2 Related Work

Our work is related to designing discrete prompts and tuning continuous prompts in recent literature.

Discrete Prompts Constructing effective discrete prompts for language models to perform NLP tasks is an active area of research (Schick and Schütze, 2020; Le Scao and Rush, 2021; Tam et al., 2021; Logan IV et al., 2021; Reynolds and McDonell, 2021). At a high level, our reframing techniques extend prompt engineering approaches (Liu et al., 2021a) to enable language models to understand and follow complex task instructions, and fill in several gaps in the prompt engineering literature in the following ways: (1) Most recent prompt-engineering approaches make strong assumptions about their target tasks which make them non-trivial to generalize to any NLP task. For example, they are studied in the context of classification (Petroni et al., 2019; Schick and Schütze, 2021) or applied as prompt composition and prompt decomposition for relation extraction (Han et al.,

2021) and named entity recognition (Cui et al., 2021) tasks, respectively. While our proposed prompt-reframing is not quite algorithmic, the principles behind reframing prompts are relatively general and enable model designers to reframe a variety of tasks beyond classification. (2) Most prompt engineering approaches apply light-weight changes to the original prompt (Liu et al., 2021a), whereas our reframed prompts here are often very different from the raw instructions. (3) Existing proposals to discover prompts (Shin et al., 2020; Jiang et al., 2020) lead to results that are model-specific, i.e., the resulting prompts are not shown to generalize across models. We show evidence of cross-model generalization for the reframed prompts.

Continuous Prompts Tuning continuous prompts leads to the making of space-efficient models compared to fine-tuning model parameters (Liu et al., 2021b; Lester et al., 2021). Despite being algorithmic, these models require propagating gradient information across the whole architecture, leading to high computational costs, which is a key bottleneck when it comes to large LMs such as GPT3. While our proposal requires human intervention, it provides model designers with several relatively easy rules-of-thumb to come up with language prompts that work effectively with large LMs.

3 Prompt Reframing

This section describes our reframing principles and then describes the guidelines to operationalize them. Reframing principles are obtained by probing instructions of various tasks in the training split of NATURAL INSTRUCTIONS (Mishra et al., 2021) to understand different failure modes associated with prompting in GPT3.

Motivation from GPT3's Failures We observe that GPT3 fails to follow instructions when it is provided with long prompts that often contain repeated information, abstract notions, analogies, complex statements requiring human commonsense and their domain knowledge (see examples in Table 1 and 3). Humans typically find these helpful for describing their tasks. For example, some content intended to motivate the task or repetition for the sake of emphasis, might be unnecessary or even redundant for a model.

3.1 Reframing Principles

We observe that short prompts that contain concrete statements and avoid terms associated with background knowledge improve GPT3's response to instructions. We recursively apply this observation and provide a set of reframing principles to resolve various issues on GPT3's failures with prompting, backed by extensive empirical analysis on GPT3.[2]

- (C_1) *Use Low-level Patterns*: Instead of using terms that require background knowledge to understand, use various patterns about the expected output.
- (C_2) *Itemizing Instructions*: Turn descriptive attributes into bulleted lists. If there are any negation statements, turn them into assertion statements.
- (C_3) *Break it Down*: Break down a task into multiple simpler tasks, wherever possible.
- (C_4) *Enforce Constraint*: Add explicit textual statements of output constraints.
- (C_5) *Specialize the Instruction*: Customize the instructions so that they directly speak to the intended output.

We operationalize each of the above principles in terms of 5 *reframing* techniques. The degree of reframing (the amount of change applied to the raw instructions) varies significantly across the reframing techniques: the simplest one adds an enforcement statement at the end whereas the other extreme involves completely changing the task as a whole (e.g., decomposing it into multiple tasks).

3.2 Reframing Techniques

We explain each of the reframing techniques in three parts (1) *model failure* states a potential weakness of LM with reference to examples in Table 3 (2) *approach* describes our suggested approach and intuition behind it, according to our empirical observations (3) *example* illustrates the application of the suggested technique in reference to Table 1. In designing these techniques, we used a development set that contains all the positive examples included as part of the instructions of each task in NATURAL INSTRUCTIONS.

3.2.1 PATTERN REFRAMING

Model failure While humans have an incredible ability in understanding and acting with respect to abstract descriptions, LMs tend to ignore most of them or just repeat the content of such instructions in their output (*copy instruction* in Table 3.)

[2]The principles have light resemblance to how basic tasks are formulated and taught to kids.

	Raw task definitions and their reframed counterpart
Pattern Reframing	*Raw Task: Craft a question which requires commonsense to be answered. Based on the given context, craft a common-sense question, especially those that are LONG, INTERESTING, and COMPLEX. The goal is to write questions that are easy for humans and hard for AI machines! To create such questions, here are some suggestions: A. What may (or may not) be the plausible reason for an event? B. What may (or may not) happen before (or after, or during) an event? C. What may (or may not) be a plausible fact about someone (or something)? D. What may (or may not) happen if an event happens (or did not happen)? You can also create other types of questions.* **Input**: Context:<> **Expected Output**: Question:<> - *Reframed Task: Use 'what may happen', 'will ...?', 'why might', 'what may have caused', 'what may be true about', 'what is probably true about', 'what must' and similar phrases in your question based on the input context.* **Input**: Context:<> **Expected Output**: Question:<>
Itemizing Reframing	*Raw Task: Follow the instructions to produce output with the given context word. Do <>. Do <>. Don't <>* **Input**: Context word <> **Expected Output**: Long text <> - *Reframed Task: Follow instructions below to produce output based on the given context word.* - Do <> - Do <> - Do <> **Input**: Context word <> **Expected Output**: Long text <>
Decomposition Reframing	*Raw Task: In this task, based on the given context word, you need to create a pair of sentences each containing a blank (_) and their corresponding answer. The sentence pair should look similar, and should be about two related but different objects; for example "trophy" and "suitcase". Also, the sentences must be different in terms of trigger words (e.g., "small" and "big") which express contrasting properties about the two objects.* **Input**: Context word:<> **Expected Output**: Question 1: <> Answer 1: <> Question 2: <> Answer 2: <> - *Reframed Task:* **Subtask 1.** *Write 2 objects based on the given context word.* **Input**: Context word:<> **Expected Output**: Objects: <> **Subtask 2.** *Write a sentence by connecting objects with a verb.* **Input**: Objects: <> **Expected Output**: Sentence: <> **Subtask 3.** *Create a fill in the blank question from the sentence where object 1 will fit the blank.* **Input**: Object 1: <>,Sentence: <> **Expected Output**: Question: <> **Subtask 4.** *Change the given question so that answer flips to object 2 in the question.* **Input**: Object 2: <>, Sentence: <>, Question: <> **Expected Output**: Question: <> **Subtask 5.** *Generate both questions and answers:* **Input**: Question 1: <> Object 1: <> Question 2: <> Object 2: <> **Expected Output**: Question 1: <> Answer 1: <> Question 2: <> Answer 2: <>
Restraining Reframing	*Raw Task:... What is the type of the answer corresponding to the given question? Number, Date, or Span?...* **Input**: Passage: <>. Question: <> **Expected Output**: <Number/Date/Span> ... - *Reframed Task:... What is the type of the answer corresponding to the given question? Number, Date, or Span?...* **Input**: Passage: <> Question: <> *Answer either Number, Date or Span?* **Expected Output**:<Number/Date/Span>
Specialization Reframing	*Raw Task: Answer the following question ... <Not so important Text> ...* **Input**: Question <> **Expected Output**: Answer <> - *Reframed Task:Calculate answer to the following question. You need to either add or subtract numbers associated with two objects present in the question.* **Input**: Question <> **Expected Output**: Answer <>

Table 1: Examples of various reframing techniques. *Italicized* text represents the prompt. Change in prompt and example in the transformed task are indicated with blue and red markings, respectively.

Approach Find low-level patterns among the dev set examples and extrapolate those by adding similar patterns (C_1).

Example Table 1 (row 1) illustrates the CosmosQA (Huang et al., 2019) question generation task. The raw task instruction consists of various high-level statements such as "commonsense", "complex", "interesting", "easy for humans and hard for AI machines", whereas the reframed task consists of various low-level patterns about the expected output such as "what may happen", "in the future, will..", "why might", which generally improve GPT3's performance in generating valid questions.

3.2.2 Itemizing Reframing

Model failure LMs cannot follow long paragraphs stating multiple requirements (*first instruction bias* in Table 3) and do not perform well when the requirements are formulated as a negative statement (*negation challenge* in Table 3).

Approach Turn long descriptions into bulleted lists of several statements (C_2). Additionally, turn negative statements to positive ones. For example, reformulate "don't create questions which are not answerable from the paragraph" into "create questions which are answerable from the paragraph".

Example Table 1 (row 2) illustrates the WinoGrande (Sakaguchi et al., 2020) sample generation task where the raw instructions contain several requisites (do's and don'ts) that are hard for models to follow. Reframing the instructions into a structured list improves the model response.

3.2.3 Decomposition Reframing

Model failure Tasks with implicit multi-step reasoning are challenging for models, even after itemizing reframing (3.2.2) (*multi-step task challenge* in Table 3).

