Prompting Contrastive Explanations for Commonsense Reasoning Tasks

Bhargavi Paranjape[†*] **Julian Michael**[†] **Marjan Ghazvininejad**[*]
Hannaneh Hajishirzi[†ϵ] **Luke Zettlemoyer**[†*]
[†]Allen School of Computer Science & Engineering, University of Washington, Seattle, WA
[ϵ]Allen Institute of Artificial Intelligence, Seattle
[*]Facebook AI
{bparan,julianjm,lsz,hannaneh}@cs.washington.edu

Abstract

Many commonsense reasoning NLP tasks involve choosing between one or more possible answers to a question or prompt based on knowledge that is often implicit. Large pretrained language models (PLMs) can achieve near-human performance on such tasks, while providing little human-interpretable evidence of the underlying reasoning they use. In this work, we show how to use these same models to generate such evidence: inspired by the contrastive nature of human explanations, we use PLMs to complete explanation prompts which contrast alternatives according to the key attribute(s) required to justify the correct answer (for example, *peanuts are usually salty while raisins are sweet*). Conditioning model decisions on these explanations improves performance on two commonsense reasoning benchmarks, as compared to previous non-contrastive alternatives. These explanations are also judged by humans to be more relevant for solving the task, and facilitate a novel method to evaluate explanation faithfulness.

1 Introduction

Pretrained Language Models (PLMs) (Raffel et al., 2020; Lewis et al., 2020; Radford et al., 2019; Brown et al., 2020) have been shown to encode substantial amounts of knowledge in their parameters (Petroni et al., 2019; Talmor et al., 2020; Roberts et al., 2020) and have achieved impressive performance on commonsense reasoning (CSR) tasks without the use of external knowledge (Trinh and Le, 2018; Yang et al., 2020). However, these models provide little human-interpretable evidence of the intermediate commonsense knowledge or reasoning they use, and have been observed to overly rely on superficial dataset artifacts (Poliak et al., 2018; Geva et al., 2019). To overcome this limitation, recent work has shown that PLMs can explain themselves by *generating* free-form natural language explanations of their reasoning patterns (Rajani et al., 2019a; Camburu et al., 2018; Narang et al., 2020). However, the space of possible free-form explanations is incredibly large, inherently ambiguous, and difficult to annotate or evaluate (Wiegreffe et al., 2020; Latcinnik and Berant, 2020). Furthermore, quantifying the model's dependence on free-form explanations is also challenging (Camburu et al., 2020). We address these challenges by proposing an unsupervised method that uses contrastive prompts, which require the model to *explicitly* contrast different possible answers in its explanation (Table 1).

i) I picked up a bag of peanuts and raisins for a snack. I wanted a sweeter snack out so I ate the __ for now. *Contrastive Expl. - Peanuts are salty while raisins tend to be sweet.*
ii) The geese prefer to nest in the fields rather than the forests because in the __ predators are more hidden. *Contrastive Expl. - Forests are denser than fields*

Table 1: Examples of Winograd Schema Instances where the correct and incorrect answer choices are highlighted in blue and red respectively. Choices are *contrasted* along attributes like taste (for i) and density of vegetation (for ii) by humans to explain why they prefer some answer choice.

Our approach is based on a key observation: Many commonsense reasoning tasks require the comparison or contrast of plausible alternatives along a distinguishing attribute. For instance, in Table 1, the differentiating attributes for the two answer choices maybe taste (for i) and vegetation density (for ii). People commonly use contrastive explanations to explain their reasoning (Miller, 2018). Rather than asking "Why P?", they ask "Why P rather than Q?", where Q may be implicit from the context. For example, instead of justifying why raisins are the appropriate choice, people tend to ex-

plain why they are more likely than peanuts. Miller (2018) also argues that such contrastive explanations are computationally efficient, as they only require focusing on the limited set of reasons that might make one answer more likely than the other instead of exhaustively enumerating all possible reasons for an answer. For instance, the raisin's taste (not its size, temperature, etc.) in Table 1 is adequate to explain why it is the best answer.

Our goal is to enable PLMs that explain their predictions to similarly benefit from such constraints. We develop a small set of contrastive generation prompts that can be in-filled by a PLM such as T5 (Raffel et al., 2020) or BART (Lewis et al., 2020) (see Table 3). These templates are designed to cover a multitude of language patterns used by humans to compare and contrast entities. Another PLM then conditions on both the original input and the generated contrastive explanation, to predict the final answer. This approach is inspired by Shwartz et al. (2020), who also use textual prompts to query the PLM with clarification questions. However, their prompts are generic while we prompt for instance-specific information.

Our approach shows quantitative improvements in task performance over two existing methods for model explainability (Shwartz et al., 2020; Latcinnik and Berant, 2020), for two commonsense reasoning tasks: the Winograd Schema Challenge (Levesque et al., 2012) and multiple-choice question answering about physical commonsense (Bisk et al., 2020). Our gains in the zero-shot setting are especially notable, outperforming the best reported results on publicly available PLMs and improving over Shwartz et al. (2020) by up to 11%. We also show, through human evaluations, that contrastive explanations are deemed more useful for solving the original task compared to generic clarification questions. Finally, contrastive explanations can be semantically perturbed to quantify the model's dependence on them by flipping the contrast in the explanation to support the foil, facilitating quantification of model faithfulness.[1]

2 Related Work

Models that rationalize their decisions by extracting a contiguous subsequence of the input as an explanation (Lei et al., 2016; DeYoung et al., 2020; Paranjape et al., 2020) are inadequate in explaining commonsense reasoning tasks that require knowledge that is implicit in the input. Such tasks necessitate PLMs to rely on embedded parametric knowledge. Recent work use free-form textual explanations to generate explanations for commonsense reasoning tasks like SNLI (Camburu et al., 2018), Winograd Schemas (Zhang et al., 2020) and CommonsenseQA (Rajani et al., 2019b) through explicit human supervision, which are inherently ambiguous, incomplete and consequently, expensive to collect and evaluate on (Camburu et al., 2019b,a; DeYoung et al., 2020). Most recently, Latcinnik and Berant (2020) use an unsupervised approach to generate free-form explanations as sequences of tokens that are not well-formed sentences. In contrast, our method uses specialized prompts to generate well-formed human-interpretable explanations without any additional supervision.

