

PRECOT: PROBLEM REPRESENTATION ENHANCES REASONING IN LARGE LANGUAGE MODELS

Anonymous authors

Paper under double-blind review

ABSTRACT

Chain-of-Thought (CoT) prompting has broken new ground in exploring the reasoning capabilities of large language models (LLMs). Recent studies expand this direction to specific strategies, such as question decomposition and planning, to improve the solution process. On the other hand, within cognitive psychology, problem representation construction is considered a vital aspect of human problem-solving along with the solution process. It involves a solver structurally encoding a problem by defining its initial and goal states, thereby improving the solution process. However, the construction of problem representation has yet to be tapped in further exploring the potential of LLMs’ human-like problem-solving ability. In this work, we propose **Problem Representation Enhanced CoT** (PRECOT), a novel prompting framework that enhances the solution process of LLMs with problem representation. PRECOT is divided into two main stages. First, it extracts the ingredients of the initial and goal state of the problem, which constitute the problem representation together. Next, it initiates an enhanced solution process based on the generated problem representation. In extensive evaluation on benchmarks from a wide range of domains, including arithmetic, commonsense, and symbolic reasoning, PRECOT outperforms CoT on most tasks in both few-shot and zero-shot manners. Additional analyses further demonstrate the effectiveness of problem representation and its contribution to the reasoning in LLMs, such as robustness against irrelevant context and problem context sensitivity.

1 INTRODUCTION

Scaling up language model (LM) has been a pivotal strategy to yield exceptional results across numerous Natural Language Processing (NLP) benchmarks (Devlin et al., 2019; Brown et al., 2020; Chowdhery et al., 2022). This progression in size has facilitated a shift from the conventional *pre-train and fine-tune* paradigm to *pre-train, prompt, and predict* paradigm (Kaplan et al., 2020; Liu et al., 2023). However, while prompting techniques have made remarkable progress in many areas of NLP, multi-step reasoning remained a challenging frontier where the scaling up appeared less effective (Rae et al., 2022).

The recent proposal of the *Chain-of-Thought* (CoT) *prompting* (Wei et al., 2022c), which prompts large language models (LLMs) to generate not only final answer but also intermediate reasoning steps, unleashed reasoning capabilities of LLMs. In response to this insight, zero-shot CoT (Kojima et al., 2022) shows CoT demonstrations can be replaced with a one-line instruction (e.g., “Let’s think step by step.”) and it significantly boosts zero-shot reasoning performance.

Although there has been notable progress in the field of multi-step reasoning in LLMs, there exists an untapped yet vital aspect of problem-solving. Human problem-solving is divided into two main phases in the information-processing theories of problem-solving in cognitive psychology: (a) construction of problem representation; (b) solution searching and its implementation (Newell, 1972; Greeno, 1978; Gick, 1986). When faced with a problem, humans first structurally encode the problem from two aspects, given information (initial state) and goal information (goal state) (Greeno, 1977; Gick, 1986). This structural interpretation is called problem representation and is the foundation of the human problem-solving process. Solution searching and subsequent implementation can be viewed as a series of cognitive processes that leads the initial state to the goal state (Davidson,

