Few-Shot Self-Rationalization with Natural Language Prompts

Anonymous ACL submission

Abstract

Self-rationalization models that predict task labels and generate free-text elaborations for their predictions could enable more intuitive interaction with NLP systems. These models are, however, currently trained with a large amount of human-written free-text explanations for each task which hinders their broader usage. We propose to study a more realistic setting of self-rationalization using few training examples. We present FEB—a standardized collection of four existing English-language datasets and associated metrics. We identify the right prompting approach by extensively exploring natural language prompts on FEB. Then, by using this prompt and scaling the model size, we demonstrate that making progress on few-shot self-rationalization is possible. We show there is still ample room for improvement in this task: the average plausibility of generated explanations assessed by human annotators is at most 51%, while plausibility of human explanations is 76%. We hope that FEB and our proposed approach will spur the community to take on the few-shot self-rationalization challenge.

1 Introduction

Models constrained to be more understandable to people are easier to troubleshoot and more useful in practice (Rudin et al., 2021). For instance, constraining a model that answers the question "Which linguist invented the lightbulb?" with "none" to also provide the reason—"Thomas Edison is the inventor of the lightbulb and he was not a linguist"—makes the model easier to control and interact with (Kim et al., 2021). Models that jointly predict task labels and generate *free-text explanations* for their predictions (as in the previous example) are known as *self-rationalization models* (Wiegreffe et al., 2021). Their explanations are arguably more faithful and stable than post-hoc explanations since they are intrinsic to the model (Melis and Jaakkola, 2018). The free-text format is essential for explaining tasks requiring reasoning about unstated knowledge such as commonsense (Marasović et al., 2020), and it makes explanations more intuitive to people compared to highlights of individual words (Camburu et al., 2018). Despite these benefits, self-rationalization models are not widely used, in part because their training currently requires an abundance of human-authored explanations for each task (Narang et al., 2020). A possible solution is few-shot learning, which has shown promising results in recent years. To help the research community begin tackling self-rationalization with only a few examples, we present (i) FEB—a standardized collection of four existing English-language datasets and associated metrics, and (ii) the first approach for the task established through an extensive evaluation of natural language prompts.[1]

One approach to few-shot learning is *prompt-based finetuning* with *natural language prompts*. Such prompts are produced by formatting finetuning instances using a format similar to that used in pretraining, based on the idea that finetuning examples that look similar to pretraining ones will be more informative in the fewshot setting. A few prompts are then used for finetuning. In this paper, we explore whether prompt-based finetuning can be extended to induce few-shot self-rationalization behavior in addition to few-shot prediction. To measure our progress, we first introduce FEB as benchmark dataset consisting of human authored free-text explanations across four distinct end tasks including natural language inference and commonsense tasks (§2). Since finding appropriate prompts is often challenging (Gao et al., 2021), we then extensively explore natural language prompts for few-shot self-rationalization. In our experiments, we fine-tune the T5 and UNIFIEDQA pretrained encoder-decoder transformers (Raffel et al., 2020; Khashabi et al., 2020), and show that versatile question-answering prompts (defined in §3.1) outperform prompts based on span infilling by 8.73

[1] **F**ew **E**xplanations **B**enchmark (**FEB**)

accuracy points, as well as prompts designed by following the most similar T5's supervised pretraining task by 3.21.

We then study the impact of model size on few-shot self-rationalization to investigate whether the quality of generated explanations scales with the size as good as the accuracy of predicting task labels. To this end, we also evaluate GPT-3's (Brown et al., 2020) self-rationalization behavior. Our experiments show that explanation plausibility scored by human annotators and end-task accuracy improve with increasing model size, despite models being overparametrized. Specifically, the difference in plausibility scores between the BASE and 3B model ranges from [6.24, 24.85] (on average 14.85). The average plausibility across datasets is 43.36 (UNIFIEDQA-3B) and 50.58 (GPT-3). While encouraging, our results show that there is still a large gap between model and human performance (25.75 for GPT-3), and we hope this work will help enable the research community to take on the few-shot self-rationalization challenge.

Our code for producing data splits, prompt construction, model training/evaluation, and human evaluation templates will be publicly available.

2 FEB Benchmark

There has been an explosion of interest in generating free-text explanations and in few-shot learning in the last 1–2 years. However, appropriate datasets and metrics for few-shot self-rationalization have not yet been established. We thus introduce the FEB benchmark—a suite of existing English-language datasets with human-authored free-text explanations and associated metrics for few-shot self-rationalization. We expect that FEB will simplify future model comparison and lower barriers to entry for those interested in working on this task.

Datasets in FEB To identify available datasets suitable for few-shot self-rationalization, we start with a recent overview of datasets with free-text explanations (Wiegreffe and Marasović, 2021) and filter them according to the following criteria: (i) the input is textual, (ii) the explanation consists of one sentence or 2–3 simple sentences, (iii) the task has a fixed set of possible labels, (iv) the explanation is human-authored, and (v) the dataset has at least 389 instances. We use the second and third criteria to narrow the scope to easier self-rationalization since we expect that few-shot self-rationalization is very challenging. The last requirement is introduced to have 48 training and 350 evaluation examples.

FEB Tasks		# Shots
E-SNLI (Camburu et al., 2018)	Classify the entailment relation between two sequences	16
ECQA (Aggarwal et al., 2021)	Answer a question, given five answer choices	48
COMVE (Wang et al., 2019)	Select one of two sequences as more nonsensical	24
SBIC (Sap et al., 2020)	Classify a post as offensive or not	24

Table 1: Tasks that we have included in FEB.

This gives us 5 datasets, 4 of which are included in FEB and overviewed in Table 1. These datasets span 4 different tasks: natural language inference, multiple-choice commonsense QA, nonsensical sentence selection, and offensiveness classification. We exclude COS-E (Rajani et al., 2019) as it is too noisy to be useful for modeling and evaluating self-rationalization (Narang et al., 2021).[2]

ECQA contains not only justifications of the correct answer, but also justifications that refute the incorrect answer choices. We use only the former since they answer "why is [input] assigned [label]?", just as explanations in other datasets that we have included in FEB. The SBIC dataset contains annotations of frames representing the social biases that are implied in language. We format these frames as a self-rationalization task as follows. We allow only two labels: "offensive" and "not offensive". If a post is not offensive, we assign it the explanation: "*This post does not imply anything offensive*." A post can be offensive because it targets an individual or a demographic group. In the former cases, a post is assigned the explanation: "*This post is a personal attack*." Otherwise, we define a set of rules to transform annotations of which identity-based group is targeted and what stereotypes of this group are referenced or implied into a single, coherent sentence; e.g., `group`: *women*, `stereotype`: *can't drive* → "*This post is offensive because it implies that women can't drive*".

