LARGE LANGUAGE MODELS CAN SELF-IMPROVE

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) have achieved excellent performances in various tasks. However, fine-tuning an LLM requires extensive supervision. Human, on the other hand, may improve their reasoning abilities by self-thinking without external inputs. In this work, we demonstrate that an LLM is also capable of self-improving with only unlabeled datasets. We use a pre-trained LLM to generate "high-confidence" rationale-augmented answers for unlabeled questions using Chain-of-Thought prompting and self-consistency, and fine-tune the LLM using those self-generated solutions as target outputs. We show that our approach improves the general reasoning ability of a 540B-parameter LLM (74.4%→82.1% on GSM8K, 78.2%→83.0% on DROP, 90.0%→94.4% on OpenBookQA, and 63.4%→67.9% on ANLI-A3) and achieves state-of-the-art-level performance, without any ground truth label. We conduct ablation studies and show that fine-tuning on reasoning is critical for self-improvement.

1 INTRODUCTION

Scaling has enabled Large Language Models (LLMs) to achieve state-of-the-art performance on a range of Natural Language Processing (NLP) tasks (Wang et al., 2018; 2019; Rajpurkar et al., 2016). More importantly, new capabilities have emerged from LLMs as they are scaled to hundreds of billions of parameters (Wei et al., 2022a): in-context few-shot learning (Brown et al., 2020) makes it possible for an LLM to perform well on a task it never trained on with only a handful of examples; Chain-of-Thought (CoT) prompting (Wei et al., 2022b; Kojima et al., 2022) demonstrates strong reasoning ability of LLMs across diverse tasks with or without few-shot examples; self-consistency (Wang et al., 2022b) further improves the performance via self-evaluating multiple reasoning paths.

Despite these incredible capabilities of models trained on large text corpus (Brown et al., 2020; Chowdhery et al., 2022), fundamentally improving the model performances beyond few-shot baselines still requires finetuning on an extensive amount of *high-quality supervised* datasets. FLAN (Wei et al., 2021; Chung et al., 2022) and T0 (Sanh et al., 2022) curated tens of benchmark NLP datasets to boost zero-shot task performances on unseen tasks; InstructGPT (Ouyang et al., 2022) crowd-sourced many human answers for diverse sets of text instructions to better align their model to human instructions. While significant efforts were committed on collecting high-quality supervised datasets, human brain, on the contrary, is capable of the metacognition process (Dunlosky & Metcalfe, 2008), where we can refine our own reasoning ability without external inputs.

In this paper, we study how an LLM capable of in-context few-shot learning and chain-of-thought reasoning, is able to *self-improve* its reasoning ability without supervised data. We show that using only input sequences (without ground truth output sequences) from multiple NLP task datasets, a pre-trained LLM is able to improve performances for both in-domain and out-of-domain tasks. Our method is shown in Figure 1: we first sample multiple predictions using few-shot Chain-of-Thought (CoT) (Wei et al., 2022b) as prompts, filter "high-confidence" predictions using majority voting (Wang et al., 2022b), and finally finetune the LLM on these high-confidence predictions. The resulting model shows improved reasoning in both greedy and multi-path evaluations. We call the model fine-tuned in this way as **Language Model Self-Improved (LMSI)**. Note that LMSI depends on in-context few-shot learning and chain-of-thought reasoning abilities which small language models do not necessarily have. We empirically verify **LMSI** using a pre-trained 540B LLM, where our method not only improves training task performances (74.4%→82.1% on GSM8K, 78.2%→83.0% on DROP, 90.0%→94.4% on OpenBookQA, and 63.4%→67.9% on ANLI-A3), but also enhances out-of-domain (OOD) test tasks (AQUA, StrategyQA, MNLI), achieving state-of-the-art perfor-

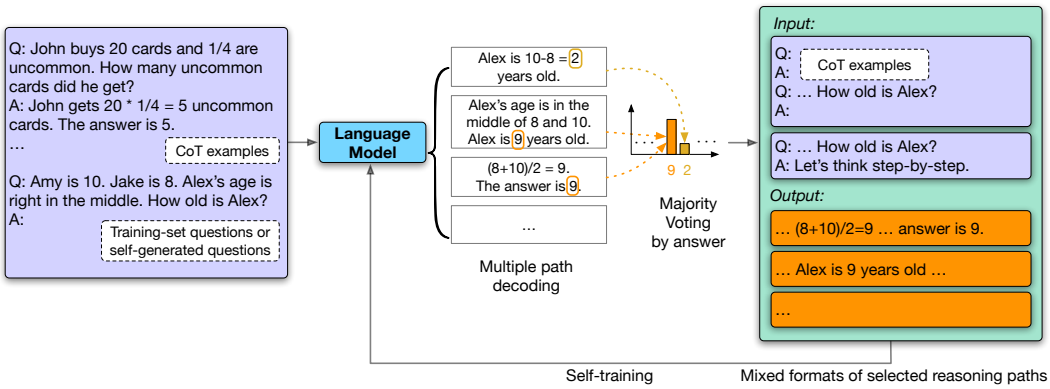


Figure 1: Overview of our method. With Chain-of-Thought (CoT) examples as demonstration (Wei et al., 2022b), the language model generates multiple CoT reasoning paths and answers (temperature $T > 0$) for each question. The most consistent answer is selected by majority voting (Wang et al., 2022b). The "high-confidence" CoT reasoning paths that lead to the majority answer are augmented by mixed formats as the final training samples to be fed back to the model for fine-tuning.

mances in many tasks without relying on supervised ground truth answers. Lastly, we conduct preliminary studies on self-generating additional input questions and few-shot CoT prompts, which could further reduce the amount of human effort required for model self-improving, and ablation studies on important hyperparameters of our approach. We hope our simple approach and strong empirical results could encourage more future work by the community to investigate optimal performances of pretrained LLMs without additional human supervision.

Our contributions are summarized as follows:

- We demonstrate that a large language model can self-improve by taking datasets without ground truth outputs, by leveraging CoT reasoning (Wei et al., 2022b) and self-consistency (Wang et al., 2022b), achieving competitive in-domain multi-task performances as well as out-of-domain generalization. We achieve state-of-the-art-level results on ARC, OpenBookQA, and ANLI datasets.
- We provide detailed ablation studies on training sample formatting and sampling temperature after fine-tuning, and identify critical design choices for most successful self-improvement by LLMs.
- We study two other approaches for self-improvements, where the model generates additional questions from finite input questions and generates few-shot CoT prompt templates itself. The latter achieves 74.2% on GSM8K, which is the state-of-the-art *zero-shot* performance, against 43.0% by Kojima et al. (2022) or 70.1% through its naive extension with Wang et al. (2022b).

2 Related Work

Learning from explanations. Augmenting a machine learning model with explanations has been studied in existing literature extensively. For example, in the supervised learning setting, a model can be fine-tuned using human-annotated rationales (Zaidan et al., 2007; Ling et al., 2017b; Narang et al., 2020; Camburu et al., 2018; Cobbe et al., 2021; Chung et al., 2022). A few works have also looked at how explanations can help the models in various settings, e.g., in-context learning (Lampinen et al., 2022) and in distillation (Pruthi et al., 2022). In this paper, we focus more on the *unsupervised learning* setting, where we do not assume we have a rationale-augmented training dataset available, since human-annotated rationales can be expensive.