Specialized prompts have been shown useful for extracting knowledge from PLMs in a targeted manner (Petroni et al., 2020; Richardson and Sabharwal, 2020; Talmor et al., 2020; Donahue et al., 2020; Lin et al., 2019) and improving performance on downstream tasks (Brown et al., 2020; Shin et al., 2020). Most relevant to our work is the self-talk model of Shwartz et al. (2020), an unsupervised approach using a fixed set of clarification questions as prompts to elicit knowledge from PLMs for commonsense reasoning tasks. Our work differs by focusing specifically on contrastive PLM prompts, which we find further improve performance by eliciting explanations which are highly relevant to the classification decision (Section 6).

Our approach to contrastive reasoning is also closely related to *counterfactuals*, which can be used to give contrastive explanations, i.e., answers to "Why P rather than Q?", by providing a counterfactual case in which Q would have held. Ross et al. (2020) use this idea to generate contrastive explanations, while it has also been used for evaluation (Gardner et al., 2020) and training (Kaushik et al., 2019) with the aim of addressing model robustness. Most of this work explicitly constructs counterfactual cases by perturbing the input data of a task in order to produce changes in the output label. In contrast, we do not construct counterfactual *inputs*, but aim to explicitly represent counterfactual *knowledge*: a contrast between the fact P and foil Q that, were it hypothetically *reversed*, would change the output label. We include an evaluation of our models on this question in Section 6.3.

[1] Code is available at https://github.com/bhargaviparanjape/RAG-X

Dataset Instance	Human-Authored Contrastive Explanation
Winograd Schema 1. The party was more interesting and uplifing than the funeral because the __ was rigid. 2. The geese prefer to nest in the fields rather than the forests because in the _ predators are more hidden.	○ Parties are for celebrating while funerals are for mourning ○ People wear colorful clothes at parties and black at funerals ○ Forests are dense while fields are sparse ○ Forests have more predators than fields.
PIQA 1. How do you get strong hamstrings? __ (a) work out your upper body (b) work out your legs 2. How do you flood a room? __ (a) fill it with objects (b) fill it with water	○ Hamstrings are located in the legs while biceps are located in the upper body ○ Filling it with objects can clutter a room while filling it with water floods the room.

Table 2: Examples of commonsense tasks that can be explained using contrastive language and some contrastive explanations authored by in-house annotators. The Fact and Foil are marked in the input.

3 Contrastive Explanations

We present the theory of contrastive explanations adopted in this work (Section 3.1) and the intuition behind using them for commonsense reasoning tasks (Section 3.2).

3.1 Definition and Motivation

A contrastive explanation is generally defined as an answer to a counterfactual question of the form "Why P rather than Q?" for two potential hypotheses P and Q that can follow from some event E. It explains why some *fact* P occurred instead of some *foil* Q, where Q can be implicit (Hesslow, 1988; Lipton, 1990; Miller, 2019). A good contrastive explanation points to differences between the fact and foil with regard to certain attributes, not just conveying that the fact has a certain attribute. Table 1 shows examples of contrastive explanations that differentiate between peanuts and raisins (on the basis of taste) or forests and fields (on the basis of vegetation densities) to explain the more probable answers to Winograd Schema instances.

Previous studies (Miller, 2019) in philosophy, psychology, and cognitive science show that humans use such contrastive explanations when explaining their decisions to each other. Importantly, Miller (2018) also argues that contrastive explanations are computationally efficient – exhaustively describing all causes for the occurrence of an event P is harder than only enlisting causes for why another event Q did not occur instead of P.

3.2 Contrastive Explanations for Commonsense Reasoning Tasks

Many recently proposed commonsense reasoning tasks are framed in a multiple-choice format that facilitates contrastive explanation (see Table 2). In this study, we focus on the following two tasks.

The **Winograd Schema Challenge** (Levesque et al., 2012, WSC) is a pronoun coreference resolution task designed as a hard benchmark for evaluating everyday knowledge and commonsense reasoning (Zhang et al., 2020). For instance, in the sentence "The city councilmen refused the demonstrators a permit because they feared violence," the pronoun *they* must be disambiguated between fact (*the city councilmen*) and foil (*the demonstrators*). Both fact and foil are explicit in such sentences.

The **Physical Interaction Question Answering** (Bisk et al., 2020, PIQA) challenge is designed to test knowledge of physical commonsense. PIQA requires choosing between which one of two *solutions* is a better way of achieving a *goal* posed as a question (see Table 2). PIQA questions relate to physical properties of entities, their affordances, and how they can be manipulated. The fact and foil are explicit in the two solutions, which typically differ from one another by a short noun phrase.

To validate our intuition that contrastive reasoning is instrumental in these tasks, we performed a pilot study with 10 annotators over 100 commonsense questions from Winogrande and PIQA. We instructed them to answer the questions and explain their reasoning, but gave no specific instructions about what the explanations should look like. Examples are shown in Table 2. In 76% of Winogrande and 64% of PIQA examples, annotators explicitly contrasted the fact and foil. The frequent use of certain phrase structures, like *P are __ while Q are __*, strongly informed our method for generating them (Section 4).

4 Our Approach

We assume the input to a commonsense reasoning problem consists of a textual context c which contains a placeholder _, and two marked answer

Prompt Pattern	Commonsense Example & Model Generated Explanation
Personal Characteristics $\Longrightarrow$ P likes/likes to _ while Q likes/likes to _ P likes/likes to _ while Q does not like/like to _ P prefers/prefers to _ while Q prefers _ Q prefers _ while P does not prefer/prefer to _ Q thinks _ while P thinks/does not think _	Megan said it would be liberating to go out without makeup like Elena does since __ never wore makeup `Explanation: Elena likes to be natural while Megan likes to wear lipstick`
Object Characteristics P is taller/shorter/smaller/larger/slower/faster than Q $\Longrightarrow$ P is/are _ while/but/however Q is/are _ Q has/have _ while/but/however P has/have _ P has/have more/less _ than Q P is/are _ than Q	How to tie pieces of paper together? __ (a) Thread ruler through the holes (b) Thread ribbon through the holes `Explanation: Ruler is hard while a ribbon is flexible`
Spatial/Temporal Contrast $\Longrightarrow$ P is inside/outside/above/below Q _ is closer to P and farther away from Q P is to the right/left of Q Q takes longer to _ than P	Emily looked up and saw Patricia racing by overhead. __ was on the ramp. `Explanation: Emily is below Patricia`
Use cases and causes P is used for _ Q P is used to do Q _ $\Longrightarrow$ P is used for/to/in _ while Q is used for/to/in _ Q is used _ while P is used _ Q because _ while P because _ Q can cause _ while P results in _	To prepare the puff pastry for your pie, line a baking sheet with parchment. Then __ (a) Unroll the pastry, lay it over baking twine. (b) Unroll the pastry, lay it over fishing line. `Explanation: Baking twine is used in baking while fishing line is used in fishing`