This is, to the best of our knowledge, the most comprehensive collection of textual self-rationalization tasks that could also be used even when working in a high-resource setting.

Automatic Evaluation Evaluating self-rationalization—predicting task labels and generating explanations for the predicted labels—

[2]Since COS-E is still actively used, we report COS-E results in Tables 8 and 9 in Appendix.

requires end-task evaluation and assessing the explanation plausibility. We use accuracy as our end-task evaluation metric. Explanation plausibility may be described as a subjective satisfaction with how a given explanation justifies a label/answer (Yang et al., 2019). Kayser et al. (2021) present the largest currently available study on the correlation of NLG metrics with human judgments of free-text explanation plausibility and report that BERTscore (Zhang et al., 2020) is most correlated (although the correlation is still weak). Thus, we use BERTscore to evaluate the similarity between gold and generated explanations. Following Kayser et al., we assign zero BERTscore to explanations of incorrectly predicted instances.

We follow recent recommendations for reliable few-shot evaluation (Bragg et al., 2021). Specifically, we fix hyperparameters (HPs) and use 60 random train-dev splits with 350 examples in each dev set. For classification tasks, the number of shots (examples per label) is chosen such that we construct a balanced training set of size 48.[3] See Table 1 (col. 3) for exact values; for ECQA we sample 48 training examples. For each model, we report the mean and standard error of 60 mean accuracy/BERTscore values calculated on 60 dev sets of 350 examples.[4] Our HPs are reported in Table 7 in Appendix.

Human Evaluation For our final models (§4), we conduct a human evaluation of plausibility of generated explanations following prior work (Kayser et al., 2021; Marasović et al., 2020). For each model evaluation, Kayser et al. (2021) take the first 300 dev examples that are correctly predicted by the model. This means that the dev set subsets used for human evaluation differ across models that are evaluated. However, the overlap between the evaluation sets is maximized by fixing the order of dev instances and taking the first 300.

Prior work used a single train-dev split, while FEB has 60 train-dev splits. Multiple splits provides the opportunity to account for the variance caused by changing the random seed to produce a reliable estimate of plausibility of explanations produced with only a few examples. Therefore, we take the first 6 correctly predicted examples per train-dev split, i.e., 6*60=360 total instances. Moreover, for classification tasks, we propose to take the first 6/#labels correctly-predicted examples per label to have a balanced evaluation set.

Following Kayser et al. (2021), we conduct the human evaluation in two steps:

- **Step1**: Select the correct label/answer.
- **Step2**: Assess whether two explanations (gold and generated) justify the label/answer above.

The first step makes sure the annotators understood the task correctly and they are not able to submit their annotations if the answers are wrong. Ground-truth explanations are evaluated to implicitly influence annotators with a gold reference point when they evaluate generated explanations, and to measure the quality of explanation datasets. To evaluate explanations, annotators are asked "Does the explanation justify the answer?" and given the options {`“yes”`,`“weak yes”`,`“weak no”`,`“no”`}. These options are mapped to plausibility scores of $\{1, \frac{2}{3}, \frac{1}{3}, 0\}$, respectively. For each of the 360 examples, we calculate the mean plausibility score of 3 annotators and report the mean and the standard error of 360 mean scores. We also report the inter-annotator agreement calculated with Fleiss' kappa. Finally, models are evaluated independently to avoid penalizing worse models in the presence of explanations generated by a better model.

3 Prompting for Self-Rationalization

We approach few-shot self-rationalization with prompt-based finetuning using natural language (NL) prompts. The key idea behind NL prompts is that a pretrained language model (LM) is already well-positioned to solve the end-task if we format finetuning end-task examples as similar as possible to the format used in pretraining. Following that principle, in this section, we describe our prompting approach with T5 (Raffel et al., 2020) and comprehensively evaluate three distinct prompt types with FEB. Our results show that a unified question-answering (QA) prompt combined with a T5 variant that includes additional supervised multitask QA training (UNIFIEDQA; Khashabi et al., 2020) performs the best overall across tasks, when compared to three different alternative prompts as described below.

Self-rationalization models (Narang et al., 2020; Wiegreffe et al., 2021) are currently based on T5 for at least two reasons. First, T5 has been pretrained

[3]In early studies, we found that 48 gives models that are at least slightly above the random baseline across all four tasks.

[4]To calculate the standard error for accuracy/BERTscore we use $n = 60$. The training (and likewise, dev) sets across splits can overlap, so this error reflects the variability expected in average scores when repeating our experiment with 60 new random splits of the same data sets.

with many supervised tasks including classification and generation tasks, and self-rationalization involves both classification and generation. Second, T5 is one of the largest *open-sourced* and widely studied pretrained models, and higher LM performance is correlated with larger model size (Kaplan et al., 2020). Thus, all of our experiments are based on T5 (and the UNIFIEDQA variant when evaluating prompts based on a QA format). In this section, all results are obtained with the base version of these models and in §4 we scale model size.

When a LM is pretrained with masked language modeling (Devlin et al., 2019) only, an appropriate NL prompt is constructed by adding and infilling masked tokens (Jiang et al., 2020). T5, however, has been pretrained with span infilling and a suite of supervised tasks whose instances were formatted in various ways. One of these supervised tasks includes SQUAD 1.1 (Rajpurkar et al., 2016) which allows us to experiment with prompts based on QA templates. As a result, we were able to design several different types of NL prompts for T5 consistent with different aspects of its pretraining:

1. QA prompts (SQUAD_{T5} , $\text{QA}_{\text{SIMPLE}}$).
2. span-filling prompts (INFILLING),
3. prompts designed by following the formatting of the most similar T5's pretraining task (≈T5 ; see Table 1),

We illustrate these prompt types for COMVE in Table 11 in Appendix. The following sections describe these formats in detail and compare their performance using FEB.