Few-shot explanations improves reasoning in LLMs. Recently, a lot of progress has been made towards improving LLMs' reasoning abilities via prompting or in-context learning. Wei et al.

(2022b) propose Chain-of-Thought prompting, which prompts the language model to generate a series of natural-language-based intermediate steps, and show it can help language models better solve complex and multi-step reasoning tasks. Wang et al. (2022b) improve Chain-of-Thought prompting by sampling multiple diverse reasoning paths and finding the most consistent answers via majority voting. Kojima et al. (2022) propose to prompt the language model with "Let's think step by step" to generate reasoning in a zero-shot fashion. Zhou et al. (2022a) further decompose the questions into multiple sub-questions, and ask the language model to solve each sub-question sequentially.

Refining explanations. More recent work proposes to further refine the generated reasoning paths as some of them could be unreliable. For example, Ye & Durrett (2022) calibrate model predictions based on the reliability of the explanations, Jung et al. (2022) show that inducing a tree of explanations and inferring the satisfiability of each explanation can further help judge the correctness of explanations. Li et al. (2022b) show that sampling a diverse set of prompts from the training data, and a voting verifier can be used to improve model's reasoning performance. Zelikman et al. (2022) proposes better rationale generation by augmenting ground truth answers as hints when predicted answers are incorrect. Our work is orthogonal to these lines of work, as we utilize refined explanations for model self-improvement, and could readily incorporate these other refinement techniques for generating higher-quality self-training data. Our work is similar to Zelikman et al. (2022) where we both propose to fine-tune a model on self-generated CoT data, but our method does not require ground truth labels and shows stronger empirical results with multi-task generalization.

Self-training models. One related line of work is self-training (see a survey from Amini et al. (2022)). The key idea is to assign pseudo labels from a learned classifier to unlabeled data, and use these pseudo-labeled examples to further improve the original model training, e.g., (RoyChowdhury et al., 2019; Xie et al., 2020; He et al., 2020; Chen et al., 2021). Different from such prior work, our proposed self-improvement framework uses CoT prompting plus self-consistency to obtain high-confidence solutions on a large set of unlabeled data to augment the fine-tuning process.

Distillation and dark knowledge. Our method also tangentially relates to rich literature on distillation (Ba & Caruana, 2014; Hinton et al., 2015). A key detail is to learn from soft targets instead of hard predicted labels, as softmax outputs with a high temperature reveal more detailed relative class likelihoods, colloquially known as *dark knowledge* (Hinton et al., 2015; Korattikara Balan et al., 2015). Recent studies (Zelikman et al., 2022; Snell et al., 2022; Eisenstein et al., 2022) show that *dark knowledge* within LLMs can be retrieved with more computation at inference time, such as adding informative instructions into the input sequence, and output CoT generation (Wei et al., 2022b; Kojima et al., 2022). In our work, we explicitly show that imperfect CoT reasoning (which may lead to incorrect answer) can be used directly for self-improving language models as evidenced in our experiments in Sections 5.2 and 5.3.

3 METHOD

The overview of our method is illustrated in Fig. 1: We are given a pre-trained Large Language Model (LLM) M and a question-only training dataset $\mathcal{D}^{\texttt{train}} = \{x_i\}_{i=1}^{D}$ with few-shot Chain-of-Thought (CoT) examples (Wei et al., 2022b). We apply multiple path decoding with a sampling temperature $T > 0$ for generating m reasoning paths and answers $\{r_{i_1}, r_{i_2}, \ldots, r_{i_m}\}$ for each question x_i in $\mathcal{D}^{\texttt{train}}$, and use majority voting (self-consistency) to select the most consistent, highest confidence answer (Wang et al., 2022b). We then keep all reasoning paths that lead to the most consistent answer, apply mixed formats of prompts and answers for augmentation, and fine-tune the model on these self-generated reasoning-answer data. We consider our approach as making the model self-improve. In the following sections, we detail important designs within our method, along with additional approaches for the model to self-improve without supervised data.

3.1 GENERATING AND FILTERING MULTIPLE REASONING PATHS

Self-consistency (Wang et al., 2022b) brings large improvements on reasoning tasks (e.g., $56.5\% \rightarrow 74.4\%$ on GSM8K test set), and the gap between greedy decoding and diverse decoding shows there is a potential for further improving the reasoning ability of M, using the self-selected high-confidence reasoning paths as training data.

Question: Stefan goes to a restaurant with his family. They order an appetizer that costs \$10 and 4 entrees that are \$20 each. If they tip 20% of the total, what is the total amount of money that they spend?
Multiple Path Decoding: **Output 1**: The appetizer costs \$10. The entrees cost \$20 each so in total 4 * \$20 = \$80. This means the total cost is \$10 + \$80 = \$90. They tip 20% of it, so the total amount they spend is \$90 * 1.2 = \$108. The answer is 108. **Output 2**: The appetizer costs \$10 and the entrees are \$20 each. There are 4 entrees so the sum is \$20 * 4 = \$80. The waiter gets 20% of the total. 20% of \$80 is \$80 * .2 = \$16. The answer is \$80 + \$16 = \$96. (*Incorrect reasoning path*) **Output 3**: The appetizer costs \$10. The entrees cost 4 * \$20 = \$80. The tip is 20% of the total, so it is 20% of the \$90 they have spent. The tip is 0.2 * 90 = \$18. The total they spent is \$90 + \$18 = \$108. The answer is 108.

Table 1: Examples of 3 self-generated CoT reasoning paths given a question. Output 1 and 3 are the most consistent reasoning paths based on majority voting and kept as self-training data.

For each training question x_i, we sample m CoT reasoning paths, denoted as $\{r_{i_1}, r_{i_2}, \ldots, r_{i_m}\}$ (see Table 1 for examples). Since M is prompted with the CoT examples from Wei et al. (2022b), we apply the same output parsing with "The answer is" to generate their predicted answers $\{y_{i_1}, y_{i_2}, \ldots, y_{i_m}\}$. The most consistent answer, which is not necessarily a correct answer, is selected by majority voting, denoted as $\tilde{y}_i = \arg\max_{y_{i_j}} \sum_{k=1}^{m} \mathbb{I}(y_{i_j} = y_{i_k})$. For all the training questions, we filter the CoT reasoning paths that reach $\tilde{y}$ as the final answer to be put into the self-training data, denoted as $\mathcal{D}^{\texttt{self-consistent}} = \{x_i, \tilde{\boldsymbol{r}}_i\}$, where $\tilde{\boldsymbol{r}}_i = \{r_{i_j} | 1 \leq j \leq m, y_{i_j} = \tilde{y}_i\}$.

Since we do not use any ground truth labels to filter out cases where $\tilde{y}_i \neq y_i$, it is important that the self-generated CoT reasoning paths are mostly reliable and incorrect answers do not hurt the self-improvement of the model. We plot the relation between the accuracy and confidence of self-generated CoT paths for each question in GSM8K training set in Fig. 2. The confidence is the number of CoT paths leading to $\tilde{y}$ divided by the total path number m. The y-axis shows the accuracy of $\tilde{y}$ under a certain confidence. The circle area and the color darkness shows the number of questions under a certain confidence. We can observe that confident answers are more likely to be correct, which means that when a question has many consistent CoT paths, then the corresponding $\tilde{y}$ is more likely to be correct. On the other hand, when $\tilde{y}$ is wrong, it is likely to be supported by fewer CoT paths, and brings little noise to the training samples.

