Promoting Contrastive Explanations for Commonsense Reasoning Tasks

Bhargavi Paranjape†∗ Julian Michael† Marjan Ghazvininejad∗
Luke Zettlemoyer†∗ Hannaneh Hajishirzi†
† Allen School of Computer Science & Engineering, University of Washington, Seattle, WA
∗ Allen Institute of Artificial Intelligence, Seattle
ϵ Facebook AI
{bparan,mandar90,thickstn,hannaneh,lsz}@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 pre-trained 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 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.

1 Introduction

Statistical significance, broader applicability Pre-trained 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 used, 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 (Wiegrefe et al., 2020; Camburu et al., 2020; Latcinnik and Berant, 2020). We address these challenges by proposing an unsupervised method that uses prompts which require the model to explicitly contrast different possible answers in its explanation (Table 1).

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.

<table>
<thead>
<tr>
<th>Instance</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
<td></td>
</tr>
<tr>
<td>Contrastive Expl. - Peanuts are salty while raisins tend to be sweet.</td>
<td></td>
</tr>
<tr>
<td>ii) The geese prefer to nest in the fields rather than the forests because in the __ predators are more hidden.</td>
<td>Contrastive Expl. - Forests are denser than fields</td>
</tr>
</tbody>
</table>

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 are 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 appropriate, people tend to explain why they are more likely than peanuts. Miller (2018) also argues that such contrastive explanations only
focus 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.) is adequate to explain why it is the best answer.

Our goal is to enable PLM explanation models 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 (Shwartz et al., 2020; Latcinnik and Berant, 2020) for two commonsense reasoning tasks – Winograd Schema Challenge (Levesque et al., 2012) and multiple-choice question answering about physical commonsense (Sap et al., 2019). 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. Finally, contrastive explanations can be perturbed to quantify the model’s dependence on them.

2 Related Work

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 generated well formed explanations without additional supervision.

The use of specialized prompts has shown to be useful in extracting knowledge from PLMs in a more targeted manner (Petroni et al., 2020; Richardson and Sabharwal, 2020; Talmor et al., 2020; Donahue et al., 2020; Lin et al., 2019) and in improving prediction on downstream tasks (Brown et al., 2020; Shin et al., 2020). Most relevant to our work is the self-talk model (Shwartz et al., 2020), an unsupervised framework that uses a fixed set of clarification questions as prompts to elicit additional knowledge from PLMs for commonsense reasoning tasks. Our work differs by focusing specifically on contrastive PLM prompts, which we find can 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).

3 Contrastive Explanations

In this section, we present the theory on contrastive explanations adopted in this work (Section 3.1) and motivate the intuition in 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
<table>
<thead>
<tr>
<th>Dataset Instance</th>
<th>Human-Authoried Contrastive Explanation</th>
</tr>
</thead>
</table>
| **Winograd Schema** | ○ 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. |
| 1. The party was more interesting and uplifting than the funeral because the _ was rigid. | ○ The party was more interesting and uplifting 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. | ○ The geese prefer to nest in the fields rather than the forests because in the _ predators are more hidden. |
| **PIQA** | ○ 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. |
| 1. How do you get strong hamstrings? _  
(a) work out your upper body  
(b) work out your legs | ○ 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. |
| 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.

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 Winograd 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 Winograd 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 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 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 selects the correct answer conditioned on both the context and the generated explanations (Sec. 4.3).
Table 3: Contrastive Patterns and Examples of outputs generated by the T5-large model. The pattern the PLM completes are marked $\Rightarrow$.