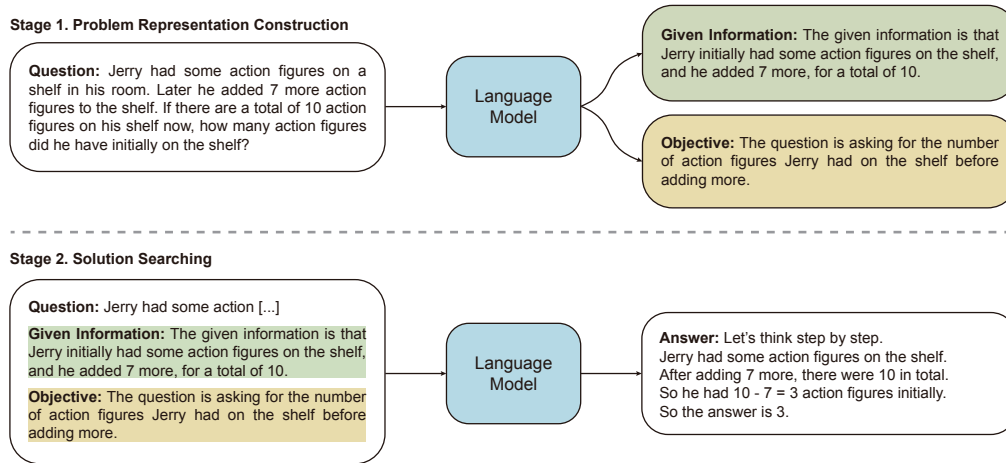


Figure 1: PRECOT consists of problem representation construction and solution searching stages. Integrating problem representation into solution searching process enhances reasoning in LLMs.

1994). However, studies in the reasoning of LLMs heavily focus on solution searching and its implementation. Those approaches include inducing LLMs to explicitly generate solution process without any control on its trajectory (Wei et al., 2022c; Kojima et al., 2022) and guiding LLMs to utilize specific solution strategies and procedures such as question decomposition (Fu et al., 2021; Zhou et al., 2023; Khot et al., 2023; Press et al., 2023), and planning (Wang et al., 2023; Hao et al., 2023; Yao et al., 2023). They initiate a reasoning process without any aid in understanding the problem. While problem representation construction is a crucial part of human problem-solving, as it contributes to solution searching or schema activation (Gick, 1986), it has yet to be explored whether this can also be beneficial to the reasoning in LLMs.

To investigate the effectiveness of problem representation in the reasoning of LLMs, we design **Problem Representation Enhanced COT (PRECOT)**, a multi-stage prompting framework that helps LLMs build the problem representation. Similar to human problem representation, we consider the information specified in a problem as the initial state and the objective of the problem as the goal state. PRECOT sets these two states and incorporates them into the prompt to prime the solution process (Figure 1). We implement PRECOT in two fashions: few-shot and zero-shot prompting. Few-shot prompting leverages task demonstrations (Brown et al., 2020) to extract both states, while zero-shot prompting only utilizes instructions to obtain the states. We employ two transformer (Vaswani et al., 2017) based LLMs, PaLM 2 (Anil et al., 2023) and GPT-3 (Brown et al., 2020), to investigate the effectiveness of problem representation in the reasoning of LLMs. In extensive evaluation on a wide range of multi-step reasoning benchmarks, including arithmetic, commonsense, and symbolic reasoning, PRECOT outperforms few-shot and zero-shot COT on most benchmarks. It validates the effectiveness of introducing problem representation in the reasoning of LLMs across such multiple domains. Additional analyses show notable strengths of problem representation grounded solution searching, such as robustness against irrelevant context and problem context sensitivity.

To sum up, our contribution is three-fold.

1. We propose a novel approach to improve reasoning in LLMs called PRECOT. It provides structured problem representation to LLMs and thereby enhances its solution process.
2. In extensive evaluation on multiple domain benchmarks (arithmetic, commonsense, and symbolic reasoning), our PRECOT outperforms both few-shot and zero-shot COTs on most benchmarks, demonstrating its effectiveness. Additional analyses further support the benefit of problem representation.
3. Inspired by studies in cognitive psychology, this proposed approach offers useful perspectives into investigating LLMs' reasoning process.

2 RELATED WORK

Large Language Model and Prompting. Training transformer-based LMs at scale yields remarkable results on a wide range of NLP benchmarks (Devlin et al., 2019; Brown et al., 2020; Chowdhery et al., 2022). Scaled LMs (i.e., LLMs) show notable gradient-free learning capabilities with few-shot and zero-shot prompting (Brown et al., 2020; Kaplan et al., 2020; Liu et al., 2023; Wei et al., 2022b). In this paradigm, LLMs can generate responses to diverse tasks based on natural language descriptions without parameter updates. Few-shot prompting provides task demonstrations as a prompt for LLMs to generate a response similar to the examples in the demonstration. In zero-shot prompting, only a task instruction is given to LLMs to generate the desired response. In this study, we employ both prompting techniques to demonstrate the effectiveness of the proposed approach universally.