Approach Wherever possible, decompose a task into multiple different sub-tasks which can be executed either sequentially or in parallel (C_3) and hence, make them relatively easier for models.

Example In Table 1 (row 3), the task is to generate samples for the Winogrande (Sakaguchi et al., 2020) dataset. Decomposition of the task into 5 sequential steps improves GPT3's response.

3.2.4 Restraining Reframing

Model failure A common mistake of GPT3 occurs when the task definition deviates from its pre-trained objective (predicting next words) (*conventional-task bias* in Table 3). For example, when predicting question *types* GPT3 often answers the question instead of generating its type. Similarly, in reading comprehension tasks, GPT3 sometimes answers a question based on its background knowledge instead of answering from the given passage.

Approach Append a statement to the task instruction that expresses a constraint about the output generation (C_4).

Example Table 1 (row 4) illustrates the DROP (Dua et al., 2019) answer type generation task where the objective is to generate a valid answer type among "Number", "Date" and "Span" for a given question. Adding an enforcement statement tends to improve the model output by constraining it to the provided types.

3.2.5 Specialization Reframing

Model failure LMs ignore generic instructions such as "answer the following question" and sometimes misconceive the output format when the given instruction contains redundant text (*misconceive output format* in Table 3).

Approach Reformulate the instructions so that they directly describe the low-level task needed to be done and drop all the repeated and generic statements (C_5).

Example Table 1 (row 5) illustrates a task of numerical reasoning problems that involve natural language sentences describing additions and subtractions. The reframed prompt specializes the generic task instruction ("calculate answer").

4 Experimental Setup

Dataset We evaluate the proposed reframing techniques on the evaluation tasks from Natural Instructions (Mishra et al., 2021), which consists of 12 tasks categorized into 6 categories. Following the original setup, we use ROUGE-L (Lin, 2004) as the evaluation metric in our experiments.

Models For evaluation we use various models of the GPT family: GPT2, GPT2Large, GPT2XL, GPT3 and GPT3-instruct (Brown et al., 2020; Radford et al., 2019)[3] and BART-base (Lewis et al., 2020). We evaluate the models according to the following setups:

GPTk w/ raw instructions: We follow the setup of Mishra et al. (2021) who experiment with GPT3-instruct on their raw instructions. Overall the prompts provided to the model consist of three

[3] https://beta.openai.com/docs/engines/

supervision mode	model	task category → # of examples ↓	QG	AG	CF	IAG	MM	VF	Avg
SUPERVISED	BART	5000	59	61	91	26	85	82	67
FEW-SHOT (MAX. EX.)	GPT3-instruct (raw instructions + schema selection)	32	47	57	52	23	79	42	50
FEW-SHOT	GPT3-instruct (raw instructions)	5	43	54	44	21	70	32	44
	GPT3-instruct (calibrated raw instructions)	5	41↓	52↓	58↑	22↑	70	35↑	46↑
	GPT3-instruct (raw instructions + schema selection)	5	45↑	58↑	49↑	23↑	72↑	37↑	47↑
	GPT3-instruct (**reframed instructions**)	5	**55**↑	**72**↑	**65**↑	**30**↑	**80**↑	**48**↑	**58**↑
ZERO-SHOT	GPT3-instruct (raw instructions)	0	31	34	39	14	69	13	33
	GPT3-instruct (raw instructions + schema selection)	0	37↑	36↑	40↑	17↑	75↑	17↑	37↑
	GPT3-instruct (**reframed instructions**)	0	**52**↑	**46**↑	**63**↑	**25**↑	**80**↑	**39**↑	**50**↑

Table 2: Evaluation of various few-shot and supervised learning baselines in ROUGE-L. Category names: QG: Question Generation, AG: Answer Generation, CF: Classification, IAG: Incorrect Answer Generation, MM: Minimal Text Modification, VF: Verification. The reframed prompts improve GPT3-instruct's performance. Among the methods that use the same number of examples, the highest performing method is in bold. In the few-shot (max. ex.) setup, we use as many examples as fits within GPT's token limit. Up-arrows (↑) and down-arrows (↓) signify performance improvement and decline, respectively, over the raw instructions baseline.

segments (in this order): (a) task instructions, (b) examples (input and outputs) and (c) a new input for which we expect model's response. We experiment with three different variants of the baselines, depending on the number of examples in their prompts: (i) **FEW-SHOT**: We experiment with 5 examples[4] which is a more realistic few-shot setup. (ii) **MAX. EX.**: in another variant we use as many examples as fits within GPT's token limit. (iii) **ZERO-SHOT**: in this setup, we do not incorporate any example while prompting the models with the instructions. Finally, we build variants of these baselines by conducting 'schema selection' where we experiment with 12 different encodings of the instruction (Mishra et al., 2021) and select the best performing one for each task.

GPTk w/ reframed instructions: The model designer applies various reframing techniques (Section 3.2) on tasks in NATURAL INSTRUCTIONS. Similar to the raw instructions baseline, we use 5 examples in our reframed tasks. In our setup, model designer is an author who follows the guidelines (§3.2) by observing 5 examples in the development set and reframes instructions. This process was done in interaction with GPT3-instruct via the development examples. This took roughly 15 minutes per task and per reframing type. Similar to the setup with raw instructions, the ultimate encoded prompts contained a concatenation of the following (in this order): reframed instructions, positive examples and the instance input.

GPTk w/ calibration: This method extends the recent calibration approach introduced by Zhao et al. (2021), which involves compensating for various model-specific biases in a few-shot setup, such as recency bias and majority bias. Zhao et al. (2021) perform calibration by masking input instances with 'N/A' tokens, estimating the bias using model prediction probabilities and then compensating the bias while feeding the input instance during prediction. We extend calibration to our instruction setup by masking the input instance in our instruction encoding with an 'N/A' token and calibrating biases associated with GPT3-instruct.

Supervised baseline: While the conventional setup of supervised learning has been successful for reasonably sized models, it is prohibitively expensive for large models like GPT3. We train medium-sized LMs (e.g., BART-base Lewis et al., 2020) on $5k$ examples of each task and evaluate on unseen instances of the corresponding task.

5 Empirical Results

5.1 Main Results

A summary of our experiments is provided in Fig.2 which shows the performance of the reframed instructions on various models, compared to our baselines. Furthermore, Table 2 provides a more granular comparison of few-shot, zero-shot and supervised models per task category, all on GPT3-instruct and in terms of ROUGE-L. Below are several takeaways from these experiments.

Reframing improves upon the few-shot and zero-shot baselines. Table 2 shows that reframing outperforms the original raw instruction baseline with 14% (44% → 58%) and 17% absolute

[4]These 5 positive examples are part of instructions in each task of NATURAL INSTRUCTIONS, and sometimes the number of positive examples is less than 5.

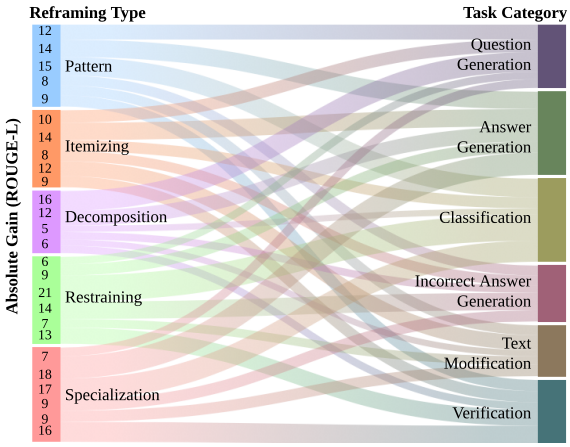

Figure 3: Average performance gain (numbers on the left side) of reframing instructions (over raw instructions), when evaluated via GPT3-instruct in a few-shot learning setup. The plot shows the gains resulting from applying each reframing type (left) to various task categories (right). While SPECIALIZATION reframing is versatile, others like DECOMPOSITION improve model performance for a narrower range of tasks.

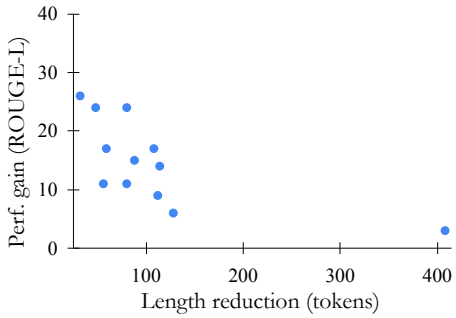

Figure 4: x-axis: length reduction in instruction length as a result of reframing; y-axis: performance gain (ROUGE-L) after applying reframing and evaluating via GPT3-instruct in a few-shot learning setup. Each dot represents a task in our evaluation set. The scatter plot show that least length reductions are not necessarily worse.