<table>
<thead>
<tr>
<th>Prompt Pattern</th>
<th>Commonsense Example &amp; Model Generated Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Personal Characteristics</strong></td>
<td></td>
</tr>
<tr>
<td>$\Rightarrow P \text{ likes/likes to } _1 \text{ while } Q \text{ likes/likes to } _2$</td>
<td>Megan said it would be liberating to go out without makeup like Elena does since _1 never wore makeup</td>
</tr>
<tr>
<td>$P \text{ likes/likes to } _1 \text{ while } Q \text{ does not like/to like } _2$</td>
<td>Explanation: Elena likes to be natural while Megan likes to wear lipstick</td>
</tr>
<tr>
<td>$P \text{ prefers/prefers to } _1 \text{ while } Q \text{ prefers } _2$</td>
<td></td>
</tr>
<tr>
<td>$Q \text{ prefers } _1 \text{ while } P \text{ does not prefer/prefers to } _2$</td>
<td></td>
</tr>
<tr>
<td>$Q \text{ thinks } _1 \text{ while } P \text{ thinks/does not think } _2$</td>
<td></td>
</tr>
<tr>
<td><strong>Object Characteristics</strong></td>
<td>How to tie pieces of paper together? _1</td>
</tr>
<tr>
<td>$P \text{ is taller/shorter/smaller/larger/slower/faster than } Q$</td>
<td>(a) Thread ruler through the holes</td>
</tr>
<tr>
<td>$\Rightarrow P \text{ is/are } _1 \text{ while/but/however } Q \text{ is/are } _2$</td>
<td>(b) Thread ribbon through the holes</td>
</tr>
<tr>
<td>$Q \text{ has/hae } _1 \text{ while/but/however } P \text{ has/hae } _2$</td>
<td>Explanation: Ruler is hard while a ribbon is flexible</td>
</tr>
<tr>
<td>$P \text{ has/have more/less than } Q$</td>
<td></td>
</tr>
<tr>
<td>$P \text{ is } _1 \text{ than } Q$</td>
<td></td>
</tr>
<tr>
<td><strong>Spatial/Temporal Contrast</strong></td>
<td>Emily looked up and saw Patricia racing by overhead. _1 was on the ramp.</td>
</tr>
<tr>
<td>$\Rightarrow P \text{ is inside/outside/above/below } Q$</td>
<td>Explanation: Emily is below Patricia</td>
</tr>
<tr>
<td>$Q \text{ is closer to } P \text{ and farther away from } Q$</td>
<td></td>
</tr>
<tr>
<td>$P \text{ takes longer to } _1 \text{ than } P$</td>
<td></td>
</tr>
<tr>
<td><strong>Use cases and causes</strong></td>
<td>To prepare the puff pastry for your pie, line a baking sheet with parchment. Then _1</td>
</tr>
<tr>
<td>$P \text{ is used for } _1 Q$</td>
<td>(a) Unroll the pastry, lay it over baking twine.</td>
</tr>
<tr>
<td>$P \text{ is used to do } _1 Q$</td>
<td>(b) Unroll the pastry, lay it over fishing line.</td>
</tr>
<tr>
<td>$\Rightarrow P \text{ is used for/to/in } _1 Q \text{ while } Q \text{ is used for/to/in } _2$</td>
<td>Explanation: Baking twine is used in baking while fishing line is used in fishing</td>
</tr>
<tr>
<td>$Q \text{ because } _1 P \text{ because } _2$</td>
<td></td>
</tr>
<tr>
<td>$Q \text{ can cause } _1 P \text{ results in } _2$</td>
<td></td>
</tr>
</tbody>
</table>

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 $Q \text{ are } _1 \text{ while } P \text{ are } _2$. 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 $P \text{ are } _1 \text{ while } Q \text{ are } _2$ is customized to $Fields \text{ are } _1 \text{ while forests are } _2$. A full explanation may be produced by filling the remaining gaps in $t_{a_1,a_2}$ by leveraging an infilling language model.

We first construct a neutral context $c_{a_0}$ by filling $c$’s placeholder with a task-specific neutral answer that does not indicate if $a_1$ or $a_2$ is correct. For Winogrande Schema, $c_{a_0}$ 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 take the maximum likelihood candidate phrases from top-K decoding and use them to fill in the gaps and transform the template into a full explanation $e$.