Table 3: Contrastive Patterns and Examples of outputs generated by the T5-large model. The pattern the PLM completes are marked $\Longrightarrow$.

choices a_1 and a_2 corresponding to the fact and foil (Table 2, left column). Let c_x denote substitution of x for the placeholder in c. The task is to predict whether c_{a_1} or c_{a_2} is more likely to be true, i.e., whether a_1 or a_2 best completes the context.

Our approach has two stages: First, an **Explainer PLM** P_{expln} generates contrastive explanations (Section 4.2) by infilling preset *contrastive templates* (Sec. 4.1) on the basis of c, a_1, and a_2. Then, a **Task Model** P_{LM} selects the correct answer conditioned on both the context and the generated explanations (Sec. 4.3).

4.1 Contrastive Templates

We develop a list of contrastive templates on the basis of an annotation study. For 250 instances from Winogrande and PIQA, we asked three annotators to explain why one answer is more likely than the other. We manually examined these explanations and abstracted them into templates containing at least two placeholders: two for the fact and foil being contrasted, and possibly more corresponding to the properties they are being contrasted on. For instance, *peanuts are salty while raisins are sweet* becomes *Q are _ while P are _*. We retained templates used by annotators at least 10 times. Table 3 shows several examples. A template is converted into an explanation by replacing placeholders for the fact and foil with answers a_1 and a_2 and the remaining placeholders with the appropriate contrastive information.

We evaluate the quality and coverage of our templates with another round of human evaluation. For 100 WSC and PIQA examples, we ask three annotators to either write contrastive explanations using one or more of the templates, or indicate that none of the them were appropriate. Annotators used the templates in over 82% of cases, indicating high coverage for the tasks we study.

4.2 Generating Explanations

Let t denote a contrastive template. We write t_{a_1,a_2} to denote the customization of t to an input by filling its marked placeholders for fact and foil with the answer choices. For instance, in Figure 1, the template *P are _ while Q are _* is customized to *Fields are _ while forests are _*.[2] A full explanation may be produced by filling the remaining gaps in t_{a_1,a_2} by leveraging an infilling language model, the explainer P_{expln}.

We first construct a neutral context c_{a_0} by filling c's placeholder with a task-specific neutral answer

[2]In practice, we randomize the order of a_1 and a_2 when customizing the template.

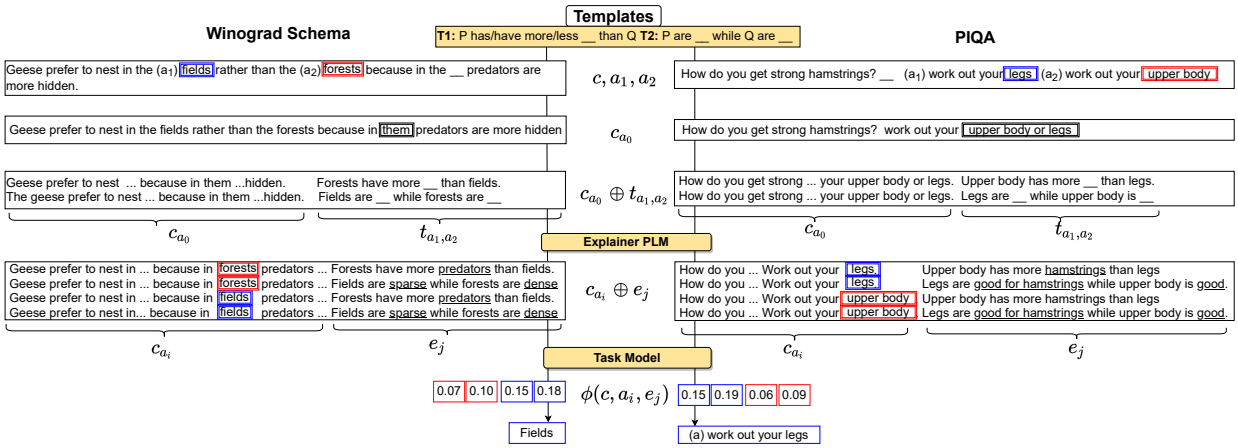

Figure 1: (1) A commonsense reasoning instance (c, a_1, a_2) is converted into a custom prompt $c_{a_0} \oplus t_{a_1,a_2}$ as input for the explainer PLM (2) The combination of input and explanation $(c_{a_i} \oplus e_j)$ is used by task model to score $a_i \forall i \forall j$. For a_1 and a_2, scores are aggregated over templates.

that does not indicate if a_1 or a_2 is correct. For Winogrande Schemas, c_{a_0} is constructed using the ambiguous pronoun in c (*them* in Figure 1). For PIQA, c_{a_0} is constructed as "$c \oplus a_1$ or a_2", where $\oplus$ is string concatenation, e.g., *upper body or legs* in Figure 1 (More dataset-specific details are in Section 5.2). We then prepend c_{a_0} to the customized template t_{a_1,a_2} and use it as input to the infilling language model to fill in the remaining gaps in the template. We use the maximum likelihood candidate phrases from top-K decoding to transform the template into a full explanation e.

We use a list of templates $t_1, \ldots, t_n$ to generate a list of candidate explanations $\mathsf{e}_1, \ldots, \mathsf{e}_n$ for each input, which are all fed into the task model. We also use some task-specific heuristics to reduce the number of prompts for each example, detailed in Appendix A.