3.1 QA Prompts

Formatting new instances as QA pairs has been shown to be useful for transfer learning from a QA model (Gardner et al., 2019). We first evaluate options for a versatile QA NL prompt for self-rationalization of tasks in FEB before comparing this approach with the other two prompt types (INFILLING and ≈T5) in §3.3. As alternative QA models, we investigate two models: T5 (which has been pretrained with QA supervision from SQUAD 1.1), and UNIFIEDQA (a T5 variant described in detail below). Since UNIFIEDQA was trained on a multitask mixture of many different QA datasets, these T5 variants allow us to examine the extent to which additional QA supervision can transfer to the few-shot self-rationalization setting.

Prior work (Bragg et al., 2021) introduced UNIFEW, a model based on UNIFIEDQA, that is finetuned on a few task-specific instances posed as QA. Despite its simplicity, UNIFEW achieves competitive few-shot learning performance with strong baselines for classification tasks. However, Bragg et al.'s prompts do not cover all task types in FEB, and the question structure in their prompts is highly task-specific (see Appendix A.1).

Alternatively, we propose to design QA prompts with a simple principle in mind—given a non-QA task, construct an equivalent QA task in the form of short "Is...?" or "What is...?" questions. Here, "Is...?" questions have yes/no answers (sometimes "maybe"), and task labels verbatim are answers to "What is...?" questions (e.g., "offensive" and "not offensive"). Then, for UNIFIEDQA and non-QA task in FEB, we develop prompts following the formats proposed in UNIFIEDQA (see Appendix A.1). We denote these prompts as $\text{QA}_{\text{SIMPLE}}$. For T5, we develop prompts following the SQUAD format for the T5's pretraining (see Appendix A.1). The output takes the form of "*answer* `because` *explanation*". There is another factor to consider. We need to decide whether to add *tags*—a single descriptions of different input elements; e.g., "`premise:`" and "`hypothesis:`" before the first and second sentence in the E-SNLI input. Without these tags the task seems impossible to understand, but UNIFIEDQA has not been trained with similar tags.

Table 11 in Appendix shows examples of our various QA prompts.

Results We present the results of UNIFIEDQA with $\text{QA}_{\text{SIMPLE}}$ in Table 2, and due to space limits, T5's results with SQUAD_{T5} prompts in Table 10 in Appendix. For ECQA with UNIFIEDQA, we use the UNIFIEDQA format for multiple-choice QA (see Appendix A.1).

We observe that for E-SNLI and COMVE it is crucial to add tags ("`premise:`"/"`hypothesis:`"; "`choice1:`"/"`choice2:`"). This result is intuitive—it should be difficult to pick one of the two sentences, or classify a relation between them, if sentences are not marked.[5] On the other hand, adding label choices is not beneficial and in some cases can even decrease the performance. When tags are included, we see that across all the tasks the "What is...?" question performs the best. This also holds for T5 and SQUAD_{T5} prompts (see Table 10). Finally, the prompt with the "What is...?" question

[5]Performance on COMVE with "Is...?" is close to random which suggests that this question form hinders the performance and tags cannot make a difference.

and tags in the input outperforms UNIFEW for both tasks UNIFEW can be applied to. This result shows that this prompt is both versatile and effective.

Finally, we compare the best performing prompts we get with UNIFIEDQA + QA_{SIMPLE} and T5+ $SQUAD_{T5}$. See prompts " $SQUAD_{T5}$ × WHAT IS...? + TAGS" and " QA_{SIMPLE} × WHAT IS...? + TAGS" in Table 11. For ECQA and COMVE, we observe notable improvements from using UNIFIEDQA, and minor improvements for SBIC. For E-SNLI, T5 is better, presumably because UNIFIEDQA has lost some useful information from MNLI after extensive continued pretraining for QA. These results suggest that UNIFIEDQA is a better model for prompting self-rationalization with QA prompts.

To recap, the analysis presented in this section suggests that QA prompting for inducing self-rationalization behavior is best done when UNIFIEDQA is combined with the NL prompt below. For true QA tasks, we use UNIFIEDQA formats.

```
explain what is this/more...? \\n tag_1: [sequence_1] tag_2: [sequence_2] ...</s>
```

3.2 INFILLING Prompts

The simplest way to design an infilling prompt is to prepend the span "<extra_id_0> because <extra_id_1>" to the input. A model should then replace <extra_id_0> with a label/answer and <extra_id_1> with an explanation. Besides being similar to T5's span infilling pretraining task, another benefit of this prompt is that is very flexible—the span above can be added to any task input. This basic infilling prompt could be easily made more natural by prepending phrases such as: "The answer is" (ECQA), "Less common is" (COMVE), or "This is" (E-SNLI, SBIC). We hypothesize that these additional phrases could be beneficial because they suggest which subset of the vocabulary is the right word for filling in <extra_id_0>. We test whether it is beneficial to make the infilling prompt more natural sounding.

Results T5 results are shown in Table 3. The outcome is mixed—while we observe notable benefits for ECQA/SBIC, for E-SNLI/COMVE there is a minor difference in favor of the basic prompt. A way to explain this is that T5 learned about NLI labels from MNLI during pretraining, so it does not an need additional phrase to nudge it in the right direction. COMVE results are comparable to the random performance, and the model could not learn

	Prompt	Accuracy	BERTscore
E-SNLI	UNIFEW	$61.68_{0.58}$	$55.85_{0.53}$
	+ tags	$63.61_{0.44}$	$57.34_{0.41}$
	Is...?	$47.47_{0.52}$	$42.70_{0.47}$
	+ tags	$66.59_{0.51}$	$60.05_{0.47}$
	+ tags & choices	$64.43_{0.53}$	$58.16_{0.49}$
	What is...?	$40.67_{0.44}$	$36.50_{0.40}$
	+ tags	$\mathbf{75.05}_{0.34}$	$\mathbf{67.52}_{0.33}$
	+ tags & choices	$69.28_{0.68}$	$62.46_{0.62}$
	RANDOM BASELINE	33.33	-
ECQA	UNIFIEDQA	$\mathbf{41.37}_{0.34}$	$\mathbf{36.72}_{0.30}$
	RANDOM BASELINE	20.00	-
ComVE	Is...?	$52.69_{0.35}$	$47.70_{0.31}$
	+ tags	$52.47_{0.32}$	$47.47_{0.30}$
	+ tags & choices	$52.19_{0.33}$	$47.27_{0.30}$
	What is...?	$50.60_{0.22}$	$45.68_{0.20}$
	+ tags	$\mathbf{67.33}_{0.71}$	$\mathbf{60.97}_{0.64}$
	+ tags & choices	$62.56_{0.65}$	$56.68_{0.59}$
	RANDOM BASELINE	50.00	-
SBIC	UNIFEW	$66.15_{0.43}$	$63.84_{0.44}$
	Is...?	$63.50_{0.44}$	$61.21_{0.42}$
	+ tags	$62.64_{0.45}$	$60.43_{0.45}$
	+ tags & choices	$63.63_{0.42}$	$61.31_{0.43}$
	What is...?	$67.35_{0.38}$	$65.03_{0.37}$
	+ tags	$\mathbf{67.55}_{0.41}$	$\mathbf{65.29}_{0.39}$
	+ tags & choices	$65.43_{0.58}$	$63.07_{0.59}$
	RANDOM BASELINE	50.00	-