In practice, we randomize the order of $a_1$ and $a_2$ when customizing the template.
4.3 Task Model

Given the context and answer choices \((c, a_1, a_2)\) and a list of explanations \(e_1, \ldots, e_n\), 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 \(e \in \{e_1, \ldots, e_n\}\):

\[
\phi(c, a, e) = \frac{1}{k} \log P_{\text{LM}}(c_a \oplus e),
\]

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

\[
P(a \mid c, a_1, a_2) = \frac{\sum^n_{i=1} e^{\phi(c, a, e_i)}}{Z},
\]

where \(Z\) is a normalizer over \(a_1\) and \(a_2\). At initialization, \(\phi\) uses a pretrained language model, and we fine-tune it to minimize the cross-entropy loss of \(P(a^* \mid c, a_1, a_2)\), 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: \(\arg\max_{a_1} \sum_j \phi(c, a_1, e_j)\).

5 Experimental Setup

5.1 Baselines

Context-Only We experiment with a baseline that does not condition on explanations at all. Here, \(\phi(a, c) = \frac{1}{k} \log P_{\text{LM}}(c_a)\), and gold answer is \(\arg\max_{a_1} \phi(a_1, c)\).

Unconstrained Generation Latemnik and Berant (2020) generate explanations from a PLM by simply beam-decoding a free-form sequence termed a hypothesis, that 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 in Appendix A.3).

Self-Talk (Shwartz et al., 2020) is 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) incorporating 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 A.3. For a fair comparison of all models, we use GPT2-XL (Radford et al., 2019) as the task model that estimates $\phi(c, a, e)$. GPT2-XL is the best performing PLM in Shwartz et al. (2020) for WSC and PIQA tasks. Hyperparameter details about finetuning are given in Appendix A.3. We describe dataset specific modifications made to create $c_{a_0}, c_{a_1}, c_{a_2}$ 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) Winograd (WGRD) (Sakaguchi et al., 2020), a large scale crowdsourced version of the WSC; and (iii) WINGENDER (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 Winograd 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, with accompanying examples are presented in Appendix A.1. 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. 2

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

6.2 Human Evaluation
Setup Following the human evaluation setup in self-talk (Shwartz et al., 2020), we sample up to 50 highest-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.

Crowdworkers 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 A.2. The annotators are also shown the same explanation with

---

2The zero-shot SOTA model (Brown et al., 2020) uses the 175B parameter GPT3, which would likely also be a stronger explainer for our approach but we did not have access to it.
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.

<table>
<thead>
<tr>
<th>Explainer PLM (# Params)</th>
<th>Task model</th>
<th>WGRD ZS</th>
<th>PIQA ZS</th>
<th>WSC ZS</th>
<th>WGND ZS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Context-only GPT2-XL (1.5B)</td>
<td>GPT2-XL</td>
<td>54.8</td>
<td>77.9</td>
<td>62.6</td>
<td>80.1</td>
</tr>
<tr>
<td>2. Unconstrained GPT2-XL</td>
<td>GPT2-XL</td>
<td>54.9</td>
<td>77.8</td>
<td>63.9</td>
<td>80.7</td>
</tr>
<tr>
<td>3. Self-Talk GPT2-XL</td>
<td>GPT2-XL</td>
<td>55.1</td>
<td>78.4</td>
<td>69.5</td>
<td>82.3</td>
</tr>
<tr>
<td>4. Contrastive BART-Large(680M)</td>
<td></td>
<td>56.8</td>
<td>78.9</td>
<td>71.8</td>
<td>82.8</td>
</tr>
<tr>
<td>5. (Ours) T5-Large (770M)</td>
<td>T5-Large (770M)</td>
<td>59.2</td>
<td>79.1</td>
<td>72.5</td>
<td>83.5</td>
</tr>
<tr>
<td>6. T5-11B(11B)</td>
<td></td>
<td>61.3</td>
<td>79.6</td>
<td>74.4</td>
<td>83.9</td>
</tr>
</tbody>
</table>