4.3 Task Model

Given the context and answer choices (c, a_1, a_2) and a list of explanations $\mathsf{e}_1, \ldots, \mathsf{e}_n$, the second stage of our pipeline is a binary classifier between a_1 and a_2 which marginalizes over the explanations. We first assign a score to each answer $a \in \{a_1, a_2\}$ and explanation $\mathsf{e} \in \{\mathsf{e}_1, ..., \mathsf{e}_n\}$:

$$\phi(c, a, \mathsf{e}) = \frac{1}{k} \log \mathsf{P}_{\text{LM}}(c_a \oplus \mathsf{e}),$$

where c_a denotes the substitution of a into c, P_{LM} is string probability under the task language model, and k is the string length of $c_a \oplus \mathsf{e}$. We use ϕ as input to a logistic regression classifier which marginalizes over explanations:

$$\mathsf{P}(a \mid c, a_1, a_2) = \frac{\sum_i^n e^{\phi(c,a,\mathsf{e}_i)}}{Z},$$

where Z is a normalizer over a_1 and a_2. At initialization, ϕ uses a pretrained language model, and we fine-tune it to minimize the cross-entropy loss of $\mathsf{P}(a^* \mid c, a_1, a_2)$, where a^* is the correct answer. We do not fine-tune the explainer PLM since the top-K beam decoding is a discrete operation that is hard to backpropagate through. In the zero-shot setting (where the task PLM is not fine-tuned) and during inference, the answer is predicted by aggregating scores assigned to an answer by all n explanations: $\text{argmax}_{a_i} \sum_j \phi(c, a_i, \mathsf{e}_j)$.

5 Experimental Setup

5.1 Baselines

Context-Only We experiment with a baseline that does not condition on explanations at all. Here,

$$\phi(a, c) = \frac{1}{k} \log \mathsf{P}_{\text{LM}}(c_a),$$

and gold answer is $\text{argmax}_{a_i} \phi(a_i, c)$

Unconstrained Generation Latcinnik and Berant (2020) generate explanations from a PLM by beam-decoding a free-form sequence termed a *hypothesis* which is then used by a classifier to solve the task. The model is trained end-to-end and loss terms are added to encourage the hypothesis to sound natural. Explanation generation is otherwise unconstrained. For fair comparison with our approach, we do not fine-tune the explainer PLM (more details are in Appendix C).

Self-Talk Shwartz et al. (2020) propose an unsupervised model that uses a PLM as the answer scorer and a (possibly different) PLM as a knowledge source, similar to our framework. They formulate the process of obtaining relevant knowledge as *self-talk* with the following steps: 1) completing clarification question prefixes such as "what is the definition of ..." conditioned on input context, 2) generating their corresponding answers (clarifications), and 3) conditioning on the clarification questions and answers to make predictions. The key difference between their approach and ours is in the choice of prompts for the PLM, and the kinds of knowledge the prompts seek. While Shwartz et al. (2020) draw inspiration from inquiry-based discovery learning (Bruner, 1961), we target contrastive reasoning.

5.2 Implementation details

We use BART-Large (Lewis et al., 2020) and T5 (Raffel et al., 2020) as the explainer PLMs. Hyperparameters for infilling are given in Appendix C. For a fair comparison of all models, we use GPT2-XL (Radford et al., 2019) as the task model that estimates $\phi(c, a, \mathsf{e})$. GPT2-XL is the best performing PLM used by Shwartz et al. (2020) for WSC and PIQA tasks. Hyperparameter details about finetuning are given in Appendix C. We describe dataset specific modifications made to create c_{a_0}, c_{a_1}, and c_{a_2} in Section 4.2.

Winograd Schema Challenge (WSC) We experiment on (i) the SuperGLUE (Wang et al., 2019) version of the WSC consisting of 285 examples of anaphora (pronoun) resolution; (ii) Winogrande (WGRD) (Sakaguchi et al., 2020), a large scale crowdsourced version of the WSC; and (iii) WINOGENDER (WGND), a diagnostic dataset created to measure gender bias in models for ambiguous pronoun resolution (Rudinger et al., 2018).

Each instance provides two answer choices, which we use directly as a_1 and a_2. For the neutral answer c_{a_0}, we use the sentence with the original ambiguous pronoun. Since Winogrande has a blank space _ for the answer, we replace it with the most likely pronoun under a masked language model (BERT), following Shwartz et al. (2020). c_{a_1}, c_{a_2} are obtained by replacing the blank space or pronoun with the answer choice.

Physical Interaction Question Answering (PIQA) (Bisk et al., 2020) PIQA provides two answer choices which mostly vary from each other on a substring (e.g., "work out your [upper body]/[legs]"). We use these differing substrings as a_1=legs and a_2=upper body. For the neutral answer a_0, we combine the answers into "a_1 or a_2" (upper body or legs). In the cases where a_1 or a_2 is longer than 2 words, we include an *or* between the full answers. More details and examples are presented in Appendix A. We use question-answer pairs for c_{a_1} and c_{a_2}.

6 Experimental Results

In this section, we present an extensive evaluation of our approach, demonstrating performance gains which are independently verified by human judges.

6.1 Task Performance

We report task accuracy as a proxy for explanation quality. Table 4 compares the task performance of our model with the baselines defined in Section 5.1. We observe that generating and conditioning on additional information from PLMs improves performance over just using the original input (Row 1 vs. 2-6). Using templates to prompt the PLM for specific knowledge is better than unconstrained generation of text (Row 2 vs. 3-6). Contrastive explanations outperform previous work that use clarification questions in self-talk (Shwartz et al., 2020). The T5-Large explainer already surpasses the results of self-talk despite being smaller than GPT2-XL, demonstrating the impact of using contrastive explanations over clarification questions.

We also observe that larger explainer PLMs (going from T5-Large to T5-11B) yield higher performance. Our zero-shot results with T5-11B are the highest reported on Winogrande, PIQA and WSC for an open-sourced model.[3]

Finally, our approach gets smaller improvements when finetuning the task model. This suggests that some of the reasoning is still learned implicitly by the task model. Figure 2 shows task performance with various training data sizes of Winogrande, indicating a larger gap between the Context-Only baseline and our approach when training data is scarce.

6.2 Human Evaluation

Setup Following the human evaluation setup of Shwartz et al. (2020), we sample up to 50 highest-

[3]The zero-shot SOTA model (Brown et al., 2020) uses the 175B parameter GPT-3 model, which would likely also be a stronger explainer for our approach, but we did not have access to it.

