STaR: Self-Taught Reasoner
Bootstrapping Reasoning With Reasoning

Eric Zelikman[*1], **Yuhuai Wu**[*12], **Noah D. Goodman**[1]
[1]Department of Computer Science, Stanford University
[2] Google Research
{ezelikman, yuhuai, ngoodman}@cs.stanford.edu

Abstract

Generating step-by-step "chain-of-thought" rationales improves language model performance on complex reasoning tasks like mathematics or commonsense question-answering. However, inducing language model rationale generation currently requires either constructing massive rationale datasets or sacrificing accuracy by using only few-shot inference. We propose a technique to iteratively leverage a small number of rationale examples and a large dataset without rationales, to bootstrap the ability to perform successively more complex reasoning. This technique, the "Self-Taught Reasoner" (STaR), relies on a simple loop: generate rationales to answer many questions, prompted with a few rationale examples; if the generated answers are wrong, try again to generate a rationale given the correct answer; fine-tune on all the rationales that ultimately yielded correct answers; repeat. We show that STaR significantly improves performance on multiple datasets compared to a model fine-tuned to directly predict final answers, and performs comparably to fine-tuning a 30$\times$ larger state-of-the-art language model on CommensenseQA. Thus, STaR lets a model improve itself by learning from its own generated reasoning.

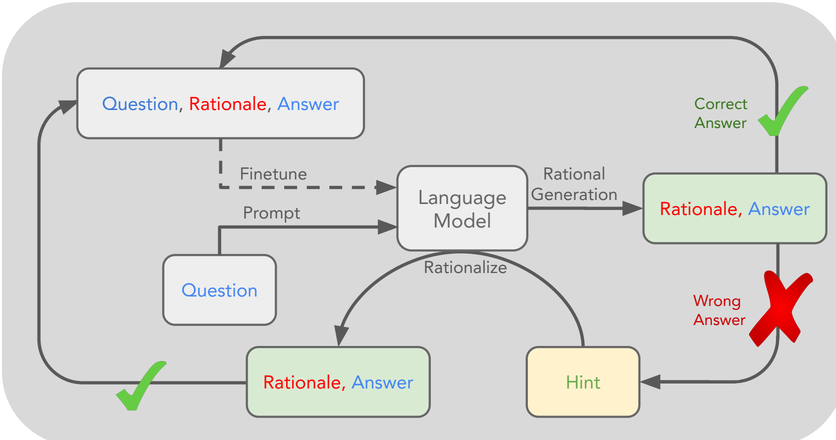
1 Introduction

Human decision-making is often the result of extended chains of thought [James et al., 1890, Ericsson and Simon, 1984]. Recent work has shown that explicit intermediate reasoning (“rationales”) can also improve the performance of language models [Rajani et al., 2019, Shwartz et al., 2020, Nye et al., 2021, Wei et al., 2022, Marasović et al., 2021]. For example, Nye et al. [2021] demonstrated that when explicitly trained to use a “scratchpad” for intermediate steps, large language models (LLMs) can attain perfect in-distribution performance, and strong out-of-distribution generalization on arithmetic, even where a model trained to predict the answer directly fails to do either of these. This line of work indicates that generating explicit rationales before giving a final answer (“rationale generation”) is valuable for LLMs across a wide range of tasks including mathematical reasoning, commonsense reasoning, code evaluation, social bias inference, and natural language inference. However, the two primary methods for inducing rationale generation both have serious drawbacks.

One approach to rationale generation is the construction of a fine-tuning dataset of rationales, either manually by human annotators or automatically using hand-crafted templates [Rajani et al., 2019, Cobbe et al., 2021, Shwartz et al., 2020, Nye et al., 2021]. Manual methods are expensive, and it is infeasible to construct such a dataset for every interesting dataset, especially larger ones for naturalistic tasks [Rajani et al., 2019]. Automatic, template-based methods rely on engineered, automatically-generated rationales but only work in contexts where a general solution is already known [Nye et al., 2021] or reasonable hard-coded heuristics can be developed [Shwartz et al., 2020].

There are also few-shot rationale methods, leveraging in-context learning, which have been shown to improve accuracy on mathematical and symbolic reasoning tasks [Nye et al., 2021, Wei et al., 2022]. Yet, while few-shot techniques with rationales tend to outperform their non-reasoning counterparts, they generally substantially underperform models fine-tuned to directly predict answers using larger datasets [Wei et al., 2022, Nye et al., 2021].

[*]These authors contributed equally to this work




```
Q: What can be used
to carry a small dog?
Answer Choices:
(a) swimming pool
(b) basket
(c) dog show
(d) backyard
(e) own home
A: The answer must be
something that can be
used to carry a small
dog. Baskets are
designed to hold things.
Therefore, the answer
is basket (b).
```

Figure 1: An overview of STaR and a STaR-generated rationale on CommonsenseQA. We indicate the fine-tuning outer loop with a dashed line. The questions and ground truth answers are expected to be present in the dataset, while the rationales are generated using STaR.

We adopt a different approach: by leveraging the LLM's pre-existing reasoning ability, we iteratively *bootstrap* the ability to generate high-quality rationales. Specifically, we few-shot prompt a large language model to self-generate rationales and refine the model's ability further by fine-tuning on those rationales that lead to correct answers. We repeat this process, using the improved large language model to generate the next training set each time. This is a synergistic process, where improvements in rationale generation improve the training data, and improvements in training data result in further improvements in rationale generation.

However, we find this basic iterative approach eventually saturates within the training set because it receives no direct training signal for problems it fails to solve. To overcome this effect, we propose the use of "rationalization": for each problem that the model fails to answer correctly, we generate a new rationale by providing the model with the correct answer. This lets the model reason backward — given the correct answer, the model can more easily generate a useful rationale. These rationales are then collected as part of the training data. We find they significantly improve training quality.

We thus develop the Self-Taught Reasoner (STaR, Fig. 1) method, a scalable bootstrapping method allowing models to learn to generate their own rationales, while also learning to solve increasingly difficult problems. In our method, we repeat the following process: in each iteration, first construct a finetuning-dataset by attempting to solve the dataset using the current model's **rationale generation** ability; then, augment this dataset using **rationalization**, justifying ground-truth answers to problems the model failed to solve; finally, finetune the large language model on the combined dataset.

Applying STaR on both arithmetic and commonsense reasoning, we observe it is able to effectively translate a small number of few-shot prompts into a large rationale dataset, while yielding corresponding performance improvements. On CommonsenseQA [Talmor et al., 2019], we find STaR improves over both a few-shot baseline (36.6%) and a baseline fine-tuned to directly predict answers (60.0%), and performs comparably to a 30× larger model (73.0%), attaining a performance of 72.3%.

Thus, we make the following contributions:

1. We propose a bootstrapping mechanism to iteratively generate a dataset of rationales from only a handful of initial examples with rationales — examples which do not require the explanations to be verified.
2. We complement **rationale generation** with **rationalization**, where a model is tasked with justifying an answer and then fine-tuned as if it had come up with the rationale without any hint. We show rationalization allows accelerates and improves this bootstrapping process.
3. We evaluate these techniques with a variety of ablations in both mathematical and commonsense reasoning domains.
4. We propose what is, to our knowledge, the first technique to allow a generally pre-trained large language model to iteratively use its language modeling capacity to improve itself.