Table 2: Prompting UNIFIEDQA with QA_{SIMPLE} with "Is...?" and "What is...?" questions, and UNIFEW. See §3.1 for descriptions of these prompts. For ECQA we use the original UNIFIEDQA format for multiple-choice QA. We also inspect the effects of adding label choices and *tags* (defined in §3.1) to the input.

	E-SNLI	ECQA	COMVE	SBIC
B	$\mathbf{75.24}_{0.38}$	$22.33_{0.29}$	$\mathbf{50.36}_{0.31}$	$61.57_{0.45}$
N	$75.09_{0.45}$	$\mathbf{27.60}_{0.36}$	$49.02_{0.28}$	$\mathbf{64.66}_{0.52}$
	E-SNLI	ECQA	COMVE	SBIC
B	$\mathbf{67.66}_{0.35}$	$19.83_{0.26}$	$\mathbf{45.51}_{0.28}$	$59.18_{0.46}$
N	$67.52_{0.42}$	$\mathbf{24.52}_{0.32}$	$44.35_{0.26}$	$\mathbf{62.00}_{0.54}$

Table 3: A comparison of the basic infilling prompt (B) with its more natural sounding version (N). The upper part shows accuracy and the lower part BERTscore.

	Task	Accuracy	BERTscore
INFILLING	E-SNLI	$75.09_{0.45}$	$67.52_{0.42}$
	ECQA	$27.60_{0.36}$	$24.52_{0.32}$
	COMVE	$49.02_{0.28}$	$44.35_{0.26}$
	SBIC	$64.66_{0.52}$	$62.00_{0.54}$
	Average	54.09	49.57
≈T5	E-SNLI	**$79.21_{0.29}$**	**$71.34_{0.27}$**
	ECQA	$38.28_{0.33}$	$33.91_{0.29}$
	COMVE	$55.88_{0.34}$	$50.45_{0.30}$
	SBIC	$65.06_{0.60}$	$62.77_{0.63}$
	Average	59.61	54.62
QA_{SIMPLE}	E-SNLI	$75.05_{0.34}$	$67.52_{0.33}$
	ECQA	**$41.37_{0.34}$**	**$36.72_{0.30}$**
	COMVE	**$67.33_{0.71}$**	**$60.97_{0.64}$**
	SBIC	**$67.55_{0.41}$**	**$65.29_{0.39}$**
	Average	**62.82**	**57.63**

Table 4: A comparison between three prompt types: INFILLING, ≈T5, and QA_{SIMPLE} prompts. See §3 for descriptions of these prompts.

the task from the infilling prompt, with or without the additional phrases. Thus, we recommend using the more natural version as it is not detrimental to E-SNLI/COMVE performance while it leads to big improvements for ECQA/SBIC.

3.3 INFILLING vs. ≈T5 vs. QA

We have established appropriate QA and INFILLING prompts in §3.1 and §3.2. We now turn to a comparison between all three prompt types: (i) INFILLING (natural), (ii) ≈T5, and (iii) QA_{SIMPLE} ("What is...?" with tags). The first two are used to prompt T5 and the last type UNIFIEDQA. To construct ≈T5 prompts, for each task in FEB, we identify the most similar T5's pretraining task (see Table 6, Appendix) and use that task's formatting (see, e.g., ≈T5 × COPA in Table 11).

Results A comparison of the three prompt types is presented in Table 4. The QA_{SIMPLE} prompt outperforms other prompt types for all tasks except E-SNLI for which unsurprisingly ≈T5 is the best. Finally, this brings us to the end of our extensive exploration of natural language prompts for a prompt-based finetuning approach to few-shot self-rationalization. We identify the QA_{SIMPLE} prompt as the most effective and we use it to study how few-shot self-rationalization performance scales with the size of the UNIFIEDQA model.

4 Improving Self-Rationalization with Increasing Model Size

In §3, we discovered that a QA prompt combined with the base UNIFIEDQA model version is as an effective combination for few-shot self-rationalization through prompt-based finetuning. In this section, we provide two additional evaluations to establish the first approach to few-shot self-rationalization.

First, we assess how plausible the generated explanations are when evaluated by annotators on Amazon MTurk. Details of how we conduct human evaluation of plausibility are given in §2. One HIT contains 10 instances and we pay $1 per HIT.

Next, we investigate how self-rationalization performance changes with the model size since larger pretrained language models typically give better few-shot performance (Brown et al., 2020). We wonder whether the same trend will hold for a complex generation task of self-rationalization where it is conceivable that an enormous model could overfit on a few examples. To this end, we evaluate three versions of UNIFIEDQA (BASE, LARGE, 3B) and GPT-3 (Brown et al., 2020).

Experimental Setup We evaluate GPT-3 using its API and "in-context demonstrations" (Brown et al., 2020). We pack as many demonstrations as we can fit in the input, followed by the input of the test example, then run GPT-3 to generate its output. The number of demonstrations we are able to fit ranges from [28,45] which are randomly selected from the 48 used for UNIFIEDQA. Since evaluation using a single prompt costs us $1,050, we do not do prompt search for GPT-3. We use the prompts shown in Fig. 1 in Appendix.