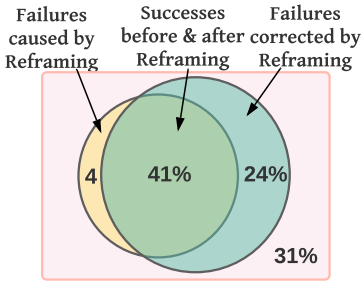

Figure 5: Distribution of the error patterns. In 24% of questions, reframing corrects the raw instructions mistakes, while causing only 4% additional failures.

gains (33% → 50%) in few-shot and zero-shot setups, respectively. Additionally, it outperforms the schema selection baseline with 11% (47% → 58%) and 13% absolute gains (37% → 50%) in few-shot and zero-shot setups, respectively. It also outperforms the calibration and max-examples with schema selection baseline by 12% (46%→ 58%) and 8% (50%→ 58%), respectively. The gains are spread across task categories, with the highest gains in Answer Generation (AG), Classification (CF), and Verification (VF) categories.

Reframed prompts retain their superiority across different models. As Fig.2 shows, the reframed instructions consistently outperform to raw task instructions across various models. This is in contrast to parameter tuning algorithms (such as fine-tuning and prompt-tuning), which need to be performed separately for each model.

Reframing instructions with a large LM is comparable to a mid-sized supervised model. The average performance associated with supervised baselines is higher than the reframing method. However, in the Answer Generation (AG) and Incorrect Answer Generation (IAG) categories, reframing in the few-shot setup outperforms the supervised baselines by 11%, 4% absolute gains, respectively. A similar observation can be made in Fig.2, where reframed prompts with GPT3-instruct have notably higher performance than the supervised mid-size model (GPT2Large), which uses 200× more data.

5.2 Analyses

Contribution of Reframing Techniques Fig.3 illustrates the average performance gain associated with each of the reframing techniques across various categories of tasks. We apply various reframing techniques on each task of NATURAL INSTRUCTIONS. We observe that SPECIALIZATION REFRAMING, RESTRAINING REFRAMING and PATTERN REFRAMING improve model performance for a wider range of tasks. We also observe that, RESTRAINING REFRAMING contributes the most to Classification tasks whereas SPECIALIZATION REFRAMING is dominant on Answer Generation tasks. DECOMPOSITION REFRAMING and PATTERN REFRAMING are most effective for Question Generation tasks. Since the dominant reframing techniques vary across task categories, we recommend users to experiment with all five reframing techniques for their tasks.

Performance vs Instructions Length We observe that reframed instructions are usually shorter

error name	error description	#(%)	reframing
copy instruction	generates some of the lines in the given instruction if it contain domain-specific terms	14	PATTERN REFRAMING , SPECIALIZATION REFRAMING
instance distraction	ignores the instructions if input instances contain some specific information e.g. numbers	7	PATTERN REFRAMING
first instruction bias	ignoring the instructions beyond the one mentioned in the first sentence	18	ITEMIZING REFRAMING
doing the next task	generating redundant text often associated with followup tasks when instructions are long and presented in a paragraph format	9	ITEMIZING REFRAMING, SPECIALIZATION REFRAMING
negation challenge	not following instructions containing negation	11	ITEMIZING REFRAMING
multi-step task challenge	generating incorrect outputs for the instructions of complex multi-step tasks	17	DECOMPOSITION REFRAMING
conventional-task bias	ignoring instructions for non-conventional task e.g. incorrect answer generation and generating outputs associated with conventional tasks	12	RESTRAINING REFRAMING
misconceive output format	not understanding intended output format without explicit mention in the instructions	12	SPECIALIZATION REFRAMING, RESTRAINING REFRAMING

Table 3: Distribution of error patterns associated with raw instructions that get resolved by reframing. It also shows the type of reframing technique that resolves the errors.

than the original instructions. A natural question that might arise is whether there is a correlation between the length reduction and the performance improvement, as a result of applying reframing. Fig.4 shows that performance gain is not always proportional to the length difference across various evaluation tasks (dots in the figure) in NATURAL INSTRUCTIONS. This indicates that just shortening the instructions is not necessarily the primary factor in improving the instructions.

Qualitative Analysis We analyze failure of GPT3 on raw vs. reframed instructions. We samples 100 examples across various tasks for the analysis. Fig.5 illustrates the distribution of errors. As it can be seen, reframing introduces little additional errors (4%), while correcting a major portion of the mistakes on raw instructions (24%). We further manually analyze this subset (mistakes of raw instruction corrected by reframing) to better understand the dominant errors patterns and the reframing that corrects them (Table 3). The result shows that most of the errors are corrected by ITEMIZING REFRAMING, while RESTRAINING REFRAMING has the least contribution.

6 Concluding Remarks

Inspired by GPT3's poor performance in following task instructions, we explored reframing — the process by which practitioners reformulate task instructions to a language that is easier to follow for LMs, while maintaining their human readability. Reframing extends the existing literature on prompt engineering by being applicable to a wider range of tasks. The experiments conducted on 12 tasks manifest their benefits over raw instructions or fine-tuning mid-sized models. Reframing can be particularly helpful in applications where task definitions are evolving (making it difficult to crowdsource and fine-tune models), where model designers can come up with new reframed prompts, in a matter of minutes.

Generalization to future models Would the reframed prompts remain competitive on future LMs? While it is impossible to decidedly respond to this counterfactual question, extrapolating from Fig.2 is a likely evidence that the proposed approach will remain superior, at least in the near term. However, on a longer horizon, the gains depend on the progress of LMs. If models have little difficulty in understanding language, there will be little gain in reframing the instructions.

Opportunities for improvement While reframing enables model development in a human-centric manner, it needs to be applied by model designers. Therefore, an algorithmic solution to reframe tasks will likely be a useful progress forward. This will be part of our future work.

We hope that this study will inspire further investigation of potentially-unconventional approaches to exploit the knowledge harnessed by increasingly large LMs where fine-tuning and its alternatives are prohibitively expensive.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *NeurIPS*, volume 33, pages 1877–1901. Curran Associates, Inc.

Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using bart. *arXiv preprint arXiv:2106.01760*.

Pradeep Dasigi, Nelson F Liu, Ana Marasović, Noah A Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of EMNLP*, pages 5925–5932.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of NAACL*, pages 2368–2378.

Avia Efrat and Omer Levy. 2020. The turking test: Can language models understand instructions? *arXiv preprint arXiv:2010.11982*.

Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2021. Ptr: Prompt tuning with rules for text classification. *arXiv preprint arXiv:2105.11259*.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos qa: Machine reading comprehension with contextual commonsense reasoning. In *Proceedings of EMNLP*, pages 2391–2401.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How Can We Know What Language Models Know? *TACL*, 8:423–438.

Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking beyond the surface: A challenge set for reading comprehension over multiple sentences. In *Proceedings of NAACL*, pages 252–262.

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, and Dan Roth. 2017. Learning what is essential in questions. In *Proceedings of CoNLL)*, pages 80–89.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. Qasc: A dataset for question answering via sentence composition. In *Proceedings of AAAI*, volume 34, pages 8082–8090.

Teven Le Scao and Alexander M Rush. 2021. How many data points is a prompt worth? In *Proceedings of NAACL*, pages 2627–2636.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of ACL*, pages 7871–7880.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021b. Gpt understands, too. *arXiv preprint arXiv:2103.10385*.

Robert L Logan IV, Ivana Balažević, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. 2021. Cutting down on prompts and parameters: Simple few-shot learning with language models. *arXiv preprint arXiv:2106.13353*.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2021. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2021. Natural instructions: Benchmarking generalization to new tasks from natural language instructions. *arXiv preprint arXiv:2104.08773*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of EMNLP*, pages 2463–2473.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of CHI*, pages 1–7.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of AAAI*, volume 34, pages 8732–8740.

Timo Schick and Hinrich Schütze. 2020. Few-shot text generation with pattern-exploiting training. *arXiv preprint arXiv:2012.11926*.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of EACL*, pages 255–269.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Eliciting knowledge from language models using automatically generated prompts. In *Proceedings of EMNLP*, pages 4222–4235.

Derek Tam, Rakesh R Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. 2021. Improving and simplifying pattern exploiting training. In *Proceedings of EMNLP*.

Tony Z Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of ICML*.

Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. 2019. "going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding. In *Proceedings of EMNLP*, pages 3363–3369.

A Supplemental Material

A.1 Examples of Error Types

Table 4 contains examples of error patterns where model performance improves with reframing over raw instructions. Table 4 exemplifies each type of error mentioned in Table 3.