2 Background and Related Work

In-context Learning Recently, a collection of works has emerged exploring the capacity for large language models to perform in-context learning [Brown et al., 2020, Wei et al., 2021]. In essence, in-context learning treats few-shot learning as a language modelling problem, by showing a few examples in the context (i.e. prompt), and allowing the model to learn and identify the pattern to apply to new examples. Some have studied in-context learning based on the language modeling objective in terms of Bayesian inference Xie et al. [2021] while others have attempted to describe the process more mechanistically in terms of "induction heads" [Olsson et al., 2022]. Moreover, differences in prompt configurations have been known to have dramatic effects on few-shot performance. Some have even found that replacing few-shot prompts with a "soft prompt" which can be optimized in embedding space results in noticeable gains [Lester et al., 2021]. Instead of emphasizing the representation of the question, we focus on the model output; in particular, we focus on the model's ability to reason through a problem before coming to a conclusion.

Rationales One of the initial works on the impact of rationales on language model performance was Rajani et al. [2019], showing that training a language model on a dataset with explicit rationales preceding the answer could improve a model's ability to generate the final answer. However, this required many thousands of training examples to be manually annotated with human reasoning. Recently, Nye et al. [2021] demonstrated that step-by-step "scratchpads" can improve fine-tuned large language model performance and generalization on tasks such as arithmetic, polynomial evaluation, and program evaluation. Similarly, Wei et al. [2022] used a single few-shot "chain-of-thoughts" reasoning prompt in order to improve model performance on a collection of tasks, without fine-tuning. Finally, Polu et al. [2022] showed that a curriculum learning approach could help solve formal math problems, as long as 1) they were translated into Lean (a theorem-proving language [Moura et al., 2015]), 2) one could directly evaluate the validity of the proofs, 3) one could sample numerous potential solutions for each problem, 4) had trained a separate value function model, and 5) started with GPT-f (a model already fine-tuned on a large math dataset [Polu and Sutskever, 2020]). Clearly, there are many domains where these conditions do not all apply.

Iterated Learning A variety of iterated learning algorithms have been proposed, where solutions or successful methods which are found are in turn used to find additional solutions [Anthony et al., 2017, Vani et al., 2021, Polu et al., 2022]. Anthony et al. [2017] introduced Expert Iteration (ExIt), a reinforcement learning technique serving as an inspiration for our approach. Essentially, it consists of a loop of self-play by an "apprentice," followed by imitation learning with feedback from a slower "expert" and then the replacement of the expert with the now-improved apprentice. Polu et al. [2022] builds off of this technique for formal reasoning, while Vani et al. [2021] applies iterated learning to visual question answering using modular networks which can be combined compositionally.

Natural Language Explanations Natural language explanations have also been discussed from the perspective of explainable machine learning, focusing on justification rather than reasoning [Camburu et al., 2018, Chen et al., 2021]. The motivation for this line of work is largely grounded in explainable decision making, and similarly to Rajani et al. [2019], generally does not find that requiring post-hoc explanations improves model performance.

3 Method

3.1 Rationale Generation Bootstrapping

We are given a dataset consisting of a set of problems X, with their corresponding answers Y, and a pretrained large language model M_0. Our technique starts with a handful of (e.g., 10) examples with rationales. We include them in a few-shot prompt, which is then used to prompt the model M_0 to solve each problem in the dataset, which will generate rationales followed by an answer. We assume that rationales that lead to correct answers are of better quality than those that lead to incorrect answers. Therefore, we filter the generated rationales to include only the ones which result in the correct answer. We fine-tune the base model M_0 on this filtered dataset, and then restart this process by generating the new rationales with the newly fine-tuned model. We keep repeating this process until the performance plateaus. Note that during this process, once we collect a new dataset, we always start training from the original pre-trained model M_0 instead of keep training one model to avoid overfitting. We provide an outline of this algorithm in Algorithm 1.

There are some similarities between STaR and expert iteration methods. For example, the filtering of generated examples based on whether their ultimate answer matches the target can be seen as expert feedback. However, we have a fixed "expert" and do not train a separate value function.

Algorithm 1 Rationale Generation Bootstrapping

Input M_0: an initial pretrained LLM; questions X w/ few-shot prompts, ground truth answers Y

$\mathrm{M} \leftarrow \mathrm{M}_0$ # Copy the original model
for `iteration` **in** `n_iterations` **do** # Outer loop
  (`rationales`, $\hat{Y}$) $\leftarrow \mathrm{M}(X)$ # Perform rationale generation
  `D`, _ $\leftarrow$ filter_correct(`rationales`, $\hat{Y}$) # Filter rationales using ground truth answers
  $\mathrm{M} \leftarrow$ train(M_0, `D`) # Finetune the original model on the correct solutions - inner loop
end for

3.2 Rationalization

The rationale generation bootstrapping algorithm described in Algorithm 1 carries one major limitation. Since the model is only trained on the examples which it answers correctly, the algorithm can saturate and stops improving when the model fails to solve new problems. This is fundamentally due to the fact that the algorithm cannot obtain any training signal from failed examples after saturation.

Inspired by Rajani et al. [2019], we propose a technique we call "rationalization". Specifically, we provide the answer as a hint to the model and ask it to generate rationales in the same style as in the previous rationale generation. Given the answer, the model is able to reason backwards, hence more easily to generate a rationale that leads to the correct answer. For example, in Figure 2, we provide the hint that "(b) grocery cart" is the correct answer in the prompt to generate the rationale. We apply rationalization to the problems which the models previously fail to solve. When adding a rationale generated by rationalization to our dataset, we do not include the hint in its corresponding prompt, as if the model had come up with the rationales without the hint. After filtering, we fine-tune our model on the previously generated dataset combined with the dataset generated by rationalization. Algorithm 2 describes the complete algorithm, while Figure 1 provides an overview diagram.

```
Q: Where do you put your grapes just
before checking out?
Answer Choices:
(a) mouth
(b) grocery cart (CORRECT)
(c) super market
(d) fruit basket
(e) fruit market
A: The answer should be the place
where grocery items are placed before
 checking out. Of the above choices,
grocery cart makes the most sense for
 holding grocery items. Therefore,
the answer is grocery cart (b).
```

Figure 2: A few-shot prompt hint we use for rationalization (and not for rationale generation), using the rationale from Wei et al., with its hint included, followed by the rationale and the answer generated by the model.

Algorithm 2 STaR

Input M_0: an initial pretrained LLM; questions X w/ few-shot prompts, ground truth answers Y

$\mathrm{M} \leftarrow \mathrm{M}_0$
for `iteration` **in** `n_iterations` **do** # Outer loop
  (`rationales`, $\hat{Y}$) $\leftarrow \mathrm{M}(X)$ # Perform rationale generation
  `D`, $X_{wrong} \leftarrow$ filter_correct(`rationales`, $\hat{Y}$)
  (`rationales`$_{hint}$, $\hat{Y}_{hint}$) $\leftarrow$ M(add_hint(X_{wrong})) # Perform rationalization
  `D`$_{rat} \leftarrow$ filter_correct(`rationales`$_{hint}$, $\hat{Y}_{hint}$)
  $\mathrm{M} \leftarrow$ train(M_0, `D` $\cup$ `D`$_{rat}$) # Finetune original model on correct solutions – inner loop
end for

Fine-tuning on the dataset generated by rationalization has a crucial benefit of exposing the model to difficult problems which otherwise would not have appeared in its finetuning dataset. This can be understood as challenging the model to "think outside the box" about the problems on which it was unsuccessful. A secondary benefit of this approach is that it expands the dataset size.