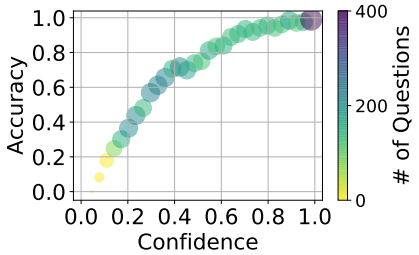


Figure 2: The relation of accuracy and confidence of the majority-voted answer after multiple path decoding on GSM8K training-set questions. Predicted confidence from self-consistency (Wang et al., 2022b) is well calibrated (Guo et al., 2017).

3.2 Training with Mixed Formats

To prevent the language model from overfitting to specific prompts or answer styles, we create four different formats for each reasoning path to be mixed in the self-training data, shown in Table 2. In the first format, a few Chain-of-Thought examples (questions followed by reasoning paths leading to the correct final answers) are prepended to the new question, while the language model output is trained to be the same with the filtered CoT reasoning paths. In the second format, we use examples of questions and their direct answers as standard prompting, and the language model output is supposed to also only contain the direct answer. The third and fourth format are similar to the first and second format, except that no example of question-answer pairs are given, so that the model will learn to think on its own in an in-context zero-shot manner. In the third format, where we want the model to output CoT reasoning without prepending examples containing CoT reasonings, we append "Let's think step by step." at the end of the input sequence, to guide the language model

to generate step-by-step CoT reasoning paths (Kojima et al., 2022). The mixed formats of training samples are then used to fine-tune the pre-trained language model M.

Question: Amy is 10 years old. Jake is 8 years old. Alex's age is right in the middle. How old is Alex? **Selected Chain-of-Thought**: Amy is 10 years old. Jake is 8 years old. Alex's age is in the middle of Amy and Jake, so Alex is (8 + 10) / 2 = 9 years old. The answer is 9.
Mixed-formats of training data: **Format 1**: **Input**: *[CoT prompting examples]* + '\n' + *[Question]* + '\n' + 'A:' **Output**: Amy is 10 years old. Jake is 8 years old. Alex's age is in the middle of Amy and Jake, so Alex is (8 + 10) / 2 = 9 years old. The answer is 9. **Format 2**: **Input**: *[Standard prompting examples]* + '\n' + *[Question]* + '\n' + 'A:' **Output**: The answer is 9. **Format 3**: **Input**: *[Question]* + '\n' + 'A: Let's think step by step.' **Output**: Amy is 10 years old. Jake is 8 years old. Alex's age is in the middle of Amy and Jake, so Alex is (8 + 10) / 2 = 9 years old. The answer is 9. **Format 4**: **Input**: *[Question]* + '\n' + 'A:' **Output**: The answer is 9.

Table 2: An example of how a reasoning path is augmented into four formats of training data with different prompts (in input) and answer styles (in output). Specifically, the *CoT prompting examples* used for each tasks are listed in Appendix A.2. The *Standard prompting examples* are the same question-answer pairs with *CoT prompting examples*, except that reasoning is removed.

3.3 Generating Questions and Prompts

Given a set of training questions and a few human-written Chain-of-Thought (CoT) examples as prompts, our proposed approach enables model self-improvement. However, when the amount of training questions or CoT examples is limited, our method may not generate sufficient training samples for language model self-training. Collecting questions from the web requires human engineering. To further reduce human effort, we investigate how to self-generate more training questions as well as example prompts.

Question Generation Previous work (Yoo et al., 2021; Meng et al., 2022) discuss few-shot data augmentation by generating diverse training samples using LLMs. However, those methods are designed for classification tasks and require ground truth label for each few-shot example. We use a simple yet effective approach to generate diverse questions (without ground truth answers) for in-domain questions. Specifically, we randomly select several existing questions, concatenate them in a random order as input prompt, and let the language model generate consecutive sequences as new questions. We repeat the process to obtain a large set of new questions, then use self-consistency (Wang et al., 2022b) to only keep the questions that have a highly confident answer. Those questions are then used as self-generated training questions.

Prompt Generation Given a set of questions, humans can write CoT examples as reasoning paths leading to the final answer. In zero-shot setting without manual prompts, we can generate these CoT paths using the model itself. Following Kojima et al. (2022), we start the answer with "A: Let's think step by step." and let the language model generate the consecutive reasoning paths. We then use those generated reasoning paths as examples for few-shot CoT prompting.

4 Experimental Setup

Tasks and Datasets. We demonstrate the effectiveness of our method on three types of tasks[1]:

[1]We evaluate on the test sets of GSM8K, ARC, OpenBookQA, and ANLI, and the dev set of DROP (ground truth labels of the test set are not publicly available).

- **Arithmetic reasoning**: We use the math problem set GSM8K (Cobbe et al., 2021), and a reading comprehension benchmark DROP (Dua et al., 2019) which requires numerical reasoning. We follow Zhou et al. (2022a) to partition the DROP dataset into football related and non-football related subsets for training.
- **Commonsense reasoning**: We use the OpenBookQA (Mihaylov et al., 2018) dataset, and the AI2 Reasoning Challenge (ARC) (Clark et al., 2018) dataset. Note that for ARC, we only use the Challenge sub-set (ARC-c) in our experiments. Both datasets contain multiple-choice questions.
- **Natural Language Inference**: We use the Adversarial NLI (ANLI) (Mihaylov et al., 2018) subsets, ANLI-A2 and ANLI-A3, which are the more challenging subsets compared to ANLI-A1. These datasets contain pairs of sentences with relations of entailment, neutral, or contradiction.

Models, Training settings and Hyperparameters We follow previous studies (Wei et al., 2022b; Wang et al., 2022b) and conduct our experiments on an autoregressive Transformer-based language model with 540 billion parameters. The CoT examples for each dataset are listed in Appendix A.2. We generate $m = 32$ reasoning paths for each question in a training set. Since each reasoning path is augmented into four formats in Sec. 3.2, the final training samples are up to the size of $128 \times |\mathcal{D}^{\texttt{train}}|$, with $|\mathcal{D}^{\texttt{train}}|$ being the size of the corresponding training set. For all datasets except DROP, we use the whole training set; To reduce the training burden, we sample 5k examples from the non-football and football partition of the DROP dataset, and sample 5k examples from ANLI-A2 and ANLI-A3. For each dataset, we fine-tune the model for 10k steps with a learning rate of $5\text{e}{-5}$ and a batch size of 32. For multiple path decoding, we use a sampling temperature of $T = 0.7$ with the pre-trained model as suggested by Wang et al. (2022b). We use $T = 1.2$ for the language model after self-improvement (**LMSI**). We set the maximum number of decoded steps to 256 for all experiments.