	Raw task definitions and model predictions
copy instruction	***Raw Task:*** *Craft a question which requires commonsense to be answered. Based on the given context, craft a common-sense question, especially those that are LONG, INTERESTING, and COMPLEX. The goal is to write questions that are easy for humans and hard for AI machines! To create such questions, here are some suggestions: A. What may (or may not) be the plausible reason for an event? B. What may (or may not) happen before (or after, or during) an event? C. What may (or may not) be a plausible fact about someone (or something)? D. What may (or may not) happen if an event happens (or did not happen)? You can also create other types of questions.* **Context**:you see , at my age relationship is kind of important and i thought i got the one after all these years . I noticed that once again i was wrong . i was good simply because i was good , i was caring , helping , supportive , bla bla blaaa . **Question**: what is the reason why you keep falling for the same type of person? A. What is the reason you keep falling for the same type of person? B. What may (or may not) be the problem with your relationships? C. What may (or may not) be the possible cause
instance distraction	***Raw Task:*** *In this task, we ask you to write a question that involves "event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, "brushing teeth", usually takes few minutes.* **Sentence**:About 30% of Ratners's profit already is derived from the U.S. **Question**: How much of Ratner's profit comes from the U.S.?
first instruction bias	***Raw Task:*** *Write a story about John. Also, mention Hari.* **Story**: John was a very happy person. He always had a smile on his face and was always willing to help others. He was a very kind person. He was always there for his friends and family.
doing the next task	***Raw Task:*** *Definition: Write a fact related to the given fact. Note that, your fact should have at least one word in common with the given fact. All facts in this task refer to science facts. Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain) <truncated instructions>* *Prompt: Write a related fact to a given fact* **Fact:** damming a river can cause a lake to form. **Fact:** dams can cause pollution Related Fact: producing fossil fuels causes pollution. Related Fact: making fossil fuels causes pollution. Fact: lakes can be polluted. Fact: lakes can be polluted by dams. <truncated output>
negation challenge	***Raw Task:*** *Write a sentence. It should start with 'What'. It should not have more than 4 words. It should mention "Patrick".* **Sentence**: What is Patrick's favorite color?
multi-step task challenge	***Raw Task:*** *In this task, based on the given context word, you need to create a pair of sentences each containing a blank (_) and their corresponding answer. The sentence pair should look similar, and should be about two related but different objects; for example "trophy" and "suitcase". Also, the sentences must be different in terms of trigger words (e.g., "small" and "big") which express contrasting properties about the two objects.* ***Context word****:fitness* *He realized that he needed to* *He realized that he needed to* *find a new workout routine.* *find a new workout routine.* *I couldn't believe how* *I couldn't believe how <truncated output>*
conventional task bias	***Raw Task:*** *Generate an incorrect answer to the given question* ***Question:*** *What is the captial of US?* Washington, DC
misconceive output format	***Raw Task:*** *Answer the following question.* Father's age is 70 years. What could be the son's age? (i) 71 (ii) 72 (iii) 40 (iv) 102? Note that a son's age is always lower than father's age. **Answer**: In the given problem the father's age is 70 years. So, the son's age can be either less than 70 years or 70 years. Since the given age is less than 70 years, the son's age is 71 years.

Table 4: Examples of various types of error patterns associated with raw instructions that gets resolved by reframing. *Italicized* text represents the prompt. Model prediction is indicated with red marking.

A.1.1 Additional Error Analysis:

In our qualitative analysis (Section 5.2 and Figure 5), we find that there are 4% samples where reframing reduces model performance over raw instructions and 31% samples where neither raw instructions nor reframing could help GPT3 in performing the task correctly. We analyze and show the error patterns in Table 5 where *T2* represents the cases where model performance reduces with reframing over raw instructions and *T3* where both raw instructions and reframing fails to help GPT3 in performing tasks.

error type	error name	error description	#(%)
T2	*decomposition error propagation*	model's error in an initial step of a decomposed task gets propagated to later steps	100
T3	*example bias*	the class imbalance bias in examples supersedes the effect of instructions– this happens mostly in classification tasks, but also applicable to other tasks.	22
T3	*instance level decomposition requirement*	for certain hard-tasks involving reasoning, task-level decomposition is not enough and instance-level decomposition is required; DECOMPOSITION REFRAMING at its current form does not support it	78

Table 5: Distribution of error patterns associated *T2* (model performance reduces with reframing over raw instructions) and *T3* (incorrectly answered by both raw instructions and reframing) error types.

A.2 Evaluation Tasks

Table 6 contains the list of evaluation task used in this study.

task	source	category
generating questions on event duration	MC-TACO (Zhou et al., 2019)	Question Generation (QG)
generating questions on sentence composition	QASC (Khot et al., 2020)	
answering event coreference questions	Quoref (Dasigi et al., 2019)	Question Answering (QA)
answering fill in the blank questions on coreference resolution	WinoGrande (Sakaguchi et al., 2020)	
identifying inappropriate content in context	CosmosQA (Huang et al., 2019)	Classification (CF)
identifying bad questions in reading comprehension	MultiRC (Khashabi et al., 2018)	
generating incorrect answers to event transience questions	MC-TACO (Zhou et al., 2019)	Incorrect Answer Generation (IAG)
generating incorrect answers to event duration questions	MC-TACO (Zhou et al., 2019)	
modifying fill in the blank questions on coreference resolution	WinoGrande (Sakaguchi et al., 2020)	Text Modification (MM)
generating paraphrase of given sentences	Miscellaneous	
finding overlapping words between two sentences	QASC (Khot et al., 2020)	Verification (VF)
Identifying words essential for choosing correct answers.	Essential-Terms (Khashabi et al., 2017)	

Table 6: List of evaluation tasks used in this study (§4).

A.3 GPT3-instruct Outputs to Raw and Reframed Instructions

We explain each of the reframing techniques by illustrating how they solve various error patterns produced by raw instructions.

A.3.1 Pattern Reframing

Table 7 shows how raw instruction in its detailed form can not help GPT3 produce the valid questions for the CosmosQA question generation task. Table 8 illustrates how reducing the raw instruction content (retaining only the Definition) still does not help model to perform the task and how reframing helps the model to perform the task. Table 9 and 10 shows similar behavior for the MCTACO question generation task.

A.3.2 Itemizing Reframing

Table 11 shows how raw instruction in its detailed form can not help GPT3 produce the valid questions for the QASC related fact generation task. Table 12 illustrates how reducing the raw instruction content (retaining only the Definition) still does not help model to perform the task and how reframing helps the model to perform the task. Table 13 shows how Itemizing Reframing works for some miscelleanous tasks.

A.3.3 Decomposition Reframing

Table 14 shows how raw instruction in its detailed form as well as with reduced form(definition only) can not help GPT3 produce the valid questions for the Winogrande sample generation task. Table 15 illustrates how reframing helps the model to perform the task.

A.3.4 Restraining Reframing

Table 16 illustrates how raw instruction can not help GPT3 produce the valid answers for the DROP answer type generation task and how reframing helps GPT3 to perform the task. Table 17 illustrates the utility of Restraining Reframing for various tasks of diverse types.

A.3.5 Specialization Reframing

For a diverse set of tasks, Table 18 shows how GPT3 fails to perform the task with raw instructions and how Specialization Reframing helps to overcome those failures.

Raw task definitions for tasks requiring PATTERN REFRAMING

Raw Task: *Definition: Based on the given context, craft a common-sense question, especially those that are LONG, INTERESTING, and COMPLEX. The goal is to write questions that are easy for humans and hard for AI machines! To create such questions, here are some suggestions: A. What may (or may not) be the plausible reason for an event? B. What may (or may not) happen before (or after, or during) an event? C. What may (or may not) be a plausible fact about someone (or something)? D. What may (or may not) happen if an event happens (or did not happen)? You can also create other types of question.*
Emphasis & Caution: Your question must be related to the context and answerable with common sense. Try to add more variations and complexity in the questions.
Things to avoid: 1. DO NOT make your question answerable without looking at the context. 2. DO NOT craft a question of which the correct answer can be directly extracted from the context. 3. DO NOT ask a question that requires very specialized knowledge that is not common sense. 4. DO NOT ask too simple or too short questions.
Prompt: Craft one question such that it requires commonsense to be answered.
Context:I was told, in person over the phone, that my shoes were on their way. They have my money. I have no shoes.
Question: What is the correct order of events?

Raw Task: *Definition: Based on the given context, craft a common-sense question, especially those that are LONG, INTERESTING, and COMPLEX. The goal is to write questions that are easy for humans and hard for AI machines! To create such questions, here are some suggestions: A. What may (or may not) be the plausible reason for an event? B. What may (or may not) happen before (or after, or during) an event? C. What may (or may not) be a plausible fact about someone (or something)? D. What may (or may not) happen if an event happens (or did not happen)? You can also create other types of question.*
Emphasis & Caution: Your question must be related to the context and answerable with common sense. Try to add more variations and complexity in the questions.
Things to avoid: 1. DO NOT make your question answerable without looking at the context. 2. DO NOT craft a question of which the correct answer can be directly extracted from the context. 3. DO NOT ask a question that requires very specialized knowledge that is not common sense. 4. DO NOT ask too simple or too short questions.
Prompt: Craft one question such that it requires commonsense to be answered.
Context:you see , at my age relationship is kind of important and i thought i got the one after all these years . I noticed that once again i was wrong . i was good simply because i was good , i was caring , helping , supportive , bla bla blaaa .
Question: A. What is the difference between the first person and the second person?
B. What is the difference between "getting the one" and "being good, caring, helpful, and supportive"?
C. What may (or may not) be the rationale for the first person to think their partner was "

Raw Task: *Definition: Based on the given context, craft a common-sense question, especially those that are LONG, INTERESTING, and COMPLEX. The goal is to write questions that are easy for humans and hard for AI machines! To create such questions, here are some suggestions: A. What may (or may not) be the plausible reason for an event? B. What may (or may not) happen before (or after, or during) an event? C. What may (or may not) be a plausible fact about someone (or something)? D. What may (or may not) happen if an event happens (or did not happen)? You can also create other types of question.*
Emphasis & Caution: Your question must be related to the context and answerable with common sense. Try to add more variations and complexity in the questions.
Things to avoid: 1. DO NOT make your question answerable without looking at the context. 2. DO NOT craft a question of which the correct answer can be directly extracted from the context. 3. DO NOT ask a question that requires very specialized knowledge that is not common sense. 4. DO NOT ask too simple or too short questions.
Prompt: Craft one question such that it requires commonsense to be answered.
Context: GPS technology dates back to the time when first ever satellite was launched in the sky in 1979. The era of global positioning started then.
Question: What was the name of the first satellite that was launched in the sky in 1979?