4 Experiments

For our experiments, we focus on arithmetic and commonsense reasoning to demonstrate the breadth of STaR. In particular, for arithmetic, we follow the setup introduced by Nye et al. [2021]. For the commonsense question-answering problems, we follow Xie et al. [2021], Wei et al. [2022] and use CommonsenseQA, a widely used multiple-choice dataset for this domain [Talmor et al., 2019].

4.1 Experimental Protocol

We used GPT-J as our base language model, and the fine-tuning script from the GPT-J repository [Wang, 2021]. GPT-J contains 6 billion parameters: a 28-layer decoder-only transformer, with an embedding size of 1024, 16 attention heads of dimension 256, and an FFN hidden layer of size 16384. It was pre-trained on the Pile [Gao et al., 2020], with a vocabulary size of 50.4K. We chose GPT-J because the checkpoint and fine-tuning code are publicly available [Wang, 2021], and the model is large enough to generate rationales of non-trivial quality to be bootstrapped from.

In general, unless otherwise stated, we use a batch size of 8 sequences, each of length 1024. We also use packing, namely, packing the shorter examples to form longer sequences (up to length 1024) to improve TPU utilization. We do not use weight decay, and we train and sample on a single TPU-v3 node. We performed a hyperparameter search over learning rates from 10^{-7} to 10^{-4} using the Adam optimizer Kingma and Ba [2014]. We found that 10^{-6} was consistently the best-performing learning rate. Following the default setting of Wang [2021], we perform a 100-step learning rate warmup, from which point we use a constant learning rate. Unless stated otherwise, we start with 40 training steps at the first outer loop, and increase the number of fine-tuning training steps by 20% with each outer loop. In general, we found that training more slowly at the beginning ultimately benefits the model performance. We expect that further improvement is possible with a more thorough hyperparameter search — we leave this to future work due to computational constraints.

On the arithmetic problems, we first generate a dataset of 50,000 randomly sampled questions (uniformly over the digit lengths) in the format introduced by Nye et al. [2021]. For each outer loop iteration on arithmetic, we sample 10,000 problems from the dataset. We use 10 random few-shot rationale examples for each digit for its corresponding few-shot prompt. For each outer loop on CommonsenseQA we shuffle and prompt with each example in the complete training dataset of $9,741$ commonsense reasoning questions. For few shot prompting on CommonsenseQA, we start with the same 10 questions as used in Wei et al. [2022], with the rationales modified slightly to fix an incorrect answer and to more-explicitly reference relevant knowledge. We include these modified prompts in Appendix B[1]. These prompts serve as our complete set of explanations. We keep running STaR until we see performance saturation, and we report the best results.

When performing rationalization we find that the choice to include or not include few-shot prompts on iterations after the first outer-loop iteration does not have a substantial impact on the method's ultimate performance. However, there are some nuances which we discuss further in Section 5. One technique originally proposed in Wei et al. is that training including scratchpads improves performance for scratchpads.

4.2 Datasets

Arithmetic The arithmetic dataset calculates the sum of two n digit integers. We generate the dataset based on the descriptions provided by Nye et al. [2021].

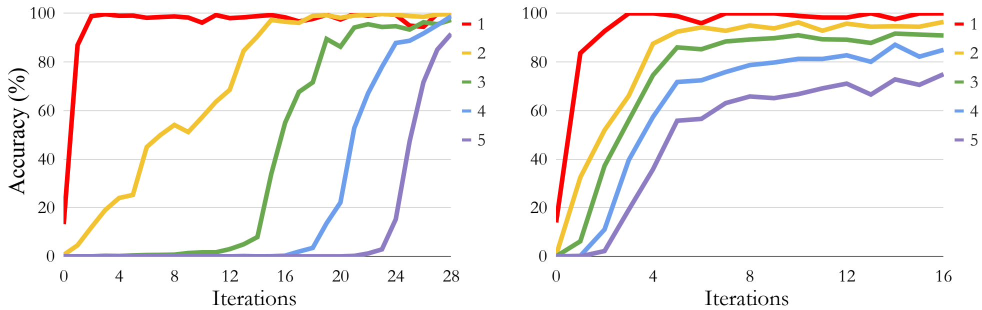
We visualize an example scratchpad in Figure 3. Everything up to and including "`Target:`" is given as part of a prompt, and the model is asked to generate the scratchpad (start/end indicated by "`<scratch>`") and the final answer, as in Nye et al. [2021]. Each line of the scratchpad corresponds to the summation of each pair of digits from the final digit to the first digit, the accumulating final digits of the answer, and a carry digit corresponding to whether the previous pair summed to at least 10. We include few-shot prompts for 1 to 5 digits, and evaluate examples of at most 8 digits. When performing rationalization, we include the correct answer after "`Target`" and query the model to produce the scratchpad and then reproduce the correct answer following the scratchpad.

```
Input:
6 2 4 + 2 5 9
Target:
<scratch>
6 2 4 + 2 5 9 , C: 0
2 + 5 , 3  C: 1
6 + 2 , 8 3  C: 0
, 8 8 3  C: 0
0 8 8 3
</scratch>
8 8 3
```

Figure 3: A visualization of a 3-digit arithmetic problem with a worked scratchpad. C corresponds to the carry from the summation from the previous digit.

CommonsenseQA The multiple-choice commonsense reasoning task, CommonsenseQA [Talmor et al., 2019] (CQA), is constructed based off of ConceptNet, a semantic graph of concepts and their relationships with over a million nodes [Speer et al., 2016]. Specifically, to construct CQA, Talmor et al. identified a set of "target" concepts in ConceptNet for each question, where the target concepts share a semantic relationship to one "source" concept. Then each crowdsourced question is generated

[1]Based on Min et al. [2022], we doubt this would affect Wei et al.'s few-shot performance meaningfully.



(a) Without rationalization

(b) With rationalization

Figure 4: A visualization of the accuracy of n-digit summation with each iteration of STaR with and without rationalization for arithmetic. Each series corresponds to the accuracy of one n-digit sum.

to allow a reader to disambiguate one concept from the others, while mentioning the source concept. In addition, two distractor answers are added. The dataset has 12,247 questions, each with five choices, with 9,741 in the train set, 1,221 in the dev set, and 1,285 in the (withheld) test set.

Corresponding to the broad variety of ConceptNet, CQA contains a diverse set of questions which require commonsense reasoning ability building off of standard world knowledge, where human performance is 89% [Talmor et al., 2019]. Many have pointed out that CQA contains a number of problematic questions and answers, along several dimensions. There are a large number of typos as well, not to mention questions which are fundamentally ambiguous[2]. We use it despite these issues as it is a particularly general and open-ended question-answering dataset relying on both common world knowledge and simple reasoning, which serves as a good test-bed for our method.

4.3 Symbolic Reasoning: Results on Arithmetic

The accuracies of the model across digits $1-5$ over each iteration of the outer loop are plotted in Figure 4. After running STaR for 16 iterations, the overall accuracy is 89.5%. For reference, a baseline trained on 10,000 examples for 5,000 steps attains 76.3% accuracy. Notably, few-shot accuracy on arithmetic problems is mostly negligible, even with rationales: accuracy on 2-digit addition is less than 1%, and accuracy on more digits is minimal. However, with STaR, the accuracy is able to improve quickly. After one fine-tuning iteration on the model's generated scratchpads, 2-digit addition improves to 32% from less than 1%. After five, the model can solve up to 5-digit summation with a higher than 50% solve rate.