	Explainer PLM (# Params)	Task model	WGRD ZS	WGRD FT	PIQA ZS	PIQA FT	WSC ZS	WGND ZS
1. Context-only	GPT2-XL (1.5B)	GPT2-XL	54.8	77.9	62.6	80.1	61.5	60.0
2. Unconstrained	GPT2-XL		54.9	77.8	63.9	80.7	61.4	60.0
3. Self-Talk	GPT2-XL		55.1	78.4	69.5	82.3	62.0	61.3
4. Contrastive	BART-Large(680M)		56.8	78.9	71.8	82.8	63.2	62.9
5. (Ours)	T5-Large (770M)		59.2	79.1	72.5	83.5	63.5	63.2
6.	T5-11B(11B)		**60.3**	**79.6**	**73.4**	**83.9**	**64.1**	**63.5**

Table 4: Test set accuracy on Winogrande (WGRD), PIQA, WSC and Winogender (WGND). ZS is Zero-shot models while FT is fine-tuned models. WSC and Winogender don't have training data for finetuning. Across all our models, the task model is GPT2-XL for fair comparison with (Shwartz et al., 2020) and to make finetuning tractable.

Metric	Self-Talk (Reported) WGRD	Self-Talk (Reported) PIQA	Self-Talk WGRD	Self-Talk PIQA	Contrastive WGRD	Contrastive PIQA
Relevant	68	60	70.4	61.7	73.1	70.7
Factual	46	42	40.8	38.8	43.0	39.4
Helpful	24	26	22.5	27.7	42.8	32.8
Grammatical	87.2	87.2	87.5	87.5	83.5	83.5
Flips	NA	NA	NA	NA	66.9	59.4

Table 5: Human Evaluation Results on Winogrande(WGRD) and PIQA. Reported human evaluation results on Self-talk are different from ours, which can be because of moderate levels of agreement (Fleiss Kappa κ = 0.43). Grammatiality is judged together for all datasets following (Shwartz et al., 2020). Only contrastive explanations can be flipped.

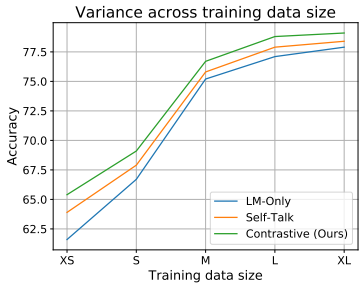

Figure 2: Performance variation in the finetuning setting across different sizes of Winogrande training data.

scoring explanations from PIQA and Winogrande examples which the T5-Large model got correct but the Context-Only baseline failed at. For comparison, we also include explanations from the self-talk model for evaluation.

Crowd workers are presented with a commonsense instance, the correct answer, and an explanation, and are asked to judge for: 1) *Grammaticality*, whether the explanation is grammatical; 2) *Relevance*, whether it's relevant to the topic of the text; 3) *Factual Correctness*, whether it's factually correct or likely true; and 4) *Helpfulness*, whether it adds helpful evidence for the correct answer. These metrics and definitions follow from Shwartz et al. (2020) with more details in Appendix B. The annotators are also shown the same explanation with fact and foil flipped (details in Section 6.3) and are asked to judge if the other answer is more likely than before if they assume the flipped explanation to be hypothetically true.

Results Table 5 shows the results of human evaluation of contrastive and self-talk explanations. The contrastive explanations are overwhelmingly preferred over self-talk explanations for relevance, factual correctness and helpfulness. They may be considered less grammatical because of in-filling noise (such as incomplete phrases). Table 6 presents some qualitative examples of instances where contrastive explanations improve over all baselines.

6.3 Analysis

We also analyze how much the task model relies on contrastive explanations for its decisions.

Flipping Explanations Our choice of contrastive language templates facilitates a novel way to evaluate explanation usefulness in prediction. The contrast in the explanation can be reversed by flipping the position of the fact and the foil in the explanation. If the choice between fact and foil actually depends on the contrastive explanation, then the flipped explanation should provide a hypothetical situation where the foil is more likely than the fact. For instance, "peanuts are salty while raisins are sweet," when switched to "raisins are sweet while peanuts are salty," may provide evidence that *peanuts* is a more likely label for the example in Table 1 (i). This may cause a model that uses the explanation to flip its prediction and lead to a drop in accuracy. The magnitude of drop can quantify the extent to which the model relies on the contrast

Example	Unconstrained	Self-Talk	Contrastive
(i) Ian volunteered to eat Dennis's menudo after already having a bowl because he despised eating intestine.	Dennis's menudo was disgusting.	What are the properties of a menudo? A menudo is made from the intestines of a pig	Dennis is a vegetarian while Ian is a carnivore. Dennis has menudo while Ian has volunteered to eat Denni's menudo.
(i) The GPS and map helped me navigate home. I got lost when the it got turned upside down.	because the GPS and map helped me navigate home.	What is going on here? The iphone app is not working.	The GPS can tell me where I am but the map can't. The GPS is right-side-up while the map is upside down
(ii) I helped my sister find her gold necklace. She couldn't wear her woven necklace to the ball because it was so casual.	She couldn't wear her woven necklace.	What are the properties of gold? The properties of gold are listed below.	Gold necklace is used for formal occasion while woven necklace is used for casual occasion.

Table 6: Qualitative Examples on Winogrande where contrastive explanations (using T5-11B explainer) improve task performance over baselines.

Explainer PLM	WGRD ZS	WGRD FT	PIQA ZS	PIQA FT	WSC ZS	WGND ZS
BART-Large	53.9 (5.4)	75.9 (4.0)	66.5 (7.9)	79.1 (4.6)	59.1 (6.9)	58.7 (7.1)
T5-Large	56.2 (5.3)	75.3 (5.0)	68.1 (6.5)	80.2 (4.2)	60.2 (5.5)	59.0 (7.1)
T5-11B	57.6 (4.5)	76.1 (4.7)	69.5 (5.4)	81.0 (3.6)	61.1 (3.3)	59.0 (5.8)

Table 7: Flipped evaluation results for contrastive explanation models. Reporting test accuracy across all datasets. Percent drop in performance as a result of flipping is indicated in parentheses.

Input	WGRD	PIQA
Fully abstracted	63.2	54.6
Abst. after expl.	70.4	64.9
No abstraction	79.1	83.5

Table 8: Evaluation of fine-tuned T5-Large contrastive models on Winogrande with abstracted answers.

provided in the explanation. In fact, humans also deem the flipped explanation to imply the opposite label in a majority of cases (Table 5), indicating that our contrastive explanations frequently capture contrastive properties that the labels truly rely on.