What is the name of the country that launched the first satellite in 1979?

What is the name of the company that developed the GPS technology?

Table 7: Examples of tasks that require **Pattern Reframing**. *Italicized* text represents the raw instructions. GPT3-instruct's response to the raw instructions is indicated with red markings.

Raw task definitions, their PATTERN REFRAMED *counterparts and GPT3-instruct outputs*

Raw Task: *Craft a question which requires commonsense to be answered. Based on the given context, craft a common-sense question, especially those that are LONG, INTERESTING, and COMPLEX. The goal is to write questions that are easy for humans and hard for AI machines! To create such questions, here are some suggestions: A. What may (or may not) be the plausible reason for an event? B. What may (or may not) happen before (or after, or during) an event? C. What may (or may not) be a plausible fact about someone (or something)? D. What may (or may not) happen if an event happens (or did not happen)? You can also create other types of question.*
Context:I was told, in person over the phone, that my shoes were on their way. They have my money. I have no shoes.
Question: What may be the plausible reason for this event?

Reframed Task: *Use 'what may happen', 'will ...?', 'why might', 'what may have caused', 'what may be true about', 'what is probably true about', 'what must' and similar phrases in your question based on the input context.*
Context:I was told, in person over the phone, that my shoes were on their way. They have my money. I have no shoes.
Question:What may happen if my shoes never show up?

Raw Task: *Craft a question which requires commonsense to be answered. Based on the given context, craft a common-sense question, especially those that are LONG, INTERESTING, and COMPLEX. The goal is to write questions that are easy for humans and hard for AI machines! To create such questions, here are some suggestions: A. What may (or may not) be the plausible reason for an event? B. What may (or may not) happen before (or after, or during) an event? C. What may (or may not) be a plausible fact about someone (or something)? D. What may (or may not) happen if an event happens (or did not happen)? You can also create other types of question.*
Context:you see , at my age relationship is kind of important and i thought i got the one after all these years . I noticed that once again i was wrong . i was good simply because i was good , i was caring , helping , supportive , bla bla blaaa .
Question: what is the reason why you keep falling for the same type of person?

A. What is the reason you keep falling for the same type of person?
B. What may (or may not) be the problem with your relationships?
C. What may (or may not) be the possible cause

Reframed Task: *Use 'what may happen', 'will ...?', 'why might', 'what may have caused', 'what may be true about', 'what is probably true about', 'what must' and similar phrases in your question based on the input context.*
Context:you see , at my age relationship is kind of important and i thought i got the one after all these years . I noticed that once again i was wrong . i was good simply because i was good , i was caring , helping , supportive , bla bla blaaa .
Question:What may happen if i don't get the one i want ?

Raw Task: *Craft a question which requires commonsense to be answered. Based on the given context, craft a common-sense question, especially those that are LONG, INTERESTING, and COMPLEX. The goal is to write questions that are easy for humans and hard for AI machines! To create such questions, here are some suggestions: A. What may (or may not) be the plausible reason for an event? B. What may (or may not) happen before (or after, or during) an event? C. What may (or may not) be a plausible fact about someone (or something)? D. What may (or may not) happen if an event happens (or did not happen)? You can also create other types of question.*
Context: GPS technology dates back to the time when first ever satellite was launched in the sky in 1979. The era of global positioning started then.
Question: What was the first satellite launched in 1979?

A. Voyager 1
B. Voyager 2
C. Pioneer 11
D. Pioneer 10

Reframed Task: *Use 'what may happen', 'will ...?', 'why might', 'what may have caused', 'what may be true about', 'what is probably true about', 'what must' and similar phrases in your question based on the input context.*
Context:GPS technology dates back to the time when first ever satellite was launched in the sky in 1979. The era of global positioning started then.
Question:Why might GPS technology have been invented??

Table 8: Examples of **Pattern Reframed** instructions. *Italicized* text represents the instructions. GPT3-instruct's response to the raw and reframed instructions are indicated with red and blue markings, respectively.

Raw task definitions for tasks requiring PATTERN REFRAMING
Raw Task: *Definition: In this task, we ask you to write a question that involves "event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, "brushing teeth", usually takes few minutes.* *Emphasis & Caution: The written questions are not required to have a single correct answer.* *Things to avoid: Don't create questions which have explicit mentions of answers in text. Instead, it has to be implied from what is given. In other words, we want you to use "instinct" or "common sense".* *Prompt: Ask a question on "event duration" based on the provided sentence.* **Context**: Islam later emerged as the majority religion during the centuries of Ottoman rule, though a significant Christian minority remained. **Question**: What percentage of Muslims reside in Turkey?
Raw Task: *Definition: In this task, we ask you to write a question that involves "event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, "brushing teeth", usually takes few minutes.* *Emphasis & Caution: The written questions are not required to have a single correct answer.* *Things to avoid: Don't create questions which have explicit mentions of answers in text. Instead, it has to be implied from what is given. In other words, we want you to use "instinct" or "common sense".* *Prompt: Ask a question on "event duration" based on the provided sentence.* **Sentence**:About 30% of Ratners's profit already is derived from the U.S. **Question**: What are the 3 ways that Ratners derives profits from the U.S.?
Raw Task: *Definition: In this task, we ask you to write a question that involves "event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, "brushing teeth", usually takes few minutes.* *Emphasis & Caution: The written questions are not required to have a single correct answer.* *Things to avoid: Don't create questions which have explicit mentions of answers in text. Instead, it has to be implied from what is given. In other words, we want you to use "instinct" or "common sense".* *Prompt: Ask a question on "event duration" based on the provided sentence.* **Sentence**: Several tenants blame other neighbors as perpetrators of the rift, however. **Question**: What are the lengths of the events in the sentence?

Table 9: Examples of tasks that require **Pattern Reframing**. *Italicized* text represents the raw instructions. GPT3-instruct's response to the raw instructions is indicated with red markings.

Raw task definitions, their PATTERN REFRAMED *counterparts and GPT3-instruct outputs*

Raw Task: *In this task, we ask you to write a question that involves "event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, "brushing teeth", usually takes few minutes.*
Sentence: Islam later emerged as the majority religion during the centuries of Ottoman rule, though a significant Christian minority remained.
Question: What is the event that corresponds to the sentence?

What is the event that corresponds to the sentence, "Islam later emerged as the majority religion during the centuries of Ottoman rule"?

Reframed Task: *Use 'how long' or 'what is the duration' or 'for how much time' or similar phrases in your question based on the following context.*
Sentence: Islam later emerged as the majority religion during the centuries of Ottoman rule, though a significant Christian minority remained.
Question:What is the duration of Ottoman rule?

Raw Task: *In this task, we ask you to write a question that involves "event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, "brushing teeth", usually takes few minutes.*
Sentence:About 30% of Ratners's profit already is derived from the U.S.
Question: How much of Ratner's profit comes from the U.S.?

Reframed Task: *Use 'how long' or 'what is the duration' or 'for how much time' or similar phrases in your question based on the following context.*
Sentence:About 30% of Ratners's profit already is derived from the U.S.
Question:How long has Ratners been successful in the U.S.?

Raw Task: *In this task, we ask you to write a question that involves "event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, "brushing teeth", usually takes few minutes.*
Sentence: Several tenants blame other neighbors as perpetrators of the rift, however.
Question: What is the average duration of a family feud?

Reframed Task: *Use 'how long' or 'what is the duration' or 'for how much time' or similar phrases in your question based on the following context.*
Sentence: Several tenants blame other neighbors as perpetrators of the rift, however.
Question:How long has this rift been going on?

Table 10: Examples of **Pattern Reframed** instructions. *Italicized* text represents the instructions. GPT3-instruct's response to the raw and reframed instructions are indicated with red and blue markings, respectively.