5 Results

We conduct a series of experiments to demonstrate the effectiveness of our proposed self-improving method. First, we apply our method on each individual dataset (task) and report the results. We then merge the generated data from all datasets and train one model to study the generalization ability of the model on unseen datasets as in (Wei et al., 2021). In addition to the results of using generated CoT reasoning paths, we show studies on generating input questions and few-shot prompts. We end with ablation studies on model sizes and hyperparameters.

5.1 Main Results

	Prompting Method	**GSM8K**	**DROP**	**ARC**-c	**OpenBookQA**	**ANLI**-A2	**ANLI**-A3
	Previous SOTA	82.3[a]	84.9[b]	88.7[c]	91.0[d]	64.9[d]	66.0[d]
w/o **LMSI**	Standard-Prompting	17.9	60.0	87.1	84.4	55.8	55.8
	CoT-Prompting	56.5	70.6	85.2	86.4	58.9	60.6
	Self-Consistency	74.4	78.2	88.7	90.0	64.5	63.4
LMSI	Standard-Prompting	32.2	71.7	87.2	92.0	64.8	66.9
	CoT-Prompting	73.5	76.2	88.3	93.0	65.3	67.3
	Self-Consistency	82.1	83.0	**89.8**	**94.4**	**66.5**	**67.9**

Table 3: Accuracy results on six reasoning benchmarks. The previous SOTA results are from: (a) Li et al. (2022a), (b) Zhou et al. (2022b), (c) Wang et al. (2022b), (d) Wang et al. (2022a).

We list the results of using the 540B model before and after **LMSI** in Table 3. For each model, during test time, we apply three separate prompting methods on all six datasets: standard-prompting, CoT-Prompting, and Self-Consistency. We observe that after **LMSI**, the performance of all three prompting methods increase by a large margin. We observe significant improvement, comparing self-consistency versus **LMSI** with self-consistency: $+7.7\%$ on GSM8K, $+4.8\%$ on DROP, $+4.4\%$ on OpenBookQA, and $+4.5\%$ on ANLI-A3. This shows that our proposed method is quite effective. Furthermore, the single path CoT-Prompting performance of **LMSI** is close to or even better than

the multiple path Self-Consistency performance of the model without **LMSI**, showing that **LMSI** truly helps the language model learn from the multiple consistent reasoning paths. We also compare our results with previous SOTA, achieved by different methods on different datasets, listed in Table 3. On ARC-c, OpenBookQA, ANLI-A2 and ANLI-A3, **LMSI** outperforms previous SOTA. On GSM8K dataset, **LMSI** is close to the DiVeRSe approach (Li et al., 2022a) which uses diverse prompts and a voting verifier to ensemble 100 output paths. On the contrary, we only use 32 output paths for self-generating training samples and for self-consistency with **LMSI**. On the DROP dataset, **LMSI** is close to the OPERA approach (Zhou et al., 2022b) which uses ground truth labels for training. On the other hand, our method only leverages the questions in the training set, without using ground truth labels.

	Self-training data	**AQUA**	**SVAMP**	**StrategyQA**	**ANLI**-A1	**RTE**	**MNLI**-M/MM
w/o **LMSI**	-	35.8	79.0	75.3	68.8	79.1	72.0/74.0
LMSI	GSM8K + DROP + ...	39.0	82.8	77.8	79.2	80.1	81.8/82.2

Table 4: Comparison of CoT-prompting accuracy results on six Out-Of-Domain benchmarks with or without training on six In-Domain (GSM8K, DROP, ARC-c, OpenBookQA, ANLI-A2, ANLI-A3) training-set questions.

Multi-task self-training for unseen tasks To demonstrate the generalization ability of **LMSI**, we conduct experiments of self-training on a mixture of the training-set questions from the above six datasets (denoted as In-Domain tasks), then use the same model checkpoint for the evaluation on six Out-Of-Domain (OOD) tasks, as shown in Table 4. Of all the OOD tasks: (1) **AQUA** (Ling et al., 2017a) and **SVAMP** (Patel et al., 2021) are arithmetic reasoning tasks; (2) **StrategyQA** (Geva et al., 2021) is a commonsense reasoning task; (3) **ANLI-A1** (Mihaylov et al., 2018), **RTE** (Dagan et al., 2005) and **MNLI-M/MM** (Williams et al., 2018) are natural language inference tasks.[2] Among these tasks, **AQUA**, **StrategyQA**, and **RTE** are significantly different from any In-Domain task. These three tasks have their own few-shot prompts. From Table 4, we can observe that **LMSI** achieves higher accuracy results on all OOD tasks, showing that the overall reasoning ability of the language model is improved.

	Results on **GSM8K**	
	Standard Prompting	CoT Prompting
w/o **LMSI**	17.9	56.5
LMSI w/o CoT formats	23.6	61.6
LMSI	32.2	73.5

Table 5: Ablation study: w/ or w/o CoT reasoning paths as training format on GSM8K dataset.

Importance of training with Chain-of-Thought formats We demonstrate the importance of training language models with Chain-of-Thoughts compared to training with only direct answers. In Table 5, we list the results of **LMSI** with all four formats, and the results of **LMSI** with only direct answer formats. The results clearly show that without the CoT formats, the language model can still self-improve, but the performance gain drops by a large amount compared to using all four formats.

5.2 Pushing the limit of self-improvements

Self-Generating Questions We further explore the few-shot setting where there are only limited training questions in the target domain. On GSM8K, we sample 10 real questions as few-shot

[2]We evaluate on the test set of SVAMP and ANLI, the dev set of MNLI and RTE (ground truth labels of the test sets are not publicly available). For StrategyQA we use the question-only set from bench collaboration (2022).

	Questions used for Self-Training	Results on **GSM8K** CoT-Prompting	Self-Consistency
w/o **LMSI**	-	56.5	74.4
LMSI	Generated Questions	66.2	78.1
LMSI	Training-set Questions	**73.5**	**82.1**

Table 6: Accuracy on GSM8K test set after self-training on self-generated or training set questions.

samples, and use the language model to generate more training questions using the method in Section 3.3. We then self-train the language model with these generated questions and list the results in Table 6. The results show that using self-generated questions still improves the reasoning ability of language models, but using the real training-set questions leads to better results.

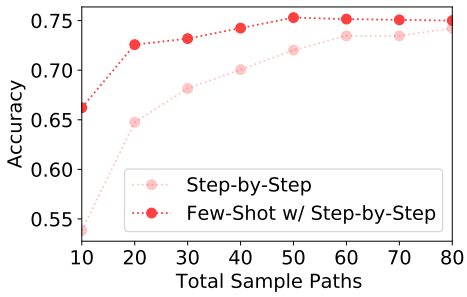


Figure 3: Accuracy results on GSM8K test set using 540B model with multi-path sampling and self-consistency (Wang et al., 2022b). "Step-by-Step" is the baseline performance of Kojima et al. (2022) plus self-consistency (Wang et al., 2022b), while our "Few-Shot w/ Step-by-Step" uses exemplers self-generated from Step-by-Step (greedy decoding) for few-shot prompting the LLM.