Further, we found the model tended to saturate towards high-accuracy on all digits. However, we observed that improvement, while fairly consistent, was not strictly monotonic: as n-digit performance saturated, accuracy varied slightly from iteration to iteration. As rationalization allows the model to solve problems few-shot, we start STaR training with 300 steps rather than 40 (note, doing so sans rationalization causes overfitting on 1-digit addition), and increase training by 20 steps per iteration.

We thus draw attention to the difference between the performance curves of STaR with and without rationalization. Without rationalization, the performance improvement is punctuated: the model generally has poor performance on the n-digit sum until it has good performance on the $(n-1)$-digit sum. With rationalization, the model can learn many lengths at once, though not with equal accuracy.

4.4 Natural Language Reasoning: Commonsense Question Answering

The CommonsesenseQA (CQA) setting introduces several new challenges. In the arithmetic task, an incorrect scratchpad in the reasoning step, and to a lesser degree in the rationalization step, was extremely likely to result in an incorrect answer. On the other hand, CQA problems are 5-way multiple choice questions. Thus, one will get the right answer at random approximately 20% of the time, regardless of the quality of reasoning. Moreover, some simple heuristics (e.g. semantic similarity) can meaningfully improve this to ≈30% without any reasoning, as shown by Talmor et al.. We evaluate this dataset as described in the experimental protocol.

[2]For example, "Billy bought coffee and waited for his wife to arrive from France. Where might he have been?" includes airport and train station as options. The correct answer, perhaps surprisingly, is train station.

```
Q: John wants to eat at an Indian
restaurant in on top of the space
needle, where would he have to go?
Answer Choices:
(a) reservation (b) curry
(c) britsh town (d) seattle
(e) large city
A: The answer must be a place with
a space needle. The space needle is
a landmark in Seattle. Therefore,
the answer is Seattle (d).

Q: I put a towel in a cardboard box for
the kitten, I wanted it to have what?
Answer Choices:
(a) farmhouse (b) zoos (c) cold place
(d) floor (e) warm place
A: The answer must be a place where the
kitten would be comfortable. The kitten
would be comfortable in a warm place.
Therefore, the answer is warm place (e).

Q: Needlepoint requires many
accessories, what holds the work taut?
Answer Choices:
(a) sewing (b) collection of folk art
(c) livingroom (d) frame (e) bathroom
A: The answer must be something that
holds the work taut. The work is taut
because it is held in place. Therefore,
the answer is frame (d).
```

(a) Problems where the model was able to solve questions which it had previously answered incorrectly.

Q: They prided themselves on being a
wealth of knowledge, and that's why many
chose to attend their what?
Answer Choices:
(a) book store (b) university (c) meeting
(d) class (e) encyclopedia
~~A: The answer must be a place where people~~
~~go to learn about things. The answer is~~
~~university (b).~~
A: The answer must be a place where people
go to learn about things. Universities are
places where people go to learn about
things. Therefore, the answer is
university (b).

Q: You want a real estate agent that is
straight forward with you, the last
thing you want is for them to what a
house's quality?
Answer Choices:
(a) cover up (b) offer price
(c) misrepresent (d) lie (e) falsify
~~A: The answer must be a way to determine~~
~~the quality of a house. The answer is~~
~~misrepresent (c).~~
A: The answer must be something that
would be a bad thing for the real estate
agent to do. The real estate agent would
be misrepresenting the house's quality
if they were to lie about it. Therefore,
the answer is misrepresent (c).

(b) Problems where the model was already able to answer them with few-shot reasoning, but was able to answer better after more iterations.

Figure 5: Case study on STaR. We highlight some problems which the model is not initially able to solve, as well as rationales which are initially correct but improve. We note that for CQA, one is tasked with selecting the "best" answer, where multiple answers may be technically correct.

We compared our method to several baselines. The first baseline is to finetune GPT-J to directly output the final answer, which we call "GPT-J Finetuned". We also compare to GPT-3 finetuned to directly predict the final answer, based on Xu et al. [2021], which we label "GPT-3 Finetuned" and a 137B parameter Lambda model few-shot prompted with chain-of-thought rationales from Wei et al. [2022], labeled "Few-shot CoT LaMDA 137B."

We found that, as shown in Table 1, STaR without rationalization outperformed GPT-J fine-tuned directly on the final answer for the entire dataset, despite training on less of the data. However, the inclusion of rationalization improved this performance to 72.3%, far closer to the 73% of the $30\times$ larger GPT-3. As expected, we also see our model surpassed the few-shot baselines, including the much-larger 137B LaMDA model [Thoppilan et al., 2022, Wei et al., 2022]. We expect accuracy would be further improved if we applied STaR to a model with higher few-shot performance.

Case Study Note that it is harder to judge the rationale quality: for arithmetic, one can compare them to the ground truth rationales, but for CQA the evaluation is necessarily qualitative. For this reason, we include a case study in Figure 5. We observe that the rationales provided are generally coherent and of a similar structure to the few-shot rationales. We make the following two observations:

1. After training with STaR, we see the model was able to generate reasonable rationales that solve new problems, which explains part of the observed performance gain.
2. We also see that there were many instances in which STaR improved the quality of rationales over those generated in a few-shot manner.

	CQA Dev Set Accuracy (%)	Train Data Used (%)
GPT-3 Direct Finetuned [Xu et al., 2021]	*73.0*	100
Few-shot Direct GPT-J	20.9	~0
Few-shot CoT GPT-J [3]	36.6	~0
Few-shot CoT LaMDA 137B [Wei et al., 2022]	55.6	~0
GPT-J Direct Finetuned	60.0	100
STaR without Rationalization	68.8	69.7
STaR	**72.3**	86.7

Table 1: We evaluate a variety of baselines, including a few-shot GPT-J evaluation both with and without scratchpads, a GPT-J baseline finetuned to directly predict the answer, and two versions of STaR applied to GPT-J, both with and without rationalization. We use CoT to denote non-STaR models which output rationales, and Direct to indicate that which directly predict the final answer. Note the final STaR model is trained on 78.2% of the training dataset with rationale generation, and an additional 8.5% of examples from rationalization.

Preliminary Qualitative Analysis Based on the observation that STaR may improve reasoning quality for problems even when they were initially answered correctly via few-shot prompting, we performed a preliminary qualitative analysis. We randomly selected 20 rationales generated from few-shot CoT and STaR-generated rationales on questions which they both answered correctly. We then presented these questions and rationales to a third party in a randomized order (such that neither model was consistently first), asking them to select the rationale which they felt best justified the answer. They selected the STaR-generated rationales for 70% of the problems, more than twice as often as the few-shot rationales. We reproduce the test prompts in Appendix C. This indicates that, as mentioned in the case study, STaR can improve the quality of rationale generation.

Failure Cases Finally, we found a variety of interesting failure cases, many of which corresponded to standard logical fallacies. For example, the model often made statements related to the topic of the question but which were not actually arguments for why the answer should be true. Sometimes, the model claimed the question implied the answer as an argument, without explaining why. Other times, especially early in training, the model answered as if it has knowledge about a particular individual, instead of making a general statement - e.g. "the king's castle is a place where he feels safe" instead of "castles are places where kings feel safe." We provide examples and analyze errors in Appendix A.

Few-shot Scratchpad Prompt Training We note that including few-shot prompts during fine-tuning [Wei et al., 2021] appears to have a meaningful performance benefit (60.9% to 68.8% without rationalization, 69.9% to 72.3% with rationalization). For this reason we generally recommend its use for at least some portion of the training, though we discuss some caveats on the inclusion of scratchpads in sampling in Section 5.