A detailed description of evaluation metrics is given in §2. Each dev set size for GPT-3 is 18 instead of 350 (because of the API cost). Ground-truth explanations are evaluated together with explanations generated by 4 models. Therefore, for GOLD explanations, we report the average of 4 plausibility scores, std. errors, and κ values calculated with 4 Mturk batches (corresponding to 4 models).

4.1 Results

Results are shown in Table 5. Note that we use T5 with the ≈T5 prompt for E-SNLI, and UNIFIEDQA with QA_{SIMPLE} (§3) for other datasets to establish the best possible performance for each dataset. The exact prompts for each task

	Model	# Par.	Accuracy	BERTscore	Plausibility *All* Score	*All* κ	*Label*$_1$ Score	*Label*$_1$ κ	*Label*$_2$ Score	*Label*$_2$ κ	*Label*$_3$ Score	*Label*$_3$ κ
E-SNLI	BASE	220M	$79.21_{0.29}$	$71.34_{0.27}$	$16.75_{1.53}$	0.73	$15.65_{2.34}$	0.67	$17.50_{2.88}$	0.79	$17.13_{2.71}$	0.72
	LARGE	770M	$84.79_{0.27}$	$76.56_{0.27}$	$32.68_{1.92}$	0.57	**$27.31_{2.88}$**	0.43	$33.89_{3.44}$	0.64	$36.85_{3.58}$	0.64
	3B	2.8B	**$87.43_{0.23}$**	**$79.10_{0.23}$**	$41.60_{2.08}$	0.62	$27.13_{2.85}$	0.52	$46.76_{3.84}$	0.70	**$50.92_{3.63}$**	0.64
	GPT-3	175B	$65.37_{0.53}$	$59.83_{0.47}$	**$42.44_{2.17}$**	0.54	**$27.31_{2.87}$**	0.48	**$66.03_{4.37}$**	0.71	$43.80_{3.46}$	0.51
	GOLD	-			$77.40_{1.59}$	0.63	$63.50_{3.01}$	0.44	$87.87_{1.85}$	0.74	$82.48_{2.42}$	0.72
	RAND	-	33.33									
ECQA	BASE	220M	$41.37_{0.34}$	$36.72_{0.30}$	$25.52_{1.25}$	0.32						
	LARGE	770M	$57.19_{0.36}$	$51.00_{0.32}$	$30.28_{1.53}$	0.38						
	3B	2.8B	**$65.86_{0.36}$**	**$58.98_{0.32}$**	$34.23_{1.56}$	0.35						
	GPT-3	175B	$60.65_{1.48}$	$54.42_{1.32}$	**$45.06_{1.44}$**	0.12						
	GOLD	-			$70.88_{1.47}$	0.45						
	RAND	-	20.00									
COMVE	BASE	220M	$67.33_{0.71}$	$60.97_{0.64}$	$13.80_{1.26}$	0.45						
	LARGE	770 M	$81.31_{0.39}$	$73.95_{0.36}$	$25.59_{1.67}$	0.52						
	3B	2.8B	**$88.96_{0.38}$**	**$81.02_{0.34}$**	$33.40_{1.71}$	0.63						
	GPT-3	175B	$73.98_{1.40}$	$67.65_{1.29}$	**$42.16_{1.80}$**	0.73						
	GOLD	-			$77.24_{1.30}$	0.55						
	RAND	-	50.00									
SBIC	BASE	220M	$67.55_{0.41}$	$65.29_{0.39}$	$57.96_{2.25}$	0.68	$21.36_{2.06}$	0.54	$94.57_{1.08}$	0.82		
	LARGE	770M	$71.06_{0.39}$	$68.55_{0.39}$	$61.82_{2.23}$	0.66	$27.16_{2.19}$	0.43	**$96.48_{0.92}$**	0.89		
	3B	2.8B	$71.66_{0.48}$	$68.90_{0.49}$	$64.20_{2.14}$	0.68	$33.76_{2.65}$	0.55	$94.63_{1.02}$	0.81		
	GPT-3	175B	**$74.17_{1.41}$**	**$71.53_{1.40}$**	**$72.68_{1.72}$**	0.53	**$52.65_{2.51}$**	0.34	$92.72_{1.05}$	0.72		
	GOLD	-			$79.81_{1.62}$	0.67	$64.92_{2.66}$	0.52	$94.69_{1.01}$	0.81		
	RAND	-	50.00									

Table 5: The first results on the FEB benchmark using T5/UNIFIEDQA (BASE, LARGE, 3B) and GPT-3. T5+≈T5 prompt is used only for E-SNLI, and UNIFIEDQA + $\text{QA}_{\text{SIMPLE}}$ prompt is used for other datasets. The descriptions of these prompts are given in §3 and details of how evaluation metrics are calculated in §2. RAND stands for a random baseline and GOLD for human-authored explanations. *Label*$_1$/*Label*$_2$/*Label*$_3$ are entailment/neutral/contradiction in E-SNLI and offensive/not offensive in SBIC.

are given in Appendix A.2. We observe that all metrics—accuracy, BERTscore, and plausibility—monotonically increase with the size of UNIFIEDQA for all datasets. That is, larger models learn to predict task labels and generate explanations from a few examples better, despite being overparametrized. UNIFIEDQA-3B has a higher accuracy/BERTscore than GPT-3 for all datasets except SBIC, but GPT-3 generates explanations that are notably more plausible.

The following observations suggest that few-shot self-rationalization is a promising research direction. The difference in plausibility scores between the BASE and 3B model versions ranges from [6.24, 24.85] (on average 14.85). In other words, since it is possible to generate more plausible explanation by only increasing the model size, it is conceivable that further progress could be made with more creative approaches. Next, the plausibility score of the best model (GPT-3) ranges from [42.16, 72.68] ([42.16, 52.65] if we consider only SBIC "offensive" (*Label*$_1$) subset. This shows that a moderate plausibility can already be achieved with current models without any task-specific enhancements.

Despite that, the gap between our best models and human-authored explanations remains large. The average plausibility score across datasets is 43.36 (UNIFIEDQA-3B), 50.58 (GPT-3), and 76.33 (GOLD). In other words, the difference in plausibility scores between UNIFIEDQA-3B's and human explanations is 32.98, and between GPT-3's and human explanations is 25.75. We expect that the FEB benchmark, our UNIFIEDQA approach, and first results, present a good starting point to tackle this challenge.