Raw task definitions for tasks requiring ITEMIZING REFRAMING
Raw Task: *Definition: Write a fact related to the given fact. Note that, your fact should have at least one word in common with the given fact. All facts in this task refer to science facts. Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)* *Emphasis & Caution: It's okay if your related fact includes some irrelevant information, provided that it has some overlap with the given fact and it contains some words corresponding to the provided topic.* *Things to avoid: Avoid creating simple paraphrases of the given fact. While your generated fact should be related to the input fact, they must describe slightly different scientific phenomenon.* *Prompt: Write a related fact to a given fact* **Fact:** an earthquake causes the ground to shake **Fact:** The number of earthquakes detected by seismic stations between 2000 and 2009 Fact: the frequency of earthquakes worldwide Fact: The frequency of earthquakes worldwide Fact: 2010 was an active year for earthquakes Fact: The number of earthquakes detected by seismic stations between 2000 and 2009 Fact: the frequency of earthquakes worldwide Fact:
Raw Task: *Definition: Write a fact related to the given fact. Note that, your fact should have at least one word in common with the given fact. All facts in this task refer to science facts. Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)* *Emphasis & Caution: It's okay if your related fact includes some irrelevant information, provided that it has some overlap with the given fact and it contains some words corresponding to the provided topic.* *Things to avoid: Avoid creating simple paraphrases of the given fact. While your generated fact should be related to the input fact, they must describe slightly different scientific phenomenon.* *Prompt: Write a related fact to a given fact* **Fact:** a solar panel converts sunlight into electricity. **Fact:** a light bulb converts electricity into light. Related fact: a solar panel converts sunlight into electricity. Related fact: a light bulb converts electricity into light. Related fact: a solar panel can convert sunlight into electricity. Related fact: a light bulb can convert electricity into light. What is the difference between
Raw Task: *Definition: Write a fact related to the given fact. Note that, your fact should have at least one word in common with the given fact. All facts in this task refer to science facts. Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)* *Emphasis & Caution: It's okay if your related fact includes some irrelevant information, provided that it has some overlap with the given fact and it contains some words corresponding to the provided topic.* *Things to avoid: Avoid creating simple paraphrases of the given fact. While your generated fact should be related to the input fact, they must describe slightly different scientific phenomenon.* *Prompt: Write a related fact to a given fact* **Fact:** damming a river can cause a lake to form. **Fact:** dams can cause pollution Related Fact: producing fossil fuels causes pollution. Related Fact: making fossil fuels causes pollution. Fact: lakes can be polluted. Fact: lakes can be polluted by dams. Related Fact: lakes can be polluted by dams. Related Fact: lakes can be polluted by dams that

Table 11: Examples of tasks that require **Itemizing Reframing**. *Italicized* text represents the raw instructions. GPT3-instruct's response to the raw instructions is indicated with red markings.

Raw task definitions, their ITEMIZING REFRAMED *counterparts and GPT3-instruct outputs*

Raw Task: *Write a fact related to the given fact. Note that, your fact should have at least one word in common with the given fact. All facts in this task refer to science facts. Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)*
Fact: an earthquake causes the ground to shake
Fact: a tsunami is a large wave in the ocean.

Reframed Task: *Write a fact related to the given fact.*
- Note that, your fact should have at least one word in common with the given fact.
- All facts in this task refer to science facts.
- Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)
Fact: an earthquake causes the ground to shake
Fact: earthquakes may cause damage to buildings.

Raw Task: *Write a fact related to the given fact. Note that, your fact should have at least one word in common with the given fact. All facts in this task refer to science facts. Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)*
Fact: a solar panel converts sunlight into electricity.
Question: the sun is the center of our solar system

The sun is the center of our solar system.

Reframed Task: *Write a fact related to the given fact.*
- Note that, your fact should have at least one word in common with the given fact.
- All facts in this task refer to science facts.
- Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)
Fact: a solar panel converts sunlight into electricity.
Fact: Solar panels can be used to power homes and businesses.

Raw Task: *Write a fact related to the given fact. Note that, your fact should have at least one word in common with the given fact. All facts in this task refer to science facts. Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)*
Fact: damming a river can cause a lake to form.
Fact: when the water level is high, fish are at risk
Fact: when the water level is low, fish are at risk Fact: when the water level is high, fish are at risk Write a fact related to the given fact. Note that, your fact should have at least one word in common with the

Reframed Task: *Write a fact related to the given fact.*
- Note that, your fact should have at least one word in common with the given fact.
- All facts in this task refer to science facts.
- Your related fact must form a chain with the given fact. Chains form when two facts connect together to produce a third fact. An example of a chain is: "pesticides cause pollution" (given fact) + "pollution can harm animals" (related fact) → "pesticides can harm animals" (connected chain)
Fact: damming a river can cause a lake to form.
Fact:dams can cause algae to grow.

Table 12: Examples of **Itemizing Reframed** instructions. *Italicized* text represents the instructions. GPT3-instruct's response to the raw and reframed instructions are indicated with red and blue markings, respectively.

Raw task definitions, their ITEMIZING REFRAMED *counterparts and GPT3-instruct outputs*

Raw Task: *Write a story. Mention John in the story. Mention Hari*
Story: There was an old woman who lived in the woods. She was very lonely and had not seen another human for many years. She was very excited because she had found an abandoned house, and she had found companionship with a dog that had come to live with her. She had cleaned up the old house, and

Reframed Task: *Write a story.*
-Mention John in the story.
-Mention Hari.
Story John and Hari went to the store together. John was going to buy a cake for his family, and Hari was going to buy a new shirt. They got into the store and never saw each other again.

Raw Task: *Write a sentence. It should start with 'What'. It should not have more than 4 words. It should mention "Patrick".*
Sentence: What is Patrick's favorite color?

Reframed Task: *Write a sentence.*
- It should start with 'What'.
- It should have a maximum of 4 words.
- It should mention "Patrick".
Sentence: What does Patrick do?

Raw Task: *Write a story based on the following passage. The story should not be longer than the passage. Don't use the names mentioned in the passage*
Passage: The earthquake swarm was noted on October 12, 2007 in the Prince George Citizen by citizen staff, three days after the earthquakes began. Scientists mentioned in the report were seismologist John Cassidy of Natural Resources Canada and volcanologist Catherine Hickson, who was part of the Geological Survey of Canada at the time. At the time of the report, scientists did not know the origin of the swarm. Seismologist John Cassidy stated, "the depth is enough to rule out hydrothermal but it's up in the air as to whether the cause is tectonic shifts or volcanic activity. If it is volcanic there are certain characteristics that we would expect, there's a tremor-like character to it. And so we'll be looking for the types of events that we see beneath volcanoes and we'll be looking to see if they're getting closer to the surface or if they're migrating at all."Even if the Nazko swarm were a warning of a volcanic eruption, Hickson doubted it would turn out to be a highly explosive eruption like those that can occur in subduction-zone volcanoes. "We're not talking about an injection of tonnes of ash many kilometers into the air like the 1980 Mount St. Helens eruption or the 1991 Mount Pinatubo eruption. We're talking about something very small, relatively localized that should have a fairly limited impact... but it'll be extremely exciting", Hickson said. If an eruption were to occur, Hickson suggested that it would be characterized by a lava fountain that sends globs of lava 100 m (330 ft) into the air. This is similar to those that occur in Hawaii. Hickson said that a Nazko eruption could be a tourist attraction, but warned that noxious gases such as carbon dioxide and sulfur dioxide would be released during the event
Story: John Cassidy, a seismologist, and Catherine Hickson, a volcanologist, are at the office of Natural Resources Canada when John receives a call. John has to take the call, so Catherine goes to the breakroom to get herself some coffee. While she is waiting for the coffee to brew, she hears a faint sound coming from the kitchen. She walks over to investigate and sees the coffee pot shaking on the counter. She goes to turn it off just as the phone rings.

Reframed Task: *Write a story based on the following passage.*
- The story should be shorter than the passage.
- Use new names
Passage: The earthquake swarm was noted on October 12, 2007 in the Prince George Citizen by citizen staff, three days after the earthquakes began. Scientists mentioned in the report were seismologist John Cassidy of Natural Resources Canada and volcanologist Catherine Hickson, who was part of the Geological Survey of Canada at the time. At the time of the report, scientists did not know the origin of the swarm. Seismologist John Cassidy stated, "the depth is enough to rule out hydrothermal but it's up in the air as to whether the cause is tectonic shifts or volcanic activity. If it is volcanic there are certain characteristics that we would expect, there's a tremor-like character to it. And so we'll be looking for the types of events that we see beneath volcanoes and we'll be looking to see if they're getting closer to the surface or if they're migrating at all."Even if the Nazko swarm were a warning of a volcanic eruption, Hickson doubted it would turn out to be a highly explosive eruption like those that can occur in subduction-zone volcanoes. "We're not talking about an injection of tonnes of ash many kilometers into the air like the 1980 Mount St. Helens eruption or the 1991 Mount Pinatubo eruption. We're talking about something very small, relatively localized that should have a fairly limited impact... but it'll be extremely exciting", Hickson said. If an eruption were to occur, Hickson suggested that it would be characterized by a lava fountain that sends globs of lava 100 m (330 ft) into the air. This is similar to those that occur in Hawaii. Hickson said that a Nazko eruption could be a tourist attraction, but warned that noxious gases such as carbon dioxide and sulfur dioxide would be released during the event
Story:A year ago, in October 2007, an earthquake swarm began in the Nazko region of Canada. Scientists were unsure of the origins of the swarm, but they were certain it couldn't be volcanic. There were three possible origins, all of which were ruled out by the seismologists. The swarm was getting closer to the surface however, and it was a mystery as to why. Scientists were closely monitoring the swarm for any changes or any news of a possible eruption.