Self-Generating Few-Shot CoT Prompts We explore the situation where no in-domain CoT examples are provided for a task. We apply the Step-by-Step method (Kojima et al., 2022) to generate CoT examples using the language model as described in Section 3.3, and show the results in Figure 3. We observe that few-shot prompting with self-generated Step-by-Step CoT examples substantially outperforms the Step-by-Step (Kojima et al., 2022) baseline (66.2% vs 53.8% at 10 paths, 74.2% vs 70.1% at 40 paths), and nearly matches the performance of human-written few-shot CoT (Wei et al., 2021) (74.4% at 40 paths (Wang et al., 2022b)). The strong performance of "Few-Shot w/ Step-by-Step" despite the limited accuracy of prompt examples (43.0% for greedy Step-by-Step) likely comes from leveraging more diverse CoT prompts for multi-path decoding (Li et al., 2022a), where at 40 paths it uses 20 generate prompt-templates, each with 4-shot CoT examples, i.e. a total of 80 generated CoT examples compared to 8 human-written examples use in Wei et al. (2022b). Since we did not use training questions or few-shot CoT examples, 74.2% also marks the new state-of-the-art zero-shot performance on GSM8K.

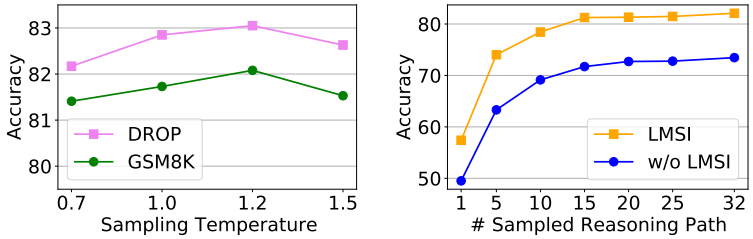
5.3 Distillation to smaller models

	Results on **GSM8K** 8 billion	62 billion	540 billion
w/o **LMSI**	5.0	29.7	56.5
Distilled from **LMSI** 540 billion	33.4	57.4	-

Table 7: Distillation from 540B model to small models. We see that distilled smaller models outperform models that are one-tier larger.

We also explore whether the knowledge can be distilled to smaller models, such as in distillation (Hinton et al., 2015) and in Zelikman et al. (2022). We use the same set of training samples generated by the 540B model, but fine-tune on models with smaller sizes (8B and 62B respectively), and show the results of CoT-prompting in Table 7. It is interesting to point out that after distillation from **LMSI**, the 62 billion model can outperform the pre-trained 540 billion model, and the 8 billion model can outperform the pre-trained 62 billion model. This implies that for downstream applications with limited computing resources, the reasoning knowledge from large models can be used to largely enhance small models to achieve competitive performance.

5.4 Hyperparameter Study



(a) Accuracy of **LMSI** on GSM8K and DROP test set when different sampling temperatures are applied for Self-Consistency.

(b) Accuracy results with or without **LMSI** on GSM8K test set using different numbers of sampled reasoning path for Self-Consistency.

Figure 4: Hyperparameter study results.

Sampling Temperature after Self-Improvement We study the effect of varying the temperature T for multiple path decoding after **LMSI** is applied. Specifically, we vary T between $[0.7, 1.0, 1.2, 1.5]$ and show the results on GSM8K and DROP dataset respectively in Fig. 4(a). As shown in the figure, $T = 1.2$ benefits both datasets the most, and is used in the Self-Consistency method for **LMSI** on all datasets. We notice that the optimal T after model self-improvement is larger than the optimal $T = 0.7$ (Wang et al., 2022b) before self-improvement. We believe the reason is that after training the model, the entropy of the output distribution is reduced.

Number of Sampled Reasoning Paths We study whether the number of sampled reasoning paths m for Self-Consistency largely affects the accuracy after **LMSI** is applied. We show the accuracy on GSM8K test set for models both with or without **LMSI** in Fig. 4(b). For both cases, setting $m = 15$ already achieves a reasonably good accuracy, and using a larger m only brings marginal improvements. We also notice that after Self-Improvement, using 5 paths for Self-Consistency can already surpass the performance of using 32 paths for model without Self-Improvement. Thus, with a well-improved model, huge computing resources can be saved when applied to real applications.

6 Conclusions

We demonstrated that a Large Language Model (LLM) is capable of improving its performance on reasoning datasets by training on its own generated labels, given input questions only. Experiments using an LLM with 540 billion parameters show that our approach improves the accuracy scores on the six datasets by 1.1% to 7.7%, achieving new state-of-the-art results on ARC, OpenBookQA, and ANLI, without training on ground truth labels. Furthermore, we show that it is possible for the LLM to self-improve even on its own generated questions and few-shot Chain-of-Thought prompts. As part of our future work, we plan to combine large-scale generated data from our approach and existing supervised data, to further improve the performance of LLMs.

REFERENCES

Massih-Reza Amini, Vasilii Feofanov, Loic Pauletto, Emilie Devijver, and Yury Maximov. Self-training: A survey, 2022. URL https://arxiv.org/abs/2202.12040.

Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? *Advances in neural information processing systems*, 27, 2014.

BIG bench collaboration. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *ArXiv*, abs/2206.04615, 2022.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Neurips*, 2020.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. e-snli: Natural language inference with natural language explanations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 9539–9549. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/8163-e-snli-natural-language-inference-with-natural-language-explanations.pdf.

Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek B Rao, Parker Barnes, Yi Tay, Noam M. Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Benton C. Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier García, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Oliveira Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Díaz, Orhan Firat, Michele Catasta, Jason Wei, Kathleen S. Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *ArXiv*, abs/2204.02311, 2022.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Adams Yu, Albert Webson, Xinyun Chen, Gaurav Mishra, Zhuyun Dai, Shixiang Shane Gu, Mirac Suzgun, Vincent Zhao, Aakanksha Chowdhery, Sharan Narang, Yanping Huang, Andrew Dai, Hongkun Yu, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. In *arxiv*, 2022.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457, 2018.

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

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *MLCW*, 2005.

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

John Dunlosky and Janet Metcalfe. *Metacognition*. Sage Publications, 2008.

Jacob Eisenstein, Daniel Andor, Bernd Bohnet, Michael Collins, and David Mimno. Honest students from untrusted teachers: Learning an interpretable question-answering pipeline from a pretrained language model. *arXiv preprint arXiv:2210.02498*, 2022.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. Revisiting self-training for neural sequence generation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SJgdnAVKDH.

Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

Jaehun Jung, Lianhui Qin, Sean Welleck, Faeze Brahman, Chandra Bhagavatula, Ronan Le Bras, and Yejin Choi. Maieutic prompting: Logically consistent reasoning with recursive explanations, 2022. URL https://arxiv.org/abs/2205.11822.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Neural Information Processing Systems (NeurIPS)*, 2022.

Anoop Korattikara Balan, Vivek Rathod, Kevin P Murphy, and Max Welling. Bayesian dark knowledge. *Advances in neural information processing systems*, 28, 2015.

Andrew K. Lampinen, Ishita Dasgupta, Stephanie C. Y. Chan, Kory Matthewson, Michael Henry Tessler, Antonia Creswell, James L. McClelland, Jane X. Wang, and Felix Hill. Can language models learn from explanations in context?, 2022. URL https://arxiv.org/abs/2204.02329.

Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. On the advance of making language models better reasoners. *ArXiv*, abs/2206.02336, 2022a.

Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. On the advance of making language models better reasoners, 2022b. URL https://arxiv.org/abs/2206.02336.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *ACL*, 2017a.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017b. doi: 10.18653/v1/P17-1015. URL https://aclanthology.org/P17-1015.