Table 7 shows the flipped evaluation results. We observe declines in accuracy of up to 8%, indicating that the model does use some contrastive knowledge to reason about the task. Finetuned models show a smaller decline in accuracy compared to the zero-shot setting. In this case, the task model may be directly fitting the data in lieu of relying on the knowledge conveyed by the explanation.

Abstracting Fact and Foil Given input context c (consisting of the fact and foil a_1, a_2) and an explanation e, the explainer PLM P_{expl} infills its explanation e on c while the task model P_{LM} conditions on both c and e. We can test the quality of the generated explanations and the task model's reliance on them by forcing the task model to rely on e when information in input c is restricted. One potential way to do so is to scrub the identities of the fact and foil, a_1 and a_2, from c.

We replace the fact and foil with placeholder tokens to create an abstract context c'. For instance, the example in Table 6 (ii) becomes "The <mask1> and <mask2> helped me navigate ... down.", where the model must now choose between <mask1> and <mask2>.[4] Running the task model on c' lower-bounds the performance possible without knowing answer identities. We can now test the relevant answer-based knowledge contrastive contained in the *explanations* by allowing the explanation model to see the original answers in c, but then abstracting them out when passing the input context and explanations to the task model. More formally, the task model conditions its decision on c' and e'. For Table 6 (ii) c' and e' are "The <mask1> and <mask2> helped me navigate ... down." and "The <mask1> is right-side-up while the <mask2> is upside down." Since only the explainer PLM is shown answer identities, the task model's decision is conditionally independent of the answer identities given the explanation.

Experiments on Winogrande and PIQA in the

[4]More examples of abstracted contexts and explanations are given in the Appendix (Table 11).

Model	Acc.
Random	20.0
Baseline	37.2
Self talk	26.9
Contrastive (V)	38.1
Contrastive (MM)	37.4
Banerjee and Baral (2020a)	38.8

Table 9: Zero-shot test performance on CommonsenseQA for baselines as well as contrastive models which ensemble fact/foil pairs by voting (V) and maximum margin (MM). The best reported unsupervised performance (Banerjee and Baral, 2020b) uses ConceptNet, which was used to construct the dataset.

fine-tuned setting (Table 8) show that performance improves significantly when the task model conditions on both c' and e' compared to a fully abstracted contrastive baseline that only conditions on c' (from 63.2 to 70.4 for Winogrande), covering almost half of the gap between the fully abstracted setting and the non-abstracted original model (79.1). This indicates that our contrastive explanations encode a significant amount of information required for commonsense tasks. Even if the full model does not always use the explanations, these evaluations show that our contrastive explanations contain rich task-relevant knowledge, and suggest that future work might focus on how to better make use of this signal.

6.4 Generalizability of Prompts

The set of contrastive prompts used in our framework are curated from an in-house analysis of training instances from Winogrande and PIQA datasets. To determine the generalizability of these prompts for other commonsense reasoning tasks, we also experiment with the CommonsenseQA dataset (Talmor et al., 2019), which consists of multiple-choice questions created over ConceptNet – "Where on a river can you hold a cup upright to catch water on a sunny day? a) waterfall, b) bridge, c) valley, d) pebble, e) mountain". Since there are more than two answer choices to contrast, we convert each instance into 10 pairwise (binary) classification instances. Contrastive explanations are generated for each pairwise decision in the zero-shot setting, similar to Winograd and PIQA datasets. To choose the final answer, we consider two inference procedures: (a) *Vote:* The answer that receives the maximum number of votes across all binary classification instances is selected, and (b) *Maximum Margin:* The choice with the maximum difference (margin) between answer likelihoods for any binary classification instance is selected. In Table 9, we observe that self-talk significantly hurts performance for this dataset. On the other hand, contrastive explanations are found to be useful and approach the zero-shot performance of the state-of-the-art, which uses ConceptNet (Banerjee and Baral, 2020b). These results demonstrate that the set of contrastive prompts are generalizable to other commonsense reasoning datasets, and that while our contrastive prompts are limited to contrasting two answer choices at a time, the framework can be easily extended to tasks with multiple foils.

7 Conclusion

We show it is possible to prompt pretrained language models (PLMs) to generate contrastive explanations of their reasoning patterns, inspired by explanations that humans naturally provide for their reasoning. Conditioning model decisions on these explanations improves performance on two commonsense reasoning benchmarks, and humans judge the explanations to be highly relevant and helpful in comparison to prior work. We also showed how contrastive explanations can facilitate in-depth evaluations of faithfulness by flipping or abstracting the fact and foil, finding that our explanations encode a significant amount of information relevant to the classification decision, and in many cases models rely on the contrast in the expected way. While we have shown that our method is flexible enough to apply to multiple-choice commonsense tasks with many foils, leveraging contrastive reasoning in a wider variety of open-ended tasks remains an exciting challenge for future work.

Acknowledgements

This research was supported by ONR N00014-18-1- 2826, DARPA N66001-19-2-403, ARO W911NF16-1-0121 and NSF IIS-1252835, IIS-1562364, an Allen Distinguished Investigator Award, and the Sloan Fellowship. We thank Vered Shwartz, Mandar Joshi, Divyansh Kaushik, H2Lab members, UW NLP and the anonymous reviewers for their helpful comments and suggestions.

References

Pratyay Banerjee and Chitta Baral. 2020a. Self-supervised knowledge triplet learning for zero-shot question answering. *arXiv preprint arXiv:2005.00316*.

Pratyay Banerjee and Chitta Baral. 2020b. Self-supervised knowledge triplet learning for zero-shot question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 151–162, Online. Association for Computational Linguistics.

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

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Jerome S Bruner. 1961. The act of discovery. *Harvard educational review*.

Oana-Maria Camburu, Eleonora Giunchiglia, Jakob Foerster, Thomas Lukasiewicz, and Phil Blunsom. 2019a. Can i trust the explainer? verifying post-hoc explanatory methods. *arXiv preprint arXiv:1910.02065*.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *Advances in Neural Information Processing Systems*, pages 9539–9549.