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 $\kappa = 0.43$). Grammaticality is judged together for all datasets following (Shwartz et al., 2020). Only contrastive explanations can be flipped.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Self-Talk(Reported) WGRD</th>
<th>Self-Talk PIQA</th>
<th>Contrastive WGRD</th>
<th>Contrastive PIQA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relevant</td>
<td>68</td>
<td>60</td>
<td>70.4</td>
<td>61.7</td>
</tr>
<tr>
<td>Factual</td>
<td>46</td>
<td>42</td>
<td>40.8</td>
<td>38.8</td>
</tr>
<tr>
<td>Helpful</td>
<td>24</td>
<td>26</td>
<td>22.5</td>
<td>27.7</td>
</tr>
<tr>
<td>Grammatical</td>
<td>87.2</td>
<td>87.2</td>
<td>87.5</td>
<td>87.5</td>
</tr>
<tr>
<td>Flipping helps</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>

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 provided in the explanation.

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. In this case, the task model may be directly fitting the data in lieu of relying on the knowledge conveyed by the explanation. We observe that larger models have a smaller decline in performance (Row 2 vs 3), indicating that these models may be better at ignoring sequences that they are less likely to produce.
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.
(ii) 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.
(iii) 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.

<table>
<thead>
<tr>
<th>Explainer</th>
<th>WGRD</th>
<th>PIQA</th>
<th>WSC</th>
<th>WGND</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLM</td>
<td>ZS</td>
<td>FT</td>
<td>ZS</td>
<td>FT</td>
</tr>
<tr>
<td>BART-Large</td>
<td>52.9(5.4)</td>
<td>75.9(4.0)</td>
<td>66.5(7.9)</td>
<td>79.1(4.6)</td>
</tr>
<tr>
<td>T5-Large</td>
<td>56.2(5.3)</td>
<td>75.3(5.0)</td>
<td>68.1(6.5)</td>
<td>80.2(4.2)</td>
</tr>
<tr>
<td>T5-11B</td>
<td>58.6(4.5)</td>
<td>76.1(4.7)</td>
<td>70.5(5.4)</td>
<td>81.0(3.6)</td>
</tr>
</tbody>
</table>

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

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

<table>
<thead>
<tr>
<th>Input</th>
<th>WGRD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fully abstracted</td>
<td>63.2</td>
</tr>
<tr>
<td>Abst. after expl.</td>
<td>70.4</td>
</tr>
<tr>
<td>No abstraction</td>
<td>79.1</td>
</tr>
</tbody>
</table>

Abstracting Fact and Foil  We can also test the quality of the generated explanations by forcing the task model to rely on them for information about the identities of the answers $a_1$ and $a_2$. (for example, peanuts being salty and raisins being sweet). To do so, we use abstracted contexts where the fact and foil are replaced with placeholder tokens. For instance, the example in Table 6 (i) becomes “$<$mask1$>$ volunteered to eat $<$mask2$>$’s menudo after ...”, where the model must choose between $<$mask1$>$ and $<$mask2$>$.

Running our contrastive explanation model on abstracted contexts lower-bounds the performance possible without knowing answer identities. We can then test the relevant answer-based knowledge contained in the explanations by allowing the explanation model to see the original answers, but abstracting them out when passing the context and explanations to the task model. This forces the task model’s decision to be conditionally independent of the answer identities given the explanation.

Experiments on Winogrande (Table 8) show that giving the answer identities only to the explainer significantly improves performance from a fully abstracted contrastive baseline (from 63.2 to 70.4), covering almost half of the gap between the fully abstracted and non-abstracted 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 future work might focus on how to better make use of this signal.