Yu Meng, Jiaxin Huang, Yu Zhang, and Jiawei Han. Generating training data with language models: Towards zero-shot language understanding. *ArXiv*, abs/2202.04538, 2022.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*, 2018.

Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. Wt5?! training text-to-text models to explain their predictions, 2020. URL https://arxiv.org/abs/2004.14546.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022.

Arkil Patel, S. Bhattamishra, and Navin Goyal. Are nlp models really able to solve simple math word problems? In *NAACL*, 2021.

Danish Pruthi, Rachit Bansal, Bhuwan Dhingra, Livio Baldini Soares, Michael Collins, Zachary C. Lipton, Graham Neubig, and William W. Cohen. Evaluating Explanations: How Much Do Explanations from the Teacher Aid Students? *Transactions of the Association for Computational Linguistics*, 10:359–375, 04 2022. ISSN 2307-387X. doi: 10.1162/tacl_a_00465. URL https://doi.org/10.1162/tacl_a_00465.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*, 2016.

Aruni RoyChowdhury, Prithvijit Chakrabarty, Ashish Singh, SouYoung Jin, Huaizu Jiang, Liangliang Cao, and Erik G. Learned-Miller. Automatic adaptation of object detectors to new domains using self-training. In *CVPR*, pp. 780–790, 2019. URL http://openaccess.thecvf.com/content_CVPR_2019/html/RoyChowdhury_Automatic_Adaptation_of_Object_Detectors_to_New_Domains_Using_Self-Training_CVPR_2019_paper.html.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask prompted training enables zero-shot task generalization. In *ICLR*, 2022.

Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by distilling context. *arXiv preprint arXiv:2209.15189*, 2022.

Yi Tay, Mostafa Dehghani, Vinh Quang Tran, Xavier García, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Huaixiu Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. Ul2: Unifying language learning paradigms. 2022.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP@EMNLP*, 2018.

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

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. Rationale-augmented ensembles in language models. *ArXiv*, abs/2207.00747, 2022a.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *ArXiv*, abs/2203.11171, 2022b.

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. *arXiv preprint arXiv:2109.01652*, 2021.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022a.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Brian Ichter, Fei Xia, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 2022b.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL*, 2018.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves imagenet classification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, 2020. doi: 10.1109/CVPR42600.2020.01070.

Xi Ye and Greg Durrett. The unreliability of explanations in few-shot in-context learning, 2022.

Kang Min Yoo, Dongju Park, Jaewook Kang, Sang-Woo Lee, and Woomyeong Park. Gpt3mix: Leveraging large-scale language models for text augmentation. In *EMNLP Findings*, 2021.

Omar Zaidan, Jason Eisner, and Christine Piatko. Using "annotator rationales" to improve machine learning for text categorization. *NAACL*, 2007. URL https://aclanthology.org/N07-1033.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning, 2022. URL https://arxiv.org/abs/2203.14465.

Denny Zhou, Nathanael Scharli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Olivier Bousquet, Quoc Le, and Ed Chi. Least-to-most prompting enables complex reasoning in large language models. *ArXiv*, abs/2205.10625, 2022a.

Yongwei Zhou, Junwei Bao, Chaoqun Duan, Haipeng Sun, Jiahui Liang, Yifan Wang, Jing Zhao, Youzheng Wu, Xiaodong He, and Tiejun Zhao. Opera: Operation-pivoted discrete reasoning over text. In *NAACL*, 2022b.

A APPENDIX

A.1 RESULTS ON UL2 MODEL

We also apply **LMSI** on a recently proposed public language model, UL2 (Tay et al., 2022), using the pre-trained model at step 2,650,000[3]. We use a fixed set of hyperparameters for fine-tuning on each dataset. Specifically, we generate $m = 40$ reasoning paths for each question in a training set for majority voting. We fine-tune the model for 10k steps with a learning rate of $5\text{e}{-5}$ and a batch size of 32. For multiple path decoding, we use a sampling temperature of $T = 0.5$ with the pre-trained UL2 model following Tay et al. (2022), and set $T = 0.7$ for the language model after **LMSI**. We set the maximum number of decode steps to 256 for all experiments.

The results are shown in Table 8. For arithmetic reasoning datasets, we follow (Tay et al., 2022) to provide both exact matching accuracy scores as well as accuracy scores after an equation-correction postprocessing step. We observe that for most datasets, **LMSI** still improves the reasoning accuracy (+1.6% on DROP, +1.2% on OpenBookQA, and +0.7% on ANLI-A2), but the improvement on UL2 is not as large as that on 540B. We think the reason is that, since **LMSI** exploits the implicit rationale of language models, and the capacity of a language model is determined by its size, larger models can capture more high-order semantics and are more likely to benefit from **LMSI**. For example, on the adversarial entailment tasks of ANLI (which is a three-class classification problem with labels "yes", "no", or "it is not possible to tell"), the UL2 model w/o **LMSI** only achieves an accuracy of marginally above $1/3$, implying that the model is slightly better than doing random guess on this challenging task without any training. Our proposed **LMSI** can still improve the performance under this hard case by training on its implicit knowledge from self-generated paths.

	Prompting Method	**GSM8K**	**DROP**	**ARC**-c	**OpenBookQA**	**ANLI**-A2	**ANLI**-A3
w/o **LMSI**	CoT-Prompting	5.4/7.1	11.1/16.8	49.9	53.6	35.9	33.8
	Self-Consistency	6.4/9.9	16.8/26.5	54.9	54.0	37.4	36.8
LMSI	CoT-Prompting	6.1/8.6	11.4/17.1	50.9	53.8	35.4	34.4
	Self-Consistency	7.9/10.2	18.1/28.1	54.7	55.2	38.1	37.4

Table 8: Accuracy results on six reasoning benchmarks with **LMSI** on UL2. On GSM8K and DROP, we also include accuracy scores after an equation-correction postprocessing step.

A.2 CHAIN-OF-THOUGHT PROMPTS FOR EACH DATASET

We list the Chain-of-Thought Prompts for each dataset for "CoT-Prompting" experiments and self-generated training samples.

[3]UL2: `https://github.com/google-research/google-research/tree/master/ul2`

Q: Poison causes harm to which of the following? (a) a Tree (b) a robot (c) a house (d) a car

A: Poison will harm living things, only a tree is a living thing. The answer is (a).

Q: As you look deeper into a Marbel you can see (a) the future (b) minut defects (c) colors (d) the other side

A: Marbel is not transparent, so you can not see the other side. Marbel does not necessarily have multiple colors. You will see minut defects. The answer is (b).

Q: When food is reduced in the stomach (a) the mind needs time to digest (b) take a second to digest what I said (c) nutrients are being deconstructed (d) reader's digest is a body of works

A: The food is being deconstructed in the stomach during digestion. The answer is (c).

Q: The sun is responsible for (a) puppies learning new tricks (b) children growing up and getting old (c) flowers wilting in a vase (d) plants sprouting, blooming and wilting

A: The sun can affect the growing of living things, like plants. The answer is (d).

Table 9: Few-shot CoT prompts for OpenBookQA, from Wang et al. (2022a).