Performance w.r.t. Labels For E-SNLI and SBIC, we can inspect the metrics with respect to labels. In E-SNLI part of the Table 5, *Label*$_1$ marks "entailment", *Label*$_2$ "neutral", and *Label*$_3$ "contradiction". There are notable differences between the plausibility scores for each label. The plausibility score for "entailment" does not scale with the model size and it is much lower than scores for other labels (the best score is 27.31 vs. 66.0/50.92). This issue stems from the difficulty of explaining the entailment label (Camburu et al., 2018). Even people struggle with explaining "entailment" as evident by the lower GOLD score for "entailment" compared to the other two labels. An interesting observation from the other two labels is that UNIFIEDQA-3B is explains "contradiction" instances best and GPT-3 "neutral" instances.

In SBIC part of the Table 5, $Label_1$ marks "offensive" and $Label_2$ "not offensive" instances. The latter achieve almost perfect plausibility since the models learn to generated "*This post does not imply anything offensive*". Thus, main plausibility scores for SBIC are those of offensive instances. We can observe that the relative differences between models for offensive instances are much larger than the relative differences when examples of both labels are counted for (column "*All* / Score"). If we had only looked into a single plausibility score we would not notice these differences. Thus, we recommend breaking down the performance w.r.t. labels whenever possible.

Annotator Agreement Finally, we observe challenges in collecting human judgments of plausibility. For all datasets except ECQA, Fleiss' κ is either moderate (between 0.41–0.6) or substantial (between 0.61–0.8). One exception is GPT-3 on SBIC ($Label_1$; offensive) where κ is only 0.34. We also observe that κ for GPT-3's explanations is lower than κ for UNIFIEDQA's or GOLD explanations, with the exception of COMVE. The most concerning is ECQA where κ is on average 0.35 for UNIFIEDQA's explanations, 0.34 for GOLD explanations, and only 0.12 for GPT-3's. Future work should investigate the reasons behind these differences more carefully.

5 Related Work

Self-Rationalization with Few Human-Written Explanations Select-then-predict method (Lei et al., 2016) that is standard to creating explanations in the form of *highlights* of the input tokens does not use any human-author highlighting explanations. On the other hand, a standard approach to generating free-text explanations is to use human-written explanations (Liu et al., 2019; Wu and Mooney, 2019; Narang et al., 2020, among others). To the best of our knowledge, only two prior works generate *free-text* explanations in a weakly-supervised way from the task prediction loss. Latcinnik and Berant (2020) approach commonsense QA in that fashion. Brahman et al. (2021) propose multiple distant supervision approach to explaining a defeasible inference task. In this paper, we introduce the FEB benchmark to unify the evaluation of few-shot self-rationalization and present the first approach and results on FEB.

Few-Shot Learning We study natural language prompts (Brown et al., 2020; Schick and Schütze, 2021) to establish the first approach to few-shot self-rationalization. Alternatively, few-shot learning researchers are studying prompts in the form of continuous/soft vectors that do not correspond to real tokens (e.g., Qin and Eisner, 2021). Such methods present a promising research direction for few-shot self-rationalization. Namely, we show that larger models generate notably more plausible explanations, and "prefix tuning" (Li and Liang, 2021) has been show to learn two condition generation tasks using only 0.1% of the parameters, while maintaining comparable performance. In practice, such approaches still require a notable amount of GPU memory. Thus, any efforts to reduce required memory such as compression (Ganesh et al., 2021) may be valuable for few-shot self-rationalization.

6 Conclusions

We draw attention to the task of few-shot self-rationalization: predicting task labels and generating *free-text* explanations for the prediction using only a few human-written explanations. We present (i) the FEB benchmark, (ii) the first prompting approach for FEB established through a comprehensive search of natural language prompts, and (iii) results using models with a number of parameters ranging from 220M to 175B. Our human evaluation results show that progress is possible on this task given that just scaling the model size increases both the plausibility of generated explanations and task accuracy by a very large margin. Despite that, few-shot self-rationalization remains very challenging, with plausibility of explanations generated by the best model being 27.75 points behind plausibility of human-authored explanations. We hope that work presented in this paper spurs the community to work on this challenging problem to enable more intuitive interaction with NLP systems.

References

Shourya Aggarwal, Divyanshu Mandowara, Vishwajeet Agrawal, Dinesh Khandelwal, Parag Singla, and Dinesh Garg. 2021. Explanations for CommonsenseQA: New Dataset and Models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3050–3065, Online. Association for Computational Linguistics.

Jonathan Bragg, Arman Cohan, Kyle Lo, and Iz Beltagy. 2021. Flex: Unifying evaluation for few-shot nlp. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

Faeze Brahman, Vered Shwartz, Rachel Rudinger, and Yejin Choi. 2021. Learning to rationalize for nonmonotonic reasoning with distant supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

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

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-SNLI: Natural language inference with natural language explanations. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Prakhar Ganesh, Yao Chen, Xin Lou, Mohammad Ali Khan, Yin Yang, Hassan Sajjad, Preslav Nakov, Deming Chen, and Marianne Winslett. 2021. Compressing Large-Scale Transformer-Based Models: A Case Study on BERT. *Transactions of the Association for Computational Linguistics*, 9:1061–1080.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

Matt Gardner, Jonathan Berant, Hannaneh Hajishirzi, Alon Talmor, and Sewon Min. 2019. Question answering is a format; when is it useful? arXiv:1909.11291.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *The International Conference on Learning Representations (ICLR)*.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeff Wu, and Dario Amodei. 2020. Scaling laws for neural language models. arXiv:2001.08361.

Maxime Kayser, Oana-Maria Camburu, Leonard Salewski, Cornelius Emde, Virginie Do, Zeynep Akata, and Thomas Lukasiewicz. 2021. e-vil: A dataset and benchmark for natural language explanations in vision-language tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.

Najoung Kim, Ellie Pavlick, Burcu Karagol Ayan, and Deepak Ramachandran. 2021. Which linguist invented the lightbulb? presupposition verification for question-answering. arXiv:2101.00391.

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

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

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Hui Liu, Qingyu Yin, and William Yang Wang. 2019. Towards explainable NLP: A generative explanation framework for text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5570–5581, Florence, Italy. Association for Computational Linguistics.