5 Discussion and Challenges

The Role of Rationalization An essential question exists about exactly what role is played by rationalization. For example, it is difficult to evaluate the contribution played by its capacity to increase the dataset size - one cannot fairly augment a dataset of rationales without new rationales, but if the model could already correctly reason through all the problems, new rationales would not be necessary at all. On the other hand, the introduction of ad-hoc or erroneous rationales would be an unfair comparison as well, as they directly worsen the model's ability to generate reasoning.

Another explanation for the benefit of rationalization is that it allows the model to reverse-engineer the solution. However, the general applicability of this is not guaranteed: difficult problems in the real world posed to human learners (as well as mathematicians and theorists) often have a known final result, where the challenge is to derive a convincing justification.

Another observation is that, due to the low sampling temperature, at least the outputs from the initial iteration correspond to the examples where the model is most confident in its answer. This results in these reasoning examples providing a weaker gradient signal than the rationalization examples. As we retrain from the initial pre-trained model every time we run a fine-tuning iteration, the degree of this effect is difficult to measure directly. Each of these hypotheses warrants further analysis.

[3] We use a slightly different set of few-shot rationales from Wei et al. [2022] with same questions, for the reasons described in Section 4.1 - namely fixing typos and improving clarity.

Finally, we must point out that the method to add the "hint" does not follow immediately from the question and answer and in some contexts providing it may be nontrivial. An exploration of the various impacts of different hinting techniques and their generality is an avenue for future work.

Temperature One intuitive alternative to rationalization, if one seeks to expand the training dataset, is more and higher-temperature sampling. However, in practice, we found that this is counterproductive. In general, it substantially increases the likelihood of a correct answer despite incorrect reasoning, and training on bad or irrelevant reasoning prevents generalization. This is particularly clear in more-structured tasks, like arithmetic, where the scratchpads that the model learns to produce with a higher-temperature sampling approach diverge into meaninglessness and cause the model to stagnate. Overall, we found that higher temperatures as an alternative to rationalization (e.g. 0.5 or 0.7) consistently led to models worse than models with reasoning alone. In addition, as text generation by large language models is sequential (i.e. one cannot produce a token without producing the preceding token), generating text is a bottleneck and this is computationally far less efficient than rationalization. For example, generating 10 sample outputs is approximately 10 times slower than generating one sample output. However, one potentially valuable way to leverage multiple samples would be to use the method proposed in Wang et al. [2022], using the majority-vote result of multiple high-temperature scratchpads as a ground truth against which we compare a low-temperature scratchpad. This may allow one to apply STaR to a dataset of only questions, without answers.

Few-shot Prompting A noteworthy phenomenon is that the inclusion of few-shot prompting during sampling seems to dramatically reduce "drift" where later rationales become increasingly dissimilar from the initial few-shot set of rationales. One benefit of this is that the model may be less constrained by the quality and difficulty of the initial rationales, theoretically allowing it to generalize more. One potentially negative consequence is that the style of the rationales may less-closely match the original prompting style. Another benefit is in terms of computational resources - a shorter prompt length allows for a shorter sequence length when sampling. Technically, the point in training at which we "disable" few-shot prompts is another hyperparameter which we could tune, but we leave this to future work. In addition, by leaving prompts out after the initial outer-loop iteration, the model tends to perform gradually worse at rationalization as it trains for longer periods of time. As a result, it may be necessary to include some hints during training for long periods of time with this approach.

Ultimately, the choice to include few-shot prompts in later iterations of training appears to depend on the use-case: when the goal is consistent adherence to a particular prompt style, which may benefit explainability, include few-shot prompts in sampling; when the goal is a faster training loop, one may remove them. Moreover, it is possible that with other datasets or larger models there is an impact on performance, so we encourage this to be generally treated as a hyperparameter.

Bias It is important to note that STaR is designed to amplify the reasoning that leads to correct solutions on a given dataset. The implication of this with a dataset like CommonsenseQA is that if biases are "useful" in solving the dataset then they will be amplified. This is made worse by rationalization, as biased answers that the model may not naturally arrive at are, in a sense, pulled out of the model. The exact interaction between the bias in a dataset and the pre-existing bias learned by the model is unclear, and something to consider before real-world deployment of all large language models in general, but STaR-trained models in particular.

We find some encouraging initial results on this however: for questions where gender is not relevant, the model appears to disregard it in its explanation. For example: "Q: Where is a good place for a woman to store her sunglasses? $\rightarrow$ A: The answer must be a place where sunglasses are stored. Sunglasses are stored in purses. Therefore, the answer is purse (e)." or "Q: The only baggage the woman checked was a drawstring bag, where was she heading with it? $\rightarrow$ A: The answer must be a place where a drawstring bag is checked. The answer is airport (e)." Obviously, this question warrants a much more comprehensive study.

6 Conclusion and Future Work

We present the Self-Taught Reasoner, which iteratively improves a model's ability to generate rationales to solve problems. Essentially, we few-shot prompt a model to solve many problems in a step-by-step manner by generating rationales, and then prompt it to rationalize the correct answer for problems it gets wrong. Then, we finetune on both the initially correct solutions and rationalized correct solutions, and then start again. We find that this technique significantly improves the model's generalization performance on both symbolic reasoning and natural language reasoning.

At a high-level, we believe that using examples without reasoning to iteratively get better at reasoning is a highly general approach, and that STaR can serve as the basis of many more sophisticated techniques. For example, not all reasoning tokens contribute equally to the final conclusion. Identifying reasoning which is useful in settings where accidental correct answers are feasible is an open and important problem. We observed that many naive approaches to this problem failed (e.g. comparing answers between few-shot prompting with and without reasoning), so we believe that this could be an important future contribution.

In addition, many problems in many domains, including language modeling more broadly and imitation learning in general, can be posed as problems compatible with this framework. For example, closely related to entailment, the implicit reasoning that occurs for humans in order to produce a new sentence given the context of what has already been written is non-trivial, and making it explicit may allow for better natural language generation. This is particularly true for natural language generation problems that require careful planning such as natural language proofs or paper-writing. The domains need not be constrained to language either – tasks leveraging language models in other domains are a natural extension such as visual question answering [Fang et al., 2015] and language-guided reinforcement learning [Mu et al., 2022, Huang et al., 2022]. We are excited about this avenue of future work. In addition, there are also still a number of as-yet-to-be-fully-resolved questions present around the interactions between the hyperparameters of this model. We look forward to exploring these as well.

Ultimately, there are many possibilities opened by this method and these results, and we believe that we have only scratched the surface.

Acknowledgements

We thank Imanol Schlag for his detailed feedback about this work, as well as Markus Rabe, Aitor Lewkowycz, Rishi Bommasani, and Alex Tamkin. We thank Cem Anil for his very helpful insight that rationale finetuning performance can be improved if the training includes the few-shot rationales. We thank Google TPU Research Cloud for TPU access.

References

William James, Frederick Burkhardt, Fredson Bowers, and Ignas K Skrupskelis. *The principles of psychology*, volume 1. Macmillan London, 1890.

K Anders Ericsson and Herbert A Simon. *Protocol analysis: Verbal reports as data.* the MIT Press, 1984.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. Explain yourself! leveraging language models for commonsense reasoning. *ACL*, 2019.

Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Unsupervised commonsense question answering with self-talk. *EMNLP 2020*, 2020.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022.

Ana Marasović, Iz Beltagy, Doug Downey, and Matthew E Peters. Few-shot self-rationalization with natural language prompts. *arXiv preprint arXiv:2111.08284*, 2021.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 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, 2019.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *ICLR 2022*, 2021.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*, 2021.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, and et al. In-context learning and induction heads. *Transformer Circuits*, Mar 2022.

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

Stanislas Polu, Jesse Michael Han, Kunhao Zheng, Mantas Baksys, Igor Babuschkin, and Ilya Sutskever. Formal mathematics statement curriculum learning. *arXiv preprint arXiv:2202.01344*, 2022.

Leonardo de Moura, Soonho Kong, Jeremy Avigad, Floris van Doorn, and Jakob von Raumer. The lean theorem prover (system description). In *International Conference on Automated Deduction*, pages 378–388. Springer, 2015.

Stanislas Polu and Ilya Sutskever. Generative language modeling for automated theorem proving. *arXiv preprint arXiv:2009.03393*, 2020.

Thomas Anthony, Zheng Tian, and David Barber. Thinking fast and slow with deep learning and tree search. *Advances in Neural Information Processing Systems*, 30, 2017.

Ankit Vani, Max Schwarzer, Yuchen Lu, Eeshan Dhekane, and Aaron Courville. Iterated learning for emergent systematicity in vqa. *ICLR 2021*, 2021.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. e-snli: Natural language inference with natural language explanations. *Advances in Neural Information Processing Systems*, 31, 2018.

Hanxiong Chen, Xu Chen, Shaoyun Shi, and Yongfeng Zhang. Generate natural language explanations for recommendation. *arXiv preprint arXiv:2101.03392*, 2021.

Ben Wang. Mesh-Transformer-JAX: Model-Parallel Implementation of Transformer Language Model with JAX. `https://github.com/kingoflolz/mesh-transformer-jax`, May 2021.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. singh 2002 (2016). *arXiv preprint arxiv:1612.03975*, 2016.

Yichong Xu, Chenguang Zhu, Shuohang Wang, Siqi Sun, Hao Cheng, Xiaodong Liu, Jianfeng Gao, Pengcheng He, Michael Zeng, and Xuedong Huang. Human parity on commonsenseqa: Augmenting self-attention with external attention. *arXiv:2112.03254*, December 2021. URL https://www.microsoft.com/en-us/research/publication/human-parity-on-commonsenseqa-augmenting-self-attention-with-external-attention/. human parity result on CommonsenseQA.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2022. URL https://arxiv.org/abs/2203.11171.

Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. From captions to visual concepts and back. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1473–1482, 2015.

Jesse Mu, Victor Zhong, Roberta Raileanu, Minqi Jiang, Noah Goodman, Tim Rocktäschel, and Edward Grefenstette. Improving intrinsic exploration with language abstractions. *arXiv preprint arXiv:2202.08938*, 2022.

Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. *arXiv preprint arXiv:2201.07207*, 2022.

Appendix

A Error Patterns

Throughout our experiments, we came across a variety of interesting failure cases for commonsense reasoning. Note that all the final answers are correct – however, we take issue with the reasoning used in order to arrive at those answers.

A.1 Question Implies Answer

One key failure case was answers in the form of "the answer must be something that is <question property>. <answer> is <question property>. Therefore, the correct answer is <answer>." In these cases, the model fails to explain *why* the answer that it has chosen satisfies the question property.

```
Q: What might someone get from learning about science?
Answer Choices:
(a) headache
(b) see things differently
(c) increased knowledge
(d) accidents
(e) appreciation of nature
A: The answer must be something that someone would get from learning about
science. Learning about science would increase knowledge. Therefore, the
answer is increased knowledge (c).
```

These rationales, while perhaps useful to the model, read to us as opaque and unexplanatory.

A.2 Begging the Question

A related but stronger version of the previous failure case, while less common, is particularly uninsightful. Sometimes the model will imply the answer that it has chosen in its question.

```
Q: The christian rock band was on a large stage, the singer's foot got caught
on a wire and he had a small what?
Answer Choices:
(a) commit sin
(b) fail
(c) disobey
(d) stumble
(e) boo boo
A: The answer must be something that would cause the singer to stumble. The
singer's foot got caught on a wire, so he would stumble. Therefore, the
answer is stumble (d).
```

A.3 Exercise to the Reader

A rare failure case is when the model finds it unnecessary to justify its answer. For example:

```
Q: Other than school or from text where can one gain knowledge?
Answer Choices:
(a) book
(b) class
(c) thinking
(d) head
(e) meeting
A: The answer must be a place where someone can gain knowledge other than
school or from text. The answer is meeting (e).
```

A.4 World State Assertions

Sometimes, the model will assume that it knows something about a subject or a person whose name was used as a variable. This leads to somewhat comical examples of reasoning. Part of the reason for this is that generally, there is an expectation that good rationales will leverage understanding of more general classes of objects and appeal to the relationship between those general classes and the particular instance. For example, the argument that "a person would typically feel exhilaration from heights" is generally more compelling than the argument that "James would feel exhilaration from heights."

```
Q: A person's hand hurts after hitting the bottle. What might happen to
the bottle?
Answer Choices:
(a) alcoholism
(b) breaking
(c) fly
(d) nausea
(e) ticket
A: The answer must be something that happens to the bottle. The bottle is
broken. Therefore, the answer is breaking (b).

Q: James loved climbing.  What emotion might he feel from heights?
Answer Choices:
(a) exhaustion
(b) exhilaration
(c) falling down
(d) nervousness
(e) vertigo
A: The answer must be a feeling that would cause someone to climb a mountain.
James would feel exhilaration from heights. Therefore, the answer is
exhilaration (b).
```

A.5 Red Herrings

Some errors in reasoning corresponded to the model making a statement which, while technically true, is not useful in demonstrating the claim.

```
Q: A squirrel ran up to it's home, where is the small dog likely to be
barking up?
Answer Choices:
(a) manhattan apartment
(b) wood
(c) neighbor's yard
(d) tree
(e) basket
A: The answer must be a place where a small dog would be likely to be
barking. Squirrels are not small dogs. Therefore, the answer is tree (d).

Q: John ran very fast, his feet touched the ground in what way?
Answer Choices:
(a) behind
(b) loose
(c) feverishly
(d) slowly
(e) lightly
A: The answer must be a way of running that is different from the way John
ran. Running is a fast, light-footed activity. Therefore, the answer is
lightly (e).
```

A.6 Hint Short-cutting

In the experiments where the model was shown some examples of "hints" during training, in order to prevent it from losing the ability to perform rationalization over time, the model appeared to pick up on the fact that the final answer would always correspond to the hinted answer. This led to answers such as

```
Q: Billy bought coffee and waited for his wife to arrive from  France.
Where might he have been?
Answer Choices:
(a) airport
(b) grocery store
(c) internet cafe
(d) supermarket
(e) train station
A: The answer must be a place where Billy could have been waiting for his
wife to arrive from France. The airport is a place where people can wait
for flights. Therefore, the answer is train station (e).
```