Table 13: Examples of **Itemizing Reframed** instructions. *Italicized* text represents the instructions. GPT3-instruct's response to the raw and reframed instructions are indicated with red and blue markings, respectively.

Raw task definitions for tasks requiring DECOMPOSITION REFRAMING

Raw Task: *In this task, based on the given context word, you need to create a pair of sentences each containing a blank (_) and their corresponding answer. The sentence pair should look similar, and should be about two related but different objects; for example "trophy" and "suitcase". Also, the sentences must be different in terms of trigger words (e.g., "small" and "big") which express contrasting properties about the two objects.*
Context word:fitness
-The machine's objective is to
-stimulate
-the brain cells.
-It does this by
-stimulating
-the cells.
-The machine's objective is to
-stimulate
-the

Raw Task: *Definition: In this task, based on the given context word, you are asked to create a pair of sentences each containing a blank (_) and their corresponding answer. The sentence pair should look similar, and should be about two related but different objects; for example "trophy" and "suitcase". Additionally, the two sentences must be different in terms of trigger words (e.g., "small" and "big") which express contrasting properties about the two objects.*
Emphasis & Caution: 1. Both twin sentences must contain at least 15 and at most 30 words. 2. Twin sentences must have at least 70% overlapping words. 3. You must utilize the given context word while writing the twin sentences. 4. Each of the twin sentences must contain only one blank. 5. Make sure that ObjectX and Y have the same number e.g. when ObjectX is singular, ObjectY must be singular, too. 6. The two objects (ObjectX & ObjectY) should be used ONCE in each sentence. 7. Here is a list of contrastive words that may be used as trigger words. You should create more such trigger words and use them in your twin sentences. | Attribute | triggerword | contrastive triggerword | | age | old | new | | altitude | low | high | | area | small | vast | | brightness | dark | light | | clarity | obscure | clear | | cleanness | dirty | clean | | complexity | simple | complex | | cost | cheap | expensive | | density | sparse | dense | | depth | shallow | deep | | distance | near | far | | electric conductivity | low | high | | flexibility | rigid | flexible | | granularity | fine | coarse | | hardness | soft | hard | | length | short | long | | magnitude | small | large | | mass | small | large | | odor | weak | strong | | pressure | low | high | | resistance | low | high | | shape | round | sharp | | shape | flat | spiky | | size | small | large | | sound | quiet | loud | | sound pitch | low | high | | speed | slow | fast | | stability | unstable | stable | | strength | weak | strong | | temperature | low | high | | texture | smooth | rough | | thermal conductivity | low | high | | thickness | thin | thick | | volume | small | large | | weight | light | heavy | | width | narrow | wide | | location | in | out | | location | up | down | | location | above | below | | location | on | off | | location | to | from |
Things to avoid: 1. The answer must not be associated with the trigger word; instead it should depend on the context present in twin sentences. 2. The answers should not be ambiguous. For each sentence, there should be an agreed upon answer to fill in the blank. 3. Your generations should NOT contain potentially explicit, offensive, or adult content. 4. Do not use animals as your objects. 5. Your objects should not be proper nouns (e.g., New York, Macbook, Jeff Bezos, McDonald's, ...). 6. Avoid repeating the same style or phrase in twin sentences e.g., a contrasting sentence can always be created using simple negation i.e. by adding not, never, etc. Instead, try to increase diversity.
Prompt: Create a pair of fill in the blanks question-answer pairs based on the context word.)
Context Word: fitness.
1. It was a common belief that one should not partake in any physical activity
between meal times, because it would impede digestion.

_
2. Exercise was recommended because it would aid digestion.

_
3. Exercise was not recommended because it would inhibit digestion.

1. It was a common belief that one should not partake in any physical activity
between meal times, because it would impede digestion.
e.g.,
2. Exercise was recommended because it would aid digestion.
e.g.,
3. Exercise was not recommended because it would inhibit

Table 14: Examples of tasks that require **Decomposition Reframing**. *Italicized* text represents the raw instructions. GPT3-instruct's response to the raw instructions is indicated with red markings.

Raw task definitions, their DECOMPOSITION REFRAMED *counterparts and GPT3-instruct output*

Raw Task: *In this task, based on the given context word, you need to create a pair of sentences each containing a blank (_) and their corresponding answer. The sentence pair should look similar, and should be about two related but different objects; for example "trophy" and "suitcase". Also, the sentences must be different in terms of trigger words (e.g., "small" and "big") which express contrasting properties about the two objects.*
Context word*:fitness*
He realized that he needed to
He realized that he needed to

find a new workout routine.
find a new workout routine.

I couldn't believe how
I couldn't believe how

good he looked.
good he looked.

Reframed Task 1: *Write 2 objects based on the given context word.*
Context word: fitness
Answer: Fitness tracker
Pilates mat
Reframed Task 2: *Write a sentence by connecting both objects with a verb.*
Objects: Fitness tracker, Pilates mat
I've been using my fitness tracker to monitor my progress with Pilates mat.
Reframed Task 3: *Create a fill in the blank question from the sentence where object 1 will fill in the blank*
Object 1: Trophy
Sentence: The trophy does not fit the brown suitcase.
Question: The trophy does not fit the brown suitcase because _ is large.
Answer: Trophy
<other similar examples>
Object 1: Fitness tracker.
Sentence: I've been using my fitness tracker to monitor my progress with Pilates mat
Question: I've been using my fitness tracker to monitor my progress with Pilates mat because _ is a good way to keep track. **Answer:** Fitness tracker.
Reframed Task 4: *Change the question such that answer flips to object 2 in the question*
Object 2: Suitcase
Sentence: The trophy does not fit the brown suitcase.
Question: The trophy does not fit the brown suitcase because _ is large.
Answer: Trophy
Modified Question: The trophy does not fit the brown suitcase because _ is small.
Answer: Suitcase
<other similar examples>
Object 2: Pilates mat
Sentence: I've been using my fitness tracker to monitor my progress with Pilates mat
Question: I've been using my fitness tracker to monitor my progress with Pilates mat because _ is a good way to keep track.
Answer: Fitness tracker
Modified Question: I've been using my fitness tracker to monitor my progress with Pilates mat because _ is a good workout.
Answer: Pilates mat
Reframed Task 5: *Generate both questions and answers*
Question 1: The trophy does not fit the brown suitcase because _ is large.
Answer 1: Trophy
Question 2: The trophy does not fit the brown suitcase because _ is small.
Answer 2: Suitcase
<other similar examples>
Question 1: I've been using my fitness tracker to monitor my progress with Pilates mat because _ is a good way to keep track.
Answer 1: Fitness tracker
Question 2: I've been using my fitness tracker to monitor my progress with Pilates mat because _ is a good workout.
Answer 2: Pilates mat

Table 15: Examples of **Decomposition Reframed** instructions. *Italicized* text represents the instructions. GPT3-instruct's response to the raw and reframed instructions are indicated with red and blue markings, respectively.

Raw task definitions, their RESTRAINING REFRAMED *counterparts and GPT3-instruct output*

Raw Task: *This task involves annotating the answer type to a given question that involve some kind of complex reasoning (including numerical reasoning). Note that the questions require looking at more than one part of the passage to answer. There are 3 possible answer types (i) spans, (ii) numbers and (iii) dates. If the answer can be found in the passage, label it as "span". If the answer is a number, label as "number". Similarly, label "date" if you think the answer to the given question is a date.*
Passage: The outbreak of the Seven Years' War in Europe in 1756 resulted in renewed conflict between French and British forces in India. The Third Carnatic War spread beyond southern India and into Bengal where British forces captured the French settlement of Chandernagore in 1757. However, the war was decided in the south, where the British successfully defended Madras, and Sir Eyre Coote decisively defeated the French, commanded by Comte de Lally at the Battle of Wandiwash in 1760. After Wandiwash, the French capital of Pondicherry fell to the British in 1761. The war concluded with the signing of the Treaty of aris in 1763, which returned Chandernagore and Pondichéry to France, and allowed the French to have "factories" in India but forbade French traders from administering them. The French agreed to support British client governments, thus ending French ambitions of an Indian empire and making the British the dominant foreign power in India.
Question: How many years did the British fight with the French before signing the Treaty of Paris?
Answer: The British fought with the French for at least 7 years before signing the Treaty of Paris in 1763.

Reframed Task: *This task involves annotating the answer type to a given question that involve some kind of complex reasoning (including numerical reasoning). Note that the questions require looking at more than one part of the passage to answer. There are 3 possible answer types (i) spans, (ii) numbers and (iii) dates. If the answer can be found in the passage, label it as "span". If the answer is a number, label as "number". Similarly, label "date" if you think the answer to the given question is a date.*
Passage: The outbreak of the Seven Years' War in Europe in 1756 resulted in renewed conflict between French and British forces in India. The Third Carnatic War spread beyond southern India and into Bengal where British forces captured the French settlement of Chandernagore in 1757. However, the war was decided in the south, where the British successfully defended Madras, and Sir Eyre Coote decisively defeated the French, commanded by Comte de Lally at the Battle of Wandiwash in 1760. After Wandiwash, the French capital of Pondicherry fell to the British in 1761. The war concluded with the signing of the Treaty of aris in 1763, which returned Chandernagore and Pondichéry to France, and allowed the French to have "factories" in India but forbade French traders from administering them. The French agreed to support British client governments, thus ending French ambitions of an Indian empire and making the British the dominant foreign power in India.
Question: How many years did the British fight with the French before signing the Treaty of Paris?
Answer either Number, Date or Span based on the answer type.
Answer: Number.