Ana Marasović, Chandra Bhagavatula, Jae sung Park, Ronan Le Bras, Noah A. Smith, and Yejin Choi. 2020. Natural language rationales with full-stack visual reasoning: From pixels to semantic frames to commonsense graphs. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2810–2829, Online. Association for Computational Linguistics.

David Alvarez Melis and Tommi Jaakkola. 2018. Towards robust interpretability with self-explaining neural networks. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

Sharan Narang, Hyung Won Chung, Yi Tay, Liam Fedus, Thibault Fevry, Michael Matena, Karishma Malkan, Noah Fiedel, Noam Shazeer, Zhenzhong Lan, Yanqi Zhou, Wei Li, Nan Ding, Jake Marcus, Adam Roberts, and Colin Raffel. 2021. Do transformer modifications transfer across implementations and applications? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5758–5773, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. 2020. WT5?! Training Text-to-Text Models to Explain their Predictions. arXiv:2004.14546.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5203–5212, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI Spring Symposium Series*.

Cynthia Rudin, Chaofan Chen, Zhi Chen, Haiyang Huang, Lesia Semenova, and Chudi Zhong. 2021. Interpretable machine learning: Fundamental principles and 10 grand challenges. arXiv: 2103.11251.

Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A. Smith, and Yejin Choi. 2020. Social bias frames: Reasoning about social and power implications of language. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5477–5490, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Cunxiang Wang, Shuailong Liang, Yue Zhang, Xiaonan Li, and Tian Gao. 2019. Does it make sense? and why? a pilot study for sense making and explanation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4020–4026, Florence, Italy. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Sarah Wiegreffe and Ana Marasović. 2021. Teach me to explain: A review of datasets for explainable nlp. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

Sarah Wiegreffe, Ana Marasović, and Noah A. Smith. 2021. Measuring association between labels and free-text rationales. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10266–10284, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Jialin Wu and Raymond Mooney. 2019. Faithful multimodal explanation for visual question answering. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 103–112, Florence, Italy. Association for Computational Linguistics.

Fan Yang, Mengnan Du, and Xia Hu. 2019. Evaluating explanation without ground truth in interpretable machine learning. 1907.06831.

Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. 2018. Record: Bridging the gap between human and machine commonsense reading comprehension. arXiv:1810.12885.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *The International Conference on Learning Representations (ICLR)*.

A Appendix

A.1 Input Formats in Related Work

T5 with SQuAD

- `question:` [*question*] `context:` [*paragraph*]

UnifiedQA

- **Multiple-choice QA:** [*question*] `\\n (A)` [*choice*$_1$] `(B)` [*choice*$_2$]...
- **Extractive QA:** [*question*] `\\n` [*paragraph*]

UniFew

- **Single text classification:** `Topic? \\n (A)` [*class*$_1$] `(B)` [*class*$_2$] `(C)` [*class*$_3$] \\n [*document*]
- **Sentence-pair classification:** [*sentence*$_1$] `Is` [*sentence*$_1$] `? \\n (A) Yes (B) No (C) Maybe`
- **Relation classification:** [*mention*$_1$] `to` [*mention*$_1$] `? \\n (A)` [*class*$_1$] `(B)` [*class*$_2$] `(C)` [*class*$_3$]
- **Entity recognition:** `What is the type of the entity between the # marks? \\n` [*text... #mention*$_1$*# ...text... #mention2# ...text*]

A.2 Prompts Used in §4

UnifiedQA/T5 prompts:

- **E-SNLI** `explai nli hypothesis:` [*hypothesis*] `premise:` [*premise*]
- **ECQA:** `explain` [*question*] `\\n (A)` [*choice*$_1$] `(B)` [*choice*$_2$] `(C)` [*choice*$_3$] `(D)` [*choice*$_4$] `(E)` [*choice*$_5$]
- **ComVE:** `explain what is more nonsensical? \\n choice1:` [*sentence*$_1$] `choice2:` [*sentence*$_2$]
- **SBIC:** `explain what is this? \\n post:` [*post*]

Prompts for GPT-3 are given in Figure 1. The output always takes the form of "[*answer/label*] `because` [*explanation*]".

```
=============== SBIC ===============
Is the following post offensive? and why?
Post: [Input]
Answer: Yes|No
Reason: [Explanation]

============== E-SNLI ==============
Answer the question and provide a reason why
the answer is correct.
[Premise]
Question: Is [Hypothesis]?
Answer: Yes|No|Maybe
Reason: [Explanation]

============== ECQA ==============
Answer the question from the provided
choices, and provide a reason why the answer
is correct.
Question: [Question]
Choices: [Choices]
Answer: [one of the choices]
Reason: [Explanation]

============== ComVE ==============
Which of the two choices makes more sense?
and why?
Choice1: [Choice1]
Choice2: [Choice2]
Answer: Choice1|Choice2
Reason: [Explanation]
```

Figure 1: GPT-3 prompt templates for all datasets.

FEB Task	Similar T5 Pretraining Tasks	
E-SNLI	MNLI (Williams et al., 2018)	Classify the entailment relation between two sequences
ECQA	RECORD (Zhang et al., 2018)	Answer a cloze-style query about a passage given entities in it
ComVE	COPA (Roemmele et al., 2011)	Select one of two sequences as the cause/effect of a premise
SBIC	COLA (Warstadt et al., 2019)	Classify a sentence as acceptable or not

Table 6: The first column shows tasks that we have included in FEB. Tasks on the right are included in T5's pretraining and they are similar to FEB's tasks. We explore self-rationalization prompts for FEB's tasks based on the tasks on the right, and compare them to prompts designed as span infilling and QA (§3).

GPUs	NVIDIA A100 on Google Cloud
Implementation	*Will be added upon acceptance.*

Hyperparameter	**Assignment**
max step number	300
batch size	4 (1 for UNIFIEDQA-3B)
gradient accumulation steps	1 (4 for UNIFIEDQA-3B)
learning rate	3e-5
learning rate scheduler	linear
warmup steps	0
decoding	greedy

Table 7: Hyperparameters used in our experiments.