B Modified CQA Prompts

For reference, we include our modified prompts based closely on those in Wei et al. [2022].

```
Q: What do people use to absorb extra ink from a fountain pen?
Answer Choices:
(a) shirt pocket
(b) calligrapher's hand
(c) inkwell
(d) desk drawer
(e) blotter
A: The answer must be used to absorb extra ink. Blotters are designed to absorb
liquids. Therefore, the answer is blotter (e).

Q: What home entertainment equipment requires cable?
Answer Choices:
(a) radio shack
(b) substation
(c) television
(d) cabinet
(e) desk
A: The answer must require cable. Cable is used to provide satellite channels to
televisions. Therefore, the answer is television (c).

Q: The fox walked from the city into the forest, what was it looking for?
Answer Choices:
(a) pretty flowers
(b) hen house
(c) natural habitat
(d) storybook
(e) dense forest
A: The answer must be a reason for a fox to go into the forest. The forest is a fox'
s natural habitat. Therefore, the answer is natural habitat (c).

Q: Sammy wanted to go to where the people were. Where might he go?
Answer Choices:
(a) populated areas
(b) race track
(c) desert
(d) apartment
(e) roadblock
A: The answer must be a place with many people. Populated areas, by definition, have
 a lot of people. Therefore, the answer is populated areas (a).

Q: Where do you put your grapes just before checking out?
Answer Choices:
(a) mouth
(b) grocery cart
(c) super market
(d) fruit basket
(e) fruit market
A: The answer should be the place where grocery items are placed before checking out
. Of the above choices, grocery cart makes the most sense for holding grocery items.
 Therefore, the answer is grocery cart (b).

Q: Google Maps and other highway and street GPS services have replaced what?
Answer Choices:
(a) united states
(b) mexico
(c) countryside
(d) atlas
(e) oceans
A: The answer must be something that used to do what Google Maps and GPS services do
, which is give directions. Atlases were also used to give directions. Therefore,
the answer is atlas (d).
```

```
Q: Before getting a divorce, what did the wife feel who was doing all the work?
Answer Choices:
(a) harder
(b) anguish
(c) bitterness
(d) tears
(e) sadness
A: The answer should be a feeling which would cause someone who was doing all the
work to get divorced. If someone feels bitter towards their spouse, they are likely
to want a divorce. Therefore, the answer is bitterness (c).
```

C Human-evaluated Test Prompts

We also selected a random sampling of 20 questions which were correctly answered both few-shot and by a STaR-trained model. Presented in a random order, a third party preferred the STaR-generated examples 70% of the time, more than twice as often as the few-shot explanations. While this is still an extremely preliminary result, we find it encouraging. We reproduce the examples here with the few-shot answer first and then the STaR-trained answer second.

```
Q: Who is likely yo have a caring heart?
Answer Choices:
(a) person
(b) species
(c) artichoke
(d) all mammals
(e) live animal
A: The answer must be a person. All mammals have a caring
heart. Therefore, the answer is person (a).
A: The answer must be someone who is likely to have a caring
 heart. Humans are the most likely to have a caring heart.
Therefore, the answer is person (a).

Q: If a person wanted to become a good writer what should
they do in college?
Answer Choices:
(a) word sentence
(b) own animal
(c) read the newspaper
(d) catch cold
(e) study literature
A: The answer must be something that would help a person
become a good writer. The answer is study literature (e).
A: The answer must be something that would help a person
become a good writer. Literature is a form of study.
Therefore, the answer is study literature (e).

Q: The human always would have fun making up questions for
the A.I. overlords, he found the task quite what?
Answer Choices:
(a) do enjoy
(b) eat cake
(c) enjoy living
(d) get laid
(e) enjoyable
A: The answer must be something that the human would enjoy
doing. The human enjoys making up questions for the A.I.
overlords. Therefore, the answer is enjoyable (e).
A: The answer must be something that the human found
enjoyable. The human found the task of making up questions
for the A.I. overlords enjoyable. Therefore, the answer is
enjoyable (e).

Q: What happens when I make a pig of myself eating breakfast
?
Answer Choices:
(a) full stomach
(b) gain energy
(c) dieting
(d) have energy
(e) feeling satisfied
A: The answer must be a feeling which would cause someone to
 feel full after eating breakfast. The answer is full
stomach (a).
A: The answer must be a feeling which would cause someone to
 make a pig of themselves eating breakfast. If someone feels
 full, they are likely to stop eating. Therefore, the answer
 is full stomach (a).

Q: In order to see a story on the big screen what must you
do?
Answer Choices:
(a) go to movies
(b) visualize
(c) reading
(d) open book
(e) sketching a picture
A: The answer must be something that you do to see a story
on the big screen. Movies are a way to see a story on the
big screen. Therefore, the answer is movies (a).
A: The answer must be something that is required to see a
story on the big screen. Movies are a form of storytelling.
Therefore, the answer is go to movies (a).

Q: What is the opposite of remembering?
Answer Choices:
(a) answering question
(b) stopping to think
(c) knowing
(d) closing eyes
(e) forgetting
A: The answer must be something that is the opposite of
remembering. The answer is forgetting (e).
A: The answer must be a feeling which would cause someone to
 not remember something. If someone is forgetting something,
 they are likely to not remember it. Therefore, the answer
is forgetting (e).

Q: If you aren't experiencing curiosity and want to ensure
viewing enjoyment, what may you watch?
Answer Choices:
(a) examine thing
(b) stare into a mirror
(c) go somewhere
(d) hear news
(e) see favorite show
A: The answer must be something that will make you want to
watch. If you are not experiencing curiosity, you may want
to watch something that will make you want to watch.
Therefore, the answer is watch (e).
A: The answer must be something that will not cause you to
want to watch. If you are not experiencing curiosity, you
are likely to want to watch something that is entertaining.
Therefore, the answer is favorite show (e).

Q: The carotid is a main artery susceptible to damage where?
Answer Choices:
(a) neck
(b) face
(c) busy city
(d) torso
(e) human body
A: The answer must be a place where the carotid artery is
susceptible to damage. The carotid artery is located in the
neck. Therefore, the answer is neck (a).
A: The answer must be a place where damage to the carotid
would cause injury. The carotid is a main artery in the neck.
 Therefore, the answer is neck (a).

Q: The king needed to feel safe, where did he go?
Answer Choices:
(a) castle
(b) throne room
(c) deck of cards
(d) fort
(e) court
A: The answer must be a place where the king feels safe. The
 king's castle is a place where he feels safe. Therefore,
the answer is castle (a).
A: The answer must be a place where the king can feel safe.
Castles are places where kings feel safe. Therefore, the
answer is castle (a).

Q: What is the result of instituting civil action?
Answer Choices:
(a) bad feelings
(b) confusion
(c) going to court
(d) hurt feelings
(e) sandwiches
A: The answer must be a result of instituting civil action.
The result of instituting civil action is a lawsuit.
Therefore, the answer is going to court (c).
A: The answer must be a result of instituting civil action.
If someone goes to court, they are likely to feel hurt
feelings. Therefore, the answer is going to court (c).
```

```
Q: The person wanted to fix his car, what did he order?
Answer Choices:
(a) selling things
(b) gas
(c) feel resentful
(d) machine part
(e) cross street
A: The answer must be something that would help the person
fix his car. The person would not want to sell things,
because he would not want to give up his car. Therefore, the
 answer is machine part (d).
A: The answer must be something that would be used to fix a
car. Machine parts are used to fix cars. Therefore, the
answer is machine part (d).

Q: He always traveled up the mountain when winter arrived,
he had a passion and desire to do what?
Answer Choices:
(a) play volleyball
(b) play hockey
(c) ski
(d) knit
(e) skate
A: The answer must be something that the person does when
winter arrives. The answer is skiing (c).
A: The answer must be something that the person did when
winter arrived. The answer is ski (c).

Q: Bob thought that his Pokemon cards were common and
worthless, but he was wrong about them. They were really
what?
Answer Choices:
(a) personal
(b) special
(c) shiny
(d) rare
(e) irregular
A: The answer must be something that is not common and
worthless. Pokemon cards are not common and worthless.
Therefore, the answer is rare (d).
A: The answer must be something that are rare. Pokemon cards
 are rare. Therefore, the answer is rare (d).

Q: The person was thirsty, what should he have done?
Answer Choices:
(a) eat
(b) sunny days
(c) compliments
(d) drink water
(e) all kinds of things
A: The answer must be something that would make the person
feel better. If the person is thirsty, he should drink water.
 Therefore, the answer is drink water (d).
A: The answer must be something that would make someone feel
 better if they were thirsty. If someone is thirsty, they
are likely to want to drink water. Therefore, the answer is
drink water (d).

Q: Where would you find many products for sale?
Answer Choices:
(a) market
(b) cost one dollar
(c) shelf
(d) salon
(e) demonstration
A: The answer must be a place where many products are sold.
A market is a place where many products are sold. Therefore,
 the answer is market (a).
A: The answer must be a place where a lot of products are
sold. Markets are the most common place to buy products.
Therefore, the answer is market (a).

Q: If man and mammoth wanted to meet, they would probably
have to do it during when?
Answer Choices:
(a) a baby shower
(b) ancient history
(c) storybook
(d) stone age
(e) forest
A: The answer must be a time when man and mammoth would meet.
 Stone age is the time when man and mammoth lived. Therefore,
 the answer is stone age (d).
A: The answer must be a time when man and mammoth would be
able to meet. The stone age was a time when man and mammoth
would be able to meet. Therefore, the answer is stone age (d
).

Q: They prided themselves on being a wealth of knowledge,
and that's why many chose to attend their what?
Answer Choices:
(a) book store
(b) university
(c) meeting
(d) class
(e) encyclopedia
A: The answer must be a place where people go to learn about
 things. The answer is university (b).
A: The answer must be a place where people go to learn about
 things. Universities are places where people go to learn
about things. Therefore, the answer is university (b).

Q: what happens when visiting museum that have narrow halls?
Answer Choices:
(a) sore feet
(b) amazement
(c) retrospection
(d) boredom
(e) crowding
A: The answer must be something that happens when visiting a
 museum that has narrow halls. Crowding is a common problem
in museums. Therefore, the answer is crowding (e).
A: The answer must be something that would cause someone to
have sore feet if they are walking in narrow halls. If
someone feels cramped, they are likely to want to go
somewhere else. Therefore, the answer is crowding (e).

Q: On their hike they brought a filtering straw, they were
worried about germs in the what?
Answer Choices:
(a) make sick
(b) doctor
(c) water
(d) stream
(e) mouth
A: The answer must be something that can filter out germs.
Filtering straws are used to filter out germs. Therefore,
the answer is filtering straw (c).
A: The answer must be something that would cause someone to
bring a filtering straw on a hike. Filtering straws are used
 to filter water. Therefore, the answer is water (c).

Q: If I am not good at playing poker, I will begin what if I
 play?
Answer Choices:
(a) discord
(b) social event
(c) losing money
(d) competition
(e) heightened awareness
A: The answer must be something that will happen if you are
not good at playing poker. If you are not good at playing
poker, you will begin to lose money. Therefore, the answer
is losing money (c).
A: The answer must be something that will help you if you
are not good at playing poker. If you are losing money, you
are likely to want to stop playing. Therefore, the answer is
 losing money (c).
```