Q: There are 15 trees in the grove. Grove workers will plant trees in the grove today. After they are done, there will be 21 trees. How many trees did the grove workers plant today?

A: We start with 15 trees. Later we have 21 trees. The difference must be the number of trees they planted. So, they must have planted 21 - 15 = 6 trees. The answer is 6.

Q: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?

A: There are 3 cars in the parking lot already. 2 more arrive. Now there are 3 + 2 = 5 cars. The answer is 5.

Q: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?

A: Leah had 32 chocolates and Leah's sister had 42. That means there were originally 32 + 42 = 74 chocolates. 35 have been eaten. So in total they still have 74 - 35 = 39 chocolates. The answer is 39.

Q: Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How many lollipops did Jason give to Denny?

A: Jason had 20 lollipops. Since he only has 12 now, he must have given the rest to Denny. The number of lollipops he has given to Denny must have been 20 - 12 = 8 lollipops. The answer is 8.

Q: Shawn has five toys. For Christmas, he got two toys each from his mom and dad. How many toys does he have now?

A: He has 5 toys. He got 2 from mom, so after that he has 5 + 2 = 7 toys. Then he got 2 more from dad, so in total he has 7 + 2 = 9 toys. The answer is 9.

Q: There were nine computers in the server room. Five more computers were installed each day, from monday to thursday. How many computers are now in the server room?

A: There are 4 days from monday to thursday. 5 computers were added each day. That means in total 4 * 5 = 20 computers were added. There were 9 computers in the beginning, so now there are 9 + 20 = 29 computers. The answer is 29.

Q: Michael had 58 golf balls. On tuesday, he lost 23 golf balls. On wednesday, he lost 2 more. How many golf balls did he have at the end of wednesday?

A: Michael initially had 58 balls. He lost 23 on Tuesday, so after that he has 58 - 23 = 35 balls. On Wednesday he lost 2 more so now he has 35 - 2 = 33 balls. The answer is 33.

Q: Olivia has $23. She bought five bagels for $3 each. How much money does she have left?

A: She bought 5 bagels for $3 each. This means she spent 5 * $3 = $15 on the bagels. She had $23 in beginning, so now she has $23 - $15 = $8. The answer is 8.

Table 10: Few-shot CoT prompts for GSM8K and SVAMP, from Wei et al. (2022b).

Q: Since the 1970s, U.S. governments have negotiated managed-trade agreements, such as the North American Free Trade Agreement in the 1990s, the Dominican Republic-Central America Free Trade Agreement in 2006, and a number of bilateral agreements. In Europe, six countries formed the European Coal and Steel Community in 1951 which became the European Economic Community in 1958. Two core objectives of the EEC were the development of a common market, subsequently renamed the single market, and establishing a customs union between its member states. How many years did the European Coal and Steel Community exist?

A: According to the passage, the European Coal and Steel Community was established in 1951 and became the EEC in 1958. 1958 - 1951 = 7. So the answer is 7.

Q: In the county, the population was spread out with 23.50% under the age of 18, 8.70% from 18 to 24, 29.70% from 25 to 44, 24.70% from 45 to 64, and 13.30% who were 65 years of age or older. How many more percent are under the age of 18 compared to the 18 to 24 group?

A: According to the passage, 23.5% are under the age of 18, and 8.7% are from ages 18 to 24. 23.5% - 8.7% = 14.8%. So the answer is 14.8.

Q: Playing in their second straight Thanksgiving game, the Eagles struggled especially on defense, where they were unable to stop the much-hyped Lions offense. The worst of it all was how unproven rookie Eric Rowe was tasked with covering wide receiver Calvin Johnson, leading to Johnson catching 3 touchdowns. Stafford's five passing touchdowns, including three of them to Johnson was too much for the Eagles to overcome and for the second consecutive time this season, the Eagles gave up 45 points in a game. With the loss, the Eagles drop to 4-7 on the season and 6-1 when playing on Thanksgiving. How many TD passes did Stafford throw other than to Johnson?

A: According to the passage, Stafford threw 5 TD passes, 3 of which were to Johnson. 5 - 3 = 2. So the answer is 2.

Table 11: Few-shot CoT prompts for DROP (nonfootball), from Zhou et al. (2022a).

Q: The Seahawks played the San Francisco 49ers. In the first quarter, the Hawks RB Julius Jones got a 27-yard TD run, along with DT Craig Terrill returning a fumble 9 yards for a touchdown. In the third quarter, the 49ers almost rallied as RB H. J. Torres made a 12-yard TD pass to Lucas Nelly, along with Mare kicking a 32-yard field goal. In the final quarter, Julius Jones got another 11-yard TD. How many yards do the shortest touchdown run and the longest touchdown pass combine for?

A: All the touchdown runs are: a 27-yard touchdown run, a 9-yard touchdown run, a 11-yard touchdown run. The smallest number among 27, 9, 11 is 9. So the shortest touchdown run was 9 yards. All the touchdown passes are: a 12-yard touchdown pass. So the longest touchdown pass was 12 yards. So the shortest touchdown run and the longest touchdown pass combine for 9 + 12 = 21 yards. So the answer is 21 yards.

Q: The Steelers went home for a duel with the Baltimore Ravens. Pittsburgh would deliver the opening punch in the first quarter with a 1-yard touchdown from running back Rashard Mendenhall. The Ravens would make it even as running back Willis McGahee got a 9-yard TD. The Ravens kicker Billy Cundiff got a 45-yard field goal in the second quarter, concluding the first half with a 10-7 lead. The Steelers brought the game into overtime with a 38-yard field goal by Andrew Foster. The Ravens Billy Cundiff pulled off a winning 33-yard field goal in overtime. How many points did the Ravens have at halftime?

A: The Ravens kicker Billy Cundiff got a 45-yard field goal in the second quarter, concluding the first half with a 10-7 lead. So the Ravens had 10 points at halftime. So the answer is 10 points.

Q: The Vikings flew to Bank of America Stadium to face the Carolina Panthers. After a scoreless first quarter, Carolina got on the board with quarterback Matt Moore finding fullback Brad Hoover on a 1-yard TD pass. After yet another scoreless quarter, Carolina sealed the game as Matt Moore completed a 42-yard touchdown pass to wide receiver Steve Smith. How many scoreless quarters were there?

A: The first and third quarters were the scoreless quarters. So there are 2 scoreless quarters. So the answer is 2.

Table 12: Few-shot CoT prompts for DROP (football), from Zhou et al. (2022a).

Premise:
"Conceptually cream skimming has two basic dimensions - product and geography."
Based on this premise, can we conclude the hypothesis "Product and geography are what make cream skimming work." is true?
OPTIONS:
- yes
- no
- it is not possible to tell

A: Based on "cream skimming has two basic dimensions" we can't infer that these two dimensions are what make cream skimming work. The answer is it is not possible to tell.

Premise:
"One of our member will carry out your instructions minutely."
Based on this premise, can we conclude the hypothesis "A member of my team will execute your orders with immense precision." is true?
OPTIONS:
- yes
- no
- it is not possible to tell

A: "one of" means the same as "a member of", "carry out" means the same as "execute", and "minutely" means the same as "immense precision". The answer is yes.