Oana-Maria Camburu, Brendan Shillingford, Pasquale Minervini, Thomas Lukasiewicz, and Phil Blunsom. 2019b. Make up your mind! adversarial generation of inconsistent natural language explanations. *arXiv preprint arXiv:1910.03065*.

Oana-Maria Camburu, Brendan Shillingford, Pasquale Minervini, Thomas Lukasiewicz, and Phil Blunsom. 2020. Make up your mind! adversarial generation of inconsistent natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4157–4165, Online. Association for Computational Linguistics.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online. Association for Computational Linguistics.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. *arXiv preprint arXiv:2005.05339*.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, et al. 2020. Evaluating models' local decision boundaries via contrast sets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1307–1323.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. *arXiv preprint arXiv:1908.07898*.

Germund Hesslow. 1988. The problem of causal selection. *Contemporary science and natural explanation: Commonsense conceptions of causality*, pages 11–32.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. 2019. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Veronica Latcinnik and Jonathan Berant. 2020. Explaining question answering models through text generation. *arXiv preprint arXiv:2004.05569*.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*. Citeseer.

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 the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2019. Commongen: A constrained text generation challenge for generative commonsense reasoning. *arXiv preprint arXiv:1911.03705*.

Peter Lipton. 1990. Contrastive explanation. *Royal Institute of Philosophy Supplements*, 27:247–266.

Tim Miller. 2018. Contrastive explanation: A structural-model approach. *arXiv preprint arXiv:1811.03163*.

Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38.

Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. 2020. Wt5?! training text-to-text models to explain their predictions. *arXiv preprint arXiv:2004.14546*.

Bhargavi Paranjape, Mandar Joshi, John Thickstun, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. An information bottleneck approach for controlling conciseness in rationale extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1938–1952, Online. Association for Computational Linguistics.

Fabio Petroni, Patrick Lewis, Aleksandra Piktus, Tim Rocktäschel, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2020. How context affects language models' factual predictions. *arXiv preprint arXiv:2005.04611*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. *NAACL HLT 2018*, page 180.

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

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

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019a. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019b. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy. Association for Computational Linguistics.

Kyle Richardson and Ashish Sabharwal. 2020. What does my qa model know? devising controlled probes using expert knowledge. *Transactions of the Association for Computational Linguistics*, 8:572–588.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*.

Alexis Ross, Ana Marasović, and Matthew E Peters. 2020. Explaining nlp models via minimal contrastive editing (mice). *arXiv preprint arXiv:2012.13985*.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 8–14.

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

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Unsupervised commonsense question answering with self-talk. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4615–4629.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. olmpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Trieu H Trinh and Quoc V Le. 2018. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *arXiv preprint 1905.00537*.

Sarah Wiegreffe, Ana Marasovic, and Noah A Smith. 2020. Measuring association between labels and free-text rationales. *arXiv preprint arXiv:2010.12762*.

Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. 2020. G-daug: Generative data augmentation for commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1008–1025.

Hongming Zhang, Xinran Zhao, and Yangqiu Song. 2020. Winowhy: A deep diagnosis of essential commonsense knowledge for answering winograd schema challenge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5736–5745.

A Generating Contrastive Templates

Table 12 shows the complete list of contrastive patterns used in our work, categorized under different types of attributes/properties. For templates with no place holders for the explainer to fill out, we only replace placeholders for answers (fact and foil). Table 10 lists $a_0, a_1, a_2, c_{a_0}, c_{a_1}, c_{a_2}$ for different examples in Winogrande and PIQA to explain dataset specific transformations made by our approach.

Detection of P, Q: For WSC, the fact and foil are typically 1-word nouns. However, they may by qualified in the context and these qualifiers are important for contrasting. For instance, in the WSC example "She remembered how annoying it is to dust her wood chair so she bought a plastic table instead.", chair and table are the fact and foil. Their qualifiers wood and plastic are important for the construction of the contrast. Hence we retain these qualifiers when preparing prompts for the explainer PLM. Similarly, for PIQA, qualifiers are retained in the prompts.

Case filtering: We detect case of entities and accordingly filter out templates that are ungrammatical depending on whether the fact and foil are singular/plural.

Template filtering for WSC: For examples that do not contain PERSON named entities, we do not include prompts about personal characteristics. Similarly, for examples that contain PERSON named entities, Temporal, Usecase and some spatial patterns were left out.

Template filtering for PIQA: We remove all templates about personal characteristics as this dataset deals with physical commonsense.

B Human Evaluation

The annotation task was carried out in Amazon Mechanical Turk, following (Shwartz et al., 2020). To ensure the quality of annotations, workers were required to be located in the US, UK, or Canada, and have a 99% approval rate for at least 5000 prior tasks. Annotators were paid .30$ per HIT to ensure participants get approximately $15/hr if they are doing the task. Annotation were aggregated from 3 workers using majority vote. The annotations yielded moderate levels of agreement, with Fleiss Kappa κ = 0.43 (Landis and Koch, 1977).

C Hyperparameters

Explainer PLM For T5 we use special symbols `<extra_id_0>` and `<extra_id_1>` in place of the blanks (_) in our templates. We observe that T5 is able to replace these tokens with multi-word phrases. For BART, we substitute blanks with a sequence with four `[MASK]` tokens to encourage generating multiple words. BART can choose to delete a `[MASK]` token during generation. Top-K decoding was done with a beam size of 200 and a maximum output sequence length of 20 for T5 models and 100 for BART. This is because both T5 is pre-trained to in-fill by only generating missing phrases while BART is pre-trained to decode the entire input with missing phrases filled in. We used early stopping for BART.

Task PLM Task PLM was finetuned for 20 epochs, using BertAdam optimizer with a learning rate of $2e-5$, batch size of 8, and dropout of 0.1, following (Latcinnik and Berant, 2020).