D Example Rationalizations on CQA

We include a randomly sampled set of rationalizations which the model is able to produce before fine-tuning. We observe that sometimes, the model constructs an argument roughly of the form "the answer must have *a set of properties*. *correct answer* has *those properties*. therefore, the answer is *correct answer*." This structure of argument is fairly standard, but given that the model originally answered those questions incorrectly, it resembles template-matching more than reasoning. The technique of rephrasing the question in a simpler way before answering it directly is also often effective. In general, the rationales from the pretrained model's rationalization appears to be of similar quality to the pretrained model's rationale generation. We reference the pretrained model here rather than the STaR-trained model, as the final iteration's rationales are the ones which the model struggled to learn from – that is, if the rationales were good, then we might expect that the model would have already incorporated them into its rationale generation.

```
Q: Why do I eat just enough when eating breakfast?
Answer Choices:
(a) gas
(b) full stomach
(c) feeling satisfied (CORRECT)
(d) have energy
(e) bloated
A: The answer must be a reason for eating just enough when eating breakfast. Eating
just enough is a way to feel satisfied. Therefore, the answer is feeling satisfied (
c).

Q: How is a dog likely to communicate with another dog?
Answer Choices:
(a) reproduce
(b) bark (CORRECT)
(c) do many things
(d) clone
(e) jump up
A: The answer must be a way for a dog to communicate with another dog. Dogs bark to
communicate with each other. Therefore, the answer is bark (b).

Q: If a person is trying to weasel out of punishment for a crime, where are they
likely to be?
Answer Choices:
(a) court room (CORRECT)
(b) cherry tree
(c) chicken coop
(d) natural history museum
(e) jail
A: The answer must be a place where a person would be trying to get out of
punishment for a crime. The answer is court room (a).

Q: What would easily hide a snake?
Answer Choices:
(a) living room of a house
(b) kew gardens
(c) terrarium
(d) thick forest (CORRECT)
(e) tropical forest
A: The answer must be something that would hide a snake. The answer is thick forest
(d).

Q: The person didn't like the varying size of the cake slices handed out, she demand
 more what?
Answer Choices:
(a) fairness (CORRECT)
(b) candies
(c) compliments
(d) well fed
(e) arguements
```

```
A: The answer must be a reason for the person to demand more cake. The person is not
 happy with the size of the cake slices. Therefore, the answer is fairness (a).

Q: In America, the courthouse of any given town has a likely location, where is it?
Answer Choices:
(a) electrical circuit
(b) getting married
(c) capital city
(d) center of town (CORRECT)
(e) michigan
A: The answer must be a location where a courthouse is located. The capital city is
the location of the capital of a state. Therefore, the answer is center of town (d).

Q: Where might I find a bowl nestled among other bowls in the kitchen?
Answer Choices:
(a) refrigerator
(b) kitchen top
(c) cupboard (CORRECT)
(d) dishwasher
(e) apartment
A: The answer must be a place where a bowl is found. The kitchen is a place where
bowls are found. Therefore, the answer is cupboard (c).

Q: Which state has blue ridge mountains that are called Blue Mountain?
Answer Choices:
(a) appalachians
(b) virginia
(c) pennsylvania (CORRECT)
(d) north carolina
(e) montana
A: The answer must be a state with mountains. The answer is Pennsylvania (c).

Q: Where could you find many radio studio?
Answer Choices:
(a) radio station
(b) country
(c) clear area
(d) office building
(e) large city (CORRECT)
A: The answer must be a place where many radio studios are located. Radio studios
are used to broadcast radio programs. Therefore, the answer is large city (e).

Q: Where would someone bring you a cup?
Answer Choices:
(a) apartment
(b) closet
(c) restaurant (CORRECT)
(d) table
(e) party
A: The answer must be a place where someone would bring you a cup. A restaurant is a
 place where people bring cups. Therefore, the answer is restaurant (c).
```