7 Conclusion

We show it is possible to prompt pretrained language models (PLMs) to generate contrastive explanations of their reasoning patterns, inspired by explanations people naturally provide for their reasoning. We use PLMs to populate contrastive explanation templates which contrast answer alternatives according to the key attribute(s) required to justify the correct answer. Conditioning model decisions on these explanations improves performance on two commonsense reasoning benchmarks. Flipping the explanation degrades performance to some extent, quantifying the faithfulness of contrastive explanations. Contrastive explanations are judged by humans to be more useful for prediction than previous non-contrastive alternatives.
References


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.


A Supplementary Materials

A.1 Generating Contrastive Templates

Table 11 shows the complete list of contrastive patterns used in our work, categorized under different types of attributes/properties. For templates with no placeholders for the explainer to fill out, we only replace placeholders for answers (fact and foil). Table 9 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 be 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, Use case 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.

A.2 Details of 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 $0.30 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 $\kappa = 0.43$ (Landis and Koch, 1977).

A.3 Details on 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 $2 \times 10^{-5}$, batch size of 8, and dropout of 0.1, following (Lattinick 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 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.
Winogrande

Ian volunteered to eat Dennis’s menudo after already having a bowl because he despised eating

\[ a_1: \text{Ian} \]
\[ a_2: \text{Dennis} \]
\[ a_0: \text{he} \]
\[ c_{a_0}: \text{Ian volunteered to eat Dennis’s menudo after already having a bowl because he despised eating} \]
\[ c_{a_1}: \text{Ian volunteered to eat Dennis’s menudo after already having a bowl because Ian despised eating} \]
\[ c_{a_2}: \text{Ian volunteered to eat Dennis’s menudo after already having a bowl because Dennis despised eating} \]

PIQA (difference between answers is 1-2 words)

To prepare carrots before cooking with them, you can

\[ a_1: \text{Run them in the sink under boiling water} \]
\[ a_2: \text{Run them in the sink under cold water} \]
\[ a_0: \text{boiling water or cold water} \]
\[ c_{a_0}: \text{To prepare carrots before cooking with them, you can run them in the sink under boiling water or cold water} \]
\[ c_{a_1}: \text{To prepare carrots before cooking with them, you can run them in the sink under boiling water} \]
\[ c_{a_2}: \text{To prepare carrots before cooking with them, you can run them in the sink under cold water} \]

PIQA (difference between answers is larger)

To prevent gunk buildup in cup holders of a car,

\[ a_1: \text{place coffee filters inside of the cup holders.} \]
\[ a_2: \text{pour a thin layer of oil into the cup holders.} \]
\[ a_0: \text{place coffee filters inside of the cup holders or pour a thin layer of oil into the cup holders.} \]
\[ c_{a_0}: \text{To prevent gunk buildup in cup holders of a car, place coffee filters inside of the cup holders or pour a thin layer of oil into the cup holders.} \]
\[ c_{a_1}: \text{To prevent gunk buildup in cup holders of a car, place coffee filters inside of the cup holders} \]
\[ c_{a_2}: \text{To prevent gunk buildup in cup holders of a car, pour a thin layer of oil into the cup holders.} \]