		Accuracy	**BERTscore**
COS-E	INFILLING (b)	$34.28_{0.36}$	$29.60_{0.32}$
	INFILLING (n)	$40.14_{0.38}$	$34.70_{0.34}$
	≈T5	$51.69_{0.41}$	$44.56_{0.36}$
	SQUAD$_{T5}$	$51.15_{0.34}$	$44.13_{0.29}$
	QA$_{SIMPLE}$	**59.96**$_{0.32}$	**48.57**$_{0.26}$

Table 8: A comparison of all prompt types introduced in §3 on COS-E. We do not support using COS-E in the future given the reported issues with it (Narang et al., 2020; Wiegreffe and Marasović, 2021), especially since ECQA is introduced.

	Size	**Accuracy**	**BERTscore**
COS-E	BASE	$58.32_{0.28}$	$50.43_{0.25}$
	LARGE	$69.44_{0.30}$	$60.11_{0.26}$
	3B	**75.37**$_{0.31}$	**65.34**$_{0.28}$
	GPT-3	$68.43_{1.35}$	$59.48_{1.18}$

Table 9: The effect of scaling the UNIFIEDQA model size on self-rationalization of COS-E. We do not support using COS-E in the future given the reported issues with it (Narang et al., 2020; Wiegreffe and Marasović, 2021), especially since ECQA is introduced.

	Prompt	**Accuracy**	**BERTscore**
E-SNLI	Is...?	$38.68_{0.44}$	$34.74_{0.40}$
	+ tags	$48.20_{0.62}$	$43.22_{0.58}$
	What is...?	$60.76_{0.85}$	$54.75_{0.77}$
	+ tags	**77.86**$_{0.34}$	**70.08**$_{0.32}$
ECQA	SQUAD$_{T5}$	**36.48**$_{0.34}$	**32.38**$_{0.30}$
	RANDOM BASELINE	20.00	-
COMVE	Is...?	$50.38_{0.16}$	$45.54_{0.14}$
	+ tags	$50.17_{0.14}$	$45.35_{0.13}$
	What is...?	$50.54_{0.21}$	$45.67_{0.19}$
	+ tags	**54.49**$_{0.46}$	**49.25**$_{0.42}$
SBIC	Is...?	$63.37_{0.58}$	$61.15_{0.57}$
	+ tags	$63.82_{0.54}$	$61.69_{0.55}$
	What is...?	$66.67_{0.49}$	$64.33_{0.51}$
	+ tags	**66.99**$_{0.53}$	**64.60**$_{0.56}$

Table 10: A comparison between SQUAD$_{T5}$ prompts with "Is...?" and "What is...?" questions. See §3.1 for more info. We also inspect the effects of adding answer choices and *tags* to the input. Tags are a single word descriptions of the input elements; e.g., E-SNLI's tags are "premise:" / "hypothesis:" before premise / hypothesis.

Sentence1: The stove was cleaned with a cleaner. **Sentence2:** The stove was cleaned with a mop.
Nonsensical Sentence: Sentence2 **Explanation:** A mop is too large to clean the stove.

Prompt: INFILLING × BASIC
Input: explain sensemaking choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.* <extra_id_0> because <extra_id_1>
Output: <extra_id_0> *choice2* <extra_id_1> *A mop is too large to clean the stove.* <extra_id_2>

Prompt: INFILLING × NATURAL SOUNDING
Input: explain sensemaking choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.* It is <extra_id_0> that choice2 is less common because <extra_id_1>
Output: <extra_id_0> *True* <extra_id_1> *A mop is too large to clean the stove.* <extra_id_2>

Prompt: ≈T5 × COPA
Input: explain sensemaking choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.* Less common is choice2
Output: *True* because *a mop is too large to clean the stove.*

Prompt: SQuAD_{T5} × YES/NO + TAGS
Input: explain sensemaking question: Is choice2 more nonsensical? context: choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.*
Output: *Yes* because *a mop is too large to clean the stove.*

Prompt: SQuAD_{T5} × WHAT IS...? + TAGS
Input: explain sensemaking question: What is more nonsensical? context: choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.*
Output: *choice2* because *a mop is too large to clean the stove.*

Prompt: $\text{QA}_{\text{SIMPLE}}$ × YES/NO
Input: explain is choice2 more nonsensical? \\n *The stove was cleaned with a cleaner. The stove was cleaned with a mop.*</s>
Output: *yes* because *a mop is too large to clean the stove.*

Prompt: $\text{QA}_{\text{SIMPLE}}$ × YES/NO + TAGS
Input: explain is choice2 more nonsensical? \\n choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.*</s>
Output: *yes* because *a mop is too large to clean the stove.*

Prompt: $\text{QA}_{\text{SIMPLE}}$ × YES/NO + TAGS + CHOICES
Input: explain is choice2 more nonsensical? \\n (A) yes (B) no \\n choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.*</s>
Output: *yes* because *a mop is too large to clean the stove.*

Prompt: $\text{QA}_{\text{SIMPLE}}$ × WHAT IS...?
Input: explain what is more nonsensical? \\n *The stove was cleaned with a cleaner. The stove was cleaned with a mop.*</s>
Output: *choice2* because *a mop is too large to clean the stove.*

Prompt: $\text{QA}_{\text{SIMPLE}}$ × WHAT IS...? + TAGS
Input: explain what is more nonsensical? \\n choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.*</s>
Output: *choice2* because *a mop is too large to clean the stove.*

Prompt: $\text{QA}_{\text{SIMPLE}}$ × WHAT IS...? + TAGS + CHOICES
Input: explain what is more nonsensical? \\n (A) choice1 (B) choice2 \\n choice1: *The stove was cleaned with a cleaner.* choice2: *The stove was cleaned with a mop.*</s>
Output: *choice2* because *a mop is too large to clean the stove.*

Table 11: COMVE self-rationalization prompts that we design and test. INFILLING marks span-filling prompts; ≈T5 prompts made by following the most similar T5 pretraining task (Table 1); SQuAD_{T5} prompts designed following SQUAD's formatting in T5 pretraining; and $\text{QA}_{\text{SIMPLE}}$ prompts made following UNIFIEDQA. This table shows variations of these prompt types. We refer to spans "choice1:"/"choice2:" as TAGS, and to "(A) yes (B) no"/"(A) choice1 (B) choice2" as CHOICES. YES/NO and WHAT IS...? refer to a question type. Following Hendrycks et al. (2021), we add </s> to the end of our $\text{QA}_{\text{SIMPLE}}$ prompts. More info in §3.