Self-Talk (Shwartz et al., 2020) generate multiple clarification questions conditioned on the context, by 1) concatenating one of several question prefixes to the input prompt or question; and 2) generating 5 questions for each prefix using Nucleus sampling with $p = 0.2$, i.e., sampling from the top 20% tokens(Holtzman et al., 2019) limiting the question length to up to 6 tokens excluding the prefix. For each well-formed question, they generate multiple answers using a similar method. They

Winogrande
Ian volunteered to eat Dennis's menudo after already having a bowl because __ despised eating a_1 : Ian a_2 : Dennis a_0 : he c_{a_0} : Ian volunteered to eat Dennis's menudo after already having a bowl because he despised eating c_{a_1} : Ian volunteered to eat Dennis's menudo after already having a bowl because Ian despised eating c_{a_2} : Ian volunteered to eat Dennis's menudo after already having a bowl because Dennis despised eating
PIQA (difference between answers is 1-2 words)
To prepare carrots before cooking with them, you can a_1 : Run them in the sink under boiling water a_2 : Run them in the sink under cold water a_0 : boiling water or cold water c_{a_0} : To prepare carrots before cooking with them, you can run them in the sink under boiling water or cold water c_{a_1} : To prepare carrots before cooking with them, you can run them in the sink under boiling water c_{a_2} : To prepare carrots before cooking with them, you can run them in the sink under cold water
PIQA (difference between answers is larger)
To prevent gunk buildup in cup holders of a car, a_1 : place coffee filters inside of the cup holders. a_2 : pour a thin layer of oil into the cup holders. a_0 : place coffee filters inside of the cup holders or pour a thin layer of oil into the cup holders. c_{a_0} : To prevent gunk buildup in cup holders of a car, place coffee filters inside of the cup holders or pour a thin layer of oil into the cup holders c_{a_1} : To prevent gunk buildup in cup holders of a car, place coffee filters inside of the cup holders c_{a_2} : To prevent gunk buildup in cup holders of a car, pour a thin layer of oil into the cup holders

Table 10: Examples of Winogrande and PIQA, with fact, foil , neutral answer and respective substituted contexts used in our approach for prompting the explainer PLM or computing answer likelihood.

Original Input: The geese prefer to nest in the fields rather than the forests because in the _ predators are more hidden.
(i) Context-Only Input to task model: The geese prefer to nest in the <mask1> rather than the <mask2> because in the _ predators are more hidden.
(ii) Fully Abstracted Input to explainer: The geese prefer to nest in the <mask1> rather than the <mask2> because in the _ predators are more hidden. Generated Explanation: <mask1> is smaller than <mask2> Input to task model: The geese prefer to nest in the <mask1> rather than the <mask2> because in the _ predators are more hidden. <mask1> is smaller than <mask2>
(iii) Abstraction after Explanation Input to explainer: The geese prefer to nest in the fields rather than the forests because in the _ predators are more hidden. Generated Explanation: Forests have more predators than fields Input to task model: The geese prefer to nest in the <mask1> rather than the <mask2> because in the _ predators are more hidden. <mask2> have more predators than <mask1>

Table 11: Input to Explainer and Task model for Abstractive Evaluation

limit the answer length to 10 generated tokens, and use Nucleus sampling with $p = 0.5$. Shwartz et al. (2020) only condition task prediction on a single clarification question and answer pair that increases the model's belief of a certain answer choice. Thus, the score of each answer choice is selected as the score of the text containing the clarification that most supports it, i.e., whose combination with it yields maximum language model likelihood.

Unconstrained Generation For unconstrained explanation baseline, maximum output sequence length was set to 20 and beam size for beam decoding was set to 200. Again we use early stopping.

Complete list of Contrastive Prompt Templates	Commonsense Task/Instance Type
Temporal: OPT1 happened before/after OPT2 OPT1 takes longer than OPT2 OPT1 takes longer to _ than OPT2 OPT1 happened for a longer time than OPT2	PIQA (Consists of events)
Personal Characteristics: OPT1 likes _ while OPT2 likes _ OPT1 likes _ while OPT2 does not like _ OPT1 likes to _ while OPT2 likes to _ OPT1 likes to _ while OPT2 does not like to _ OPT1 prefers _ while OPT2 prefers _ OPT1 prefers _ while OPT2 does not prefer _ OPT1 prefers to _ while OPT2 prefers to _ OPT1 prefers to _ while OPT2 does not prefer to _ OPT1 thinks _ while OPT2 thinks _ OPT1 thinks _ while OPT2 does not thinks _	WSC (if PERSON entity tag is detected)
Object Characteristic: OPT1 is/are smaller than OPT2 OPT1 is/are larger than OPT2 OPT1 is/are slower than OPT2 OPT1 is/are faster than OPT2 OPT1 is _ than OPT2 OPT1 are _ than OPT2 OPT1 is _ while OPT2 is _ OPT1 is _ but OPT2 is _ OPT1 is _ however OPT2 is _ OPT1 are _ while OPT2 are _ OPT1 are _ but OPT2 are _ OPT1 are _ however OPT2 are _ OPT1 has _ while/but/however OPT2 has/does not have _ OPT1 have _ while/but/however OPT2 have/do not have _ OPT1 is made of/to _ however OPT2 is made of/to _ OPT1 is made of/to _ while OPT2 is made of/to _	WSC and PIQA
Spatial: OPT1 is above OPT2 OPT1 is below OPT2 OPT1 is to the right of OPT2 OPT1 is to the left of OPT2 OPT1 is inside OPT2 OPT1 is outside OPT2 _ is closer to OPT1 and father away from OPT2 OPT1 is closer to _ while OPT2 is father away from _	WSC and PIQA
Usecase: OPT1 can _ while OPT2 can/cannot _ OPT1 is/can be used for OPT2 OPT1 is/can be used to do OPT2 OPT1 is/can be used for _ but OPT2 cannot OPT1 is/can be used for _ while OPT2 is used for _ OPT1 is/can be s used for _ but OPT2 is used for _ OPT1 is/can be used to _ while OPT2 is used to _ OPT1 is/can be used to _ but OPT2 is used to _	WSC(No PERSON entity) and PIQA
Causes: OPT1 has _ because _ while OPT2 is _ because _ OPT1 can cause _ while OPT2 causes/results in _ Since _ it can OPT1 but not OPT2 Since _ it can OPT1 but because it is not _ it can't OPT2	WSC (No PERSON entity) and PIQA
Miscellaneous: _ can be OPT1 but cannot be OPT2 OPT1 means to _ while OPT2 means to _ OPT1 is defined as _ while OPT2 is defined as _ _ OPT1 _ OPT2 _ OPT1 but not OPT2 OPT1 exists while an OPT2 doesn't	WSC (No PERSON entity) and PIQA

Table 12: Complete list of contrastive patterns used in this work.