Table 9: 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 10: Input to Explainer and Task model for Abstractive Evaluation
<table>
<thead>
<tr>
<th>Complete list of Contrastive Prompt Templates</th>
<th>Commonsense Task/Instance Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Temporal:</strong>&lt;br&gt;OPT1 happened before/after OPT2&lt;br&gt;OPT1 takes longer than OPT2&lt;br&gt;OPT1 takes longer to than OPT2&lt;br&gt;OPT1 happened for a longer time than OPT2</td>
<td>PIQA (Consists of events)</td>
</tr>
<tr>
<td><strong>Personal Characteristics:</strong>&lt;br&gt;OPT1 likes _ while OPT2 likes _&lt;br&gt;OPT1 likes _ while OPT2 does not like _&lt;br&gt;OPT1 likes to _ while OPT2 likes to _&lt;br&gt;OPT1 likes to _ while OPT2 does not like to _&lt;br&gt;OPT1 prefers _ while OPT2 prefers _&lt;br&gt;OPT1 prefers _ while OPT2 does not prefer _&lt;br&gt;OPT1 prefers to _ while OPT2 prefers to _&lt;br&gt;OPT1 prefers to _ while OPT2 does not prefer to _&lt;br&gt;OPT1 thinks _ while OPT2 thinks _&lt;br&gt;OPT1 thinks _ while OPT2 does not think _</td>
<td>WSC (if PERSON entity tag is detected)</td>
</tr>
<tr>
<td><strong>Object Characteristic:</strong>&lt;br&gt;OPT1 is/are smaller than OPT2&lt;br&gt;OPT1 is/are larger than OPT2&lt;br&gt;OPT1 is/are slower than OPT2&lt;br&gt;OPT1 is/are faster than OPT2&lt;br&gt;OPT1 are _ than OPT2&lt;br&gt;OPT1 is _ while OPT2 is _&lt;br&gt;OPT1 is _ but OPT2 is _&lt;br&gt;OPT1 is _ however OPT2 is _&lt;br&gt;OPT1 are _ while OPT2 are _&lt;br&gt;OPT1 are _ but OPT2 are _&lt;br&gt;OPT1 are _ however OPT2 are _&lt;br&gt;OPT1 has _ while/OPT2 has/does not have _&lt;br&gt;OPT1 have _ while/OPT2 have/do not have _&lt;br&gt;OPT1 is made of/to _ however OPT2 is made of/to _&lt;br&gt;OPT1 is made of/to _ while/OPT2 is made of/to _</td>
<td>WSC and PIQA</td>
</tr>
<tr>
<td><strong>Spatial:</strong>&lt;br&gt;OPT1 is above OPT2&lt;br&gt;OPT1 is below OPT2&lt;br&gt;OPT1 is to the right of OPT2&lt;br&gt;OPT1 is to the left of OPT2&lt;br&gt;OPT1 is inside OPT2&lt;br&gt;OPT1 is outside OPT2&lt;br&gt;is closer to OPT1 and farther away from OPT2&lt;br&gt;OPT1 is closer to _ while OPT2 is farther away from _</td>
<td>WSC and PIQA</td>
</tr>
<tr>
<td><strong>Usecase:</strong>&lt;br&gt;OPT1 can _ while OPT2 can/cannot _&lt;br&gt;OPT1 is/can be used for OPT2&lt;br&gt;OPT1 is/can be used to do OPT2&lt;br&gt;OPT1 is/can be used for _ but OPT2 cannot&lt;br&gt;OPT1 is/can be used for _ while OPT2 is used for _&lt;br&gt;OPT1 is/can be s used for _ but OPT2 is used for _&lt;br&gt;OPT1 is/can be used to _ while OPT2 is used to _&lt;br&gt;OPT1 is/can be used to _ but OPT2 is used to _</td>
<td>WSC(No PERSON entity) and PIQA</td>
</tr>
<tr>
<td><strong>Causes:</strong>&lt;br&gt;OPT1 has _ because _ while OPT2 is _ because _&lt;br&gt;OPT1 can cause _ while OPT2 causes/results in _&lt;br&gt;Since _ it can OPT1 but not OPT2&lt;br&gt;Since _ it can OPT1 but because it is not _ it can’t OPT2</td>
<td>WSC (No PERSON entity) and PIQA</td>
</tr>
<tr>
<td><strong>Miscellaneous:</strong>&lt;br&gt;_ can be OPT1 but cannot be OPT2&lt;br&gt;OPT1 means to _ while OPT2 means to <em>&lt;br&gt;OPT1 is defined as _ while OPT2 is defined as <em>&lt;br&gt;</em> OPT1 _ OPT2&lt;br&gt;</em> OPT1 but not OPT2&lt;br&gt;OPT1 exists while an OPT2 doesn’t</td>
<td>WSC (No PERSON entity) and PIQA</td>
</tr>
</tbody>
</table>

Table 11: Complete list of contrastive patterns used in this work.