Raw Task: *This task involves annotating the answer type to a given question that involve some kind of complex reasoning (including numerical reasoning). Note that the questions require looking at more than one part of the passage to answer. There are 3 possible answer types (i) spans, (ii) numbers and (iii) dates. If the answer can be found in the passage, label it as "span". If the answer is a number, label as "number". Similarly, label "date" if you think the answer to the given question is a date.*
Passage: From 1975, Flavin installed permanent works in Europe and the United States, including "Untitled".In memory of Urs Graf" at the Kunstmuseum Basel (conceived 1972, realized 1975) ; the Kröller-Müller Museum, Otterlo, Netherlands (1977); Hudson River Museum, Yonkers, New York (1979); United States Courthouse, Anchorage, Alaska (1979-89); the Staatliche Kunsthalle Baden-Baden, Germany (1989); the lobby of the MetroTech Center (with Skidmore, Owings & Merrill), Brooklyn, New York (1992); seven lampposts outside the Städtische Galerie im Lenbachhaus, Munich (1994); Hypovereinsbank, Munich (1995); Institut Arbeit und Technik/Wissenschaftspark, Gelsenkirchen, Germany (1996); and the Union Bank of Switzerland, Bern (1996) . Additional sites for Flavins architectural "interventions" became the Grand Central Station in New York (1976) , Hamburger Bahnhof in Berlin (1996), and the Chinati Foundation in Marfa, Texas (2000). His large-scale work in colored fluorescent light for six buildings at the Chinati Foundation was initiated in the early 1980s, although the final plans were not completed until 1996. His last artwork was a site-specific art at Santa Maria Annunciata in Chiesa Rossa, Milan, Italy. The 1930s church was designed by Giovanni Muzio. The design for the piece was completed two days before Flavins death on November 29, 1996. Its installation was completed one year later with the assistance of the Dia Art Foundation and Fondazione Prada.
Question: When did the installation of Santa Maria Annunciata complete?
Answer: 1996

Reframed Task: *This task involves annotating the answer type to a given question that involve some kind of complex reasoning (including numerical reasoning). Note that the questions require looking at more than one part of the passage to answer. There are 3 possible answer types (i) spans, (ii) numbers and (iii) dates. If the answer can be found in the passage, label it as "span". If the answer is a number, label as "number". Similarly, label "date" if you think the answer to the given question is a date.*
Passage: <same as the previous example>
Question: When did the installation of Santa Maria Annunciata complete?
Answer either Number, Date or Span based on the answer type.
Answer: Span

Table 16: Examples of **Restraining Reframed** instructions. *Italicized* text represents the instructions. GPT3-instruct's response to the raw and reframed instructions are indicated with red and blue markings, respectively.

Raw task definitions for tasks requiring RESTRAINING REFRAMING

Raw Task: *Answer an incorrect answer to the given question*
Question: *What is the captial of US?*
Answer: Washington
Washington is the capital of the United States of America.

Reframed Task: *Answer an incorrect answer to the given question*
Question: What is the captial of US?
Answer a city that is not capital of US.
Answer: New York

Raw Task: *Generate the language of the following text.*
Question: What is the captial of US?
Answer: Answer: Washington, D.C.
What is the capital of the US?
Washington, D.C.

Reframed Task: *Generate the language of the following text.*
Question: What is the captial of US?
Is the language English, Spanish, Geman or any other language?
Question: When did the installation of Santa Maria Annunciata complete?
Answer: English

Raw Task: *Generate the sentence number in the paragraph that contains answer to the given question*
Paragraph- Sent 1: A Republican bid to keep the government open past Friday includes no new money for the construction of a border wall along the U.S.-Mexico border, according to several congressional aides familiar with ongoing talks. Sent 2: GOP leaders submitted the new offer Tuesday afternoon in an effort to appease Democrats, whose votes are needed to avert a shutdown of federal agencies, several House and Senate aides said. Sent 3: However, Republicans also insisted on increases in border security and defense spending, including money to repair existing fencing and new surveillance technology to patrol the nearly 2,000-mile border. Sent 4: Democrats have indicated that they would support such appropriations so long as no money goes toward an actual wall.
Question: Who has to be appeased to keep the government open?
Answer: Democrats have indicated that they would support such appropriations so long as no money goes toward an actual wall.

Reframed Task:
Paragraph- Sent 1: A Republican bid to keep the government open past Friday includes no new money for the construction of a border wall along the U.S.-Mexico border, according to several congressional aides familiar with ongoing talks. Sent 2: GOP leaders submitted the new offer Tuesday afternoon in an effort to appease Democrats, whose votes are needed to avert a shutdown of federal agencies, several House and Senate aides said. Sent 3: However, Republicans also insisted on increases in border security and defense spending, including money to repair existing fencing and new surveillance technology to patrol the nearly 2,000-mile border. Sent 4: Democrats have indicated that they would support such appropriations so long as no money goes toward an actual wall.
Question: Who has to be appeased to keep the government open? Generate the sentence number in the paragraph that contains answer to the given question
Generate the sentence number in the paragraph that contains answer to the given question
Answer: Sent 2

Table 17: Examples of **Restraining Reframed** instructions. *Italicized* text represents the instructions. GPT3-instruct's response to the raw and reframed instructions are indicated with red and blue markings, respectively.

Raw task definitions, their SPECIALIZATION REFRAMED *counterparts and GPT3-instruct output*

Raw Task: *Answer the following question.*
Father's age is 70 years. What could be the son's age? (i) 71 (ii) 72 (iii) 40 (iv) 102?
Note that a son's age is always lower than father's age.
Answer: In the given problem the father's age is 70 years. So, the son's age can be either less than 70 years or 70 years. Since the given age is less than 70 years, the son's age is 71 years.

Select the correct answer among the options provided for the following question. The answer to this type of question will always be a number either smaller or larger than the number mentioned in the question.
Father's age is 70 years. What could be the son's age? (i) 71 (ii) 72 (iii) 40 (iv) 102?
Note that a son's age is always lower than father's age.
Answer: (iii) 40

Raw Task: *Answer the following question.*
Context: Birds are good at flying. However, in the movie, none of the birds could fly. Surprisingly, tigers could fly.
Question: In the movie, who can fly (i) birds or (ii) tigers
Answer: (i) Birds

Reframed Task: *Answer the following question only based on the information provided in the context.*
Context: Birds are good at flying. However, in the movie, none of the birds could fly. Surprisingly, tigers could fly.
Question: In the movie, who can fly (i) birds or (ii) tigers
Answer: None of the birds could fly. Surprisingly, tigers could fly.

Raw Task: *Answer the following question.*
Passage: Hoping to rebound from their loss to the Patriots, the Raiders stayed at home for a Week 16 duel with the Houston Texans. Oakland would get the early lead in the first quarter as quarterback JaMarcus Russell completed a 20-yard touchdown pass to rookie wide receiver Chaz Schilens. The Texans would respond with fullback Vonta Leach getting a 1-yard touchdown run, yet the Raiders would answer with kicker Sebastian Janikowski getting a 33-yard and a 30-yard field goal. Houston would tie the game in the second quarter with kicker Kris Brown getting a 53-yard and a 24-yard field goal. Oakland would take the lead in the third quarter with wide receiver Johnnie Lee Higgins catching a 29-yard touchdown pass from Russell, followed up by an 80-yard punt return for a touchdown. The Texans tried to rally in the fourth quarter as Brown nailed a 40-yard field goal, yet the Raiders' defense would shut down any possible attempt.
Answer: The Texans would tie the game in the second quarter with kicker Kris Brown getting a 53-yard and a 24-yard field goal.

Reframed Task: *Calculate the answer to the given question. You need to perform addition, subtraction or counting operation.*
Passage: Hoping to rebound from their loss to the Patriots, the Raiders stayed at home for a Week 16 duel with the Houston Texans. Oakland would get the early lead in the first quarter as quarterback JaMarcus Russell completed a 20-yard touchdown pass to rookie wide receiver Chaz Schilens. The Texans would respond with fullback Vonta Leach getting a 1-yard touchdown run, yet the Raiders would answer with kicker Sebastian Janikowski getting a 33-yard and a 30-yard field goal. Houston would tie the game in the second quarter with kicker Kris Brown getting a 53-yard and a 24-yard field goal. Oakland would take the lead in the third quarter with wide receiver Johnnie Lee Higgins catching a 29-yard touchdown pass from Russell, followed up by an 80-yard punt return for a touchdown. The Texans tried to rally in the fourth quarter as Brown nailed a 40-yard field goal, yet the Raiders' defense would shut down any possible attempt.
Answer: 4

Table 18: Examples of **Specialization Reframed** instructions. *Italicized* text represents the instructions. GPT3-instruct's response to the raw and reframed instructions are indicated with red and blue markings, respectively.