Premise:
"Fun for adults and children."
Based on this premise, can we conclude the hypothesis "Fun for only children." is true?
OPTIONS:
- yes
- no
- it is not possible to tell

A: "adults and children" contradicts "only children". The answer is no.

Premise:
"He turned and smiled at Vrenna."
Based on this premise, can we conclude the hypothesis "He smiled at Vrenna who was walking slowly behind him with her mother." is true?
OPTIONS:
- yes
- no
- it is not possible to tell

A: the premise does not say anything about "Vrenna was walking". The answer is it is not possible to tell.

Premise:
"well you see that on television also"
Based on this premise, can we conclude the hypothesis "You can see that on television, as well." is true?
OPTIONS:
- yes
- no
- it is not possible to tell

A: "also" and "as well" mean the same thing. The answer is yes.

Premise:
"Vrenna and I both fought him and he nearly took us."
Based on this premise, can we conclude the hypothesis "Neither Vrenna nor myself have ever fought him." is true?
OPTIONS:
- yes
- no
- it is not possible to tell

A: "Vrenna and I both" contradicts "neither Vrenna nor myself". The answer is no.

Table 13: Few-shot CoT prompts for NLI tasks, including ANLI and MNLI, from Wang et al. (2022a).

Q: George wants to warm his hands quickly by rubbing them. Which skin surface will produce the most heat? (a) dry palms. (b) wet palms. (c) palms covered with oil. (d) palms covered with lotion.

A: Dry surfaces will more likely cause more friction via rubbing than other smoother surfaces, hence dry palms will produce the most heat. The answer is (a).

Q: Which factor will most likely cause a person to develop a fever? (a) a leg muscle relaxing after exercise. (b) a bacterial population in the bloodstream. (c) several viral particles on the skin. (d) carbohydrates being digested in the stomach.

A: Option (b), bacterial population is the most likely cause for a person developing fever. The answer is (b).

Q: Which change in the state of water particles causes the particles to become arranged in a fixed position? (a) boiling. (b) melting. (c) freezing. (d) evaporating.

A: When water is freezed, the particles are arranged in a fixed position; the particles are still moving for all other options. The answer is (c).

Q: When a switch is used in an electrical circuit, the switch can (a) cause the charge to build. (b) increase and decrease the voltage. (c) cause the current to change direction. (d) stop and start the flow of current.

A: The function of a switch is to start and stop the flow of a current. The answer is (d).

Table 14: Few-shot CoT prompts for ARC-challenge, from Wang et al. (2022b).

Q: John found that the average of 15 numbers is 40. If 10 is added to each number then the mean of the numbers is? Answer Choices: (a) 50 (b) 45 (c) 65 (d) 78 (e) 64

A: If 10 is added to each number, then the mean of the numbers also increases by 10. So the new mean would be 50. The answer is (a).

Q: If a / b = 3/4 and 8a + 5b = 22,then find the value of a. Answer Choices: (a) 1/2 (b) 3/2 (c) 5/2 (d) 4/2 (e) 7/2

A: If a / b = 3/4, then b = 4a / 3. So 8a + 5(4a / 3) = 22. This simplifies to 8a + 20a / 3 = 22, which means 44a / 3 = 22. So a is equal to 3/2. The answer is (b).

Q: A person is traveling at 20 km/hr and reached his destiny in 2.5 hr then find the distance? Answer Choices: (a) 53 km (b) 55 km (c) 52 km (d) 60 km (e) 50 km

A: The distance that the person traveled would have been 20 km/hr * 2.5 hrs = 50 km. The answer is (e).

Q: How many keystrokes are needed to type the numbers from 1 to 500? Answer Choices: (a) 1156 (b) 1392 (c) 1480 (d) 1562 (e) 1788

A: There are 9 one-digit numbers from 1 to 9. There are 90 two-digit numbers from 10 to 99. There are 401 three-digit numbers from 100 to 500. 9 + 90(2) + 401(3) = 1392. The answer is (b).

Table 15: Few-shot CoT prompts for AQUA, from Wang et al. (2022b).

Q: Do hamsters provide food for any animals?

A: Hamsters are prey animals. Prey are food for predators. Thus, hamsters provide food for some animals. The answer is yes.

Q: Could Brooke Shields succeed at University of Pennsylvania?

A: Brooke Shields went to Princeton University. Princeton University is about as academically rigorous as the University of Pennsylvania. Thus, Brooke Shields could also succeed at the University of Pennsylvania. The answer is yes.

Q: Yes or no: Hydrogen's atomic number squared exceeds number of Spice Girls?

A: Hydrogen has an atomic number of 1. 1 squared is 1. There are 5 Spice Girls. Thus, Hydrogen's atomic number squared is less than 5. The answer is no.

Q: Yes or no: Is it common to see frost during some college commencements?

A: College commencement ceremonies can happen in December, May, and June. December is in the winter, so there can be frost. Thus, there could be frost at some commencements. The answer is yes.

Q: Yes or no: Could a llama birth twice during War in Vietnam (1945-46)?

A: The War in Vietnam was 6 months. The gestation period for a llama is 11 months, which is more than 6 months. Thus, a llama could not give birth twice during the War in Vietnam. The answer is no.

Q: Yes or no: Would a pear sink in water?

A: The density of a pear is about 0.6 g/cm^3, which is less than water. Objects less dense than water float. Thus, a pear would float. The answer is no.

Table 16: Few-shot CoT prompts for StrategyQA, from Wei et al. (2022b).

Premise:
"No Weapons of Mass Destruction Found in Iraq Yet."
Based on this premise, can we conclude the hypothesis "Weapons of Mass Destruction Found in Iraq." is true?

A: "No Weapons of Mass Destruction Found" contradicts "Weapons of Mass Destruction Found". The answer is no.

Premise:
"A place of sorrow, after Pope John Paul II died, became a place of celebration, as Roman Catholic faithful gathered in downtown Chicago to mark the installation of new Pope Benedict XVI."
Based on this premise, can we conclude the hypothesis "Pope Benedict XVI is the new leader of the Roman Catholic Church." is true?'

A: "installation of new Pope Benedict XVI." means "Pope Benedict XVI is the new leader". The answer is yes.

Premise:
"A man is due in court later charged with the murder 26 years ago of a teenager whose case was the first to be featured on BBC One's Crimewatch. Colette Aram, 16, was walking to her boyfriend's house in Keyworth, Nottinghamshire, on 30 October 1983 when she disappeared. Her body was later found in a field close to her home. Paul Stewart Hutchinson, 50, has been charged with murder and is due before Nottingham magistrates later."
Based on this premise, can we conclude the hypothesis "Paul Stewart Hutchinson is accused of having stabbed a girl." is true?

A: The premise does not say Paul Stewart Hutchinson "stabbed" this girl. The answer is no.

Premise:
"Herceptin was already approved to treat the sickest breast cancer patients, and the company said, Monday, it will discuss with federal regulators the possibility of prescribing the drug for more breast cancer patients."
Based on this premise, can we conclude the hypothesis "Herceptin can be used to treat breast cancer." is true?

A: "Herceptin was approved to treat breast cancer" implies that "Herceptin can be used to treat breast cancer". The answer is yes.

Table 17: Few-shot CoT prompts for RTE, from Wang et al. (2022a).