Reasoning in Large Language Models. CoT prompting (Wei et al., 2022c) exhibits that multi-step reasoning is one of the emergent abilities (Wei et al., 2022b) of LLMs. It induces the model to generate intermediate reasoning steps that lead to the final answer by priming LLMs with manually designed demonstrations. Kojima et al. (2022) substitutes the demonstration with a concise one-line instruction, “Let’s think step by step.” for the same effect. Numerous studies exploit this direction, including solution strategies such as problem decomposition (Fu et al., 2021; Khot et al., 2023; Zhou et al., 2023; Press et al., 2023) and planning (Yao et al., 2023; Wang et al., 2023). However, few works have focused on methods of representing the problem itself.

Human Problem Solving. The human problem-solving process has been modeled within information-processing theories of problem-solving in cognitive psychology (Polya, 1945; Newell, 1972; Gick, 1986; Bransford & Stein, 1993). Two processes are generally considered important: (a) generation of a problem representation, (b) solution searching and its implementation (Greeno, 1978; Simon, 1978). To solve a problem, humans first structure the problem from two perspectives, the given information (initial state) and the goal information (goal state), referred to as problem representation construction (Greeno, 1977). This representation allows the problem solver to search for a solution or invoke a schema activation (Gick, 1986). Solution searching and its implementation can be viewed as an active process that transforms the initial state into the goal state. This process can involve a variety of strategies, including solving by analogy, decomposition, planning, and means-end analysis (Polya, 1945; Simon, 1996; Gick, 1986). So far, in the reasoning in LLMs, only the solution searching process has been subject to exploration. To the best of our knowledge, our PRECOT is the first attempt to integrate problem representation into the reasoning process of LLMs, inspired by accumulated insights in cognitive psychology.

3 METHOD

As a cornerstone of the problem-solving process, humans first construct a problem representation, a structured interpretation of the problem that consists of the given information (initial state) and the goal information (goal state) (Greeno, 1977; Gick, 1986). To explore the implication of introducing this understanding process to reasoning in LLMs, we design a multi-stage prompting framework, PRECOT, and apply it to CoT prompting to reinforce it. PRECOT consists of two stages, *Problem Representation Construction* and *Solution Searching*. PRECOT builds problem representation in the earlier stage to help better generate reasoning steps in the subsequent stage (Figure 1). We implement PRECOT in both few-shot and zero-shot manners. Note that any training or fine-tuning is not required in our approach.

3.1 PROBLEM REPRESENTATION CONSTRUCTION

In this stage, every piece of information is collected from a given question and arranged to identify the initial state. In parallel, the objective (goal state) targeted by the question is refined in a concise form. To this end, the LLM is prompted with few-shot demonstrations (few-shot) or only instructions (zero-shot) to extract both states from the question. For the details of the prompts, please see Appendix A.1.

3.2 SOLUTION SEARCHING

After constructing the problem representation, we ask the LLM to solve the problem with the generated problem representation. As our main focus is to investigate the effectiveness of the problem representation in the reasoning of LLMs, we employ CoT prompting (Wei et al., 2022c; Kojima et al., 2022) as a solution searching method, which is simple and generic but powerful.

4 EXPERIMENTS

To evaluate the effectiveness of PRECoT, we conduct experiments with two different LLMs on benchmarks covering multiple reasoning domains. We compare our PRECoT with two existing approaches; Few-shot CoT (Wei et al., 2022c) and zero-shot CoT (Kojima et al., 2022). The results demonstrate that integrating problem representation into reasoning in LLMs makes notable performance improvement.

4.1 EXPERIMENTAL SETUP

Benchmarks. For comprehensive investigation, we use a wide range of tasks across three reasoning categories: (1) Arithmetic Reasoning, (2) Commonsense Reasoning, and (3) Symbolic Reasoning. These include 15 tasks from different sources, including Big-Bench Hard (Suzgun et al., 2023) that is a challenging subset of Big-Bench (Srivastava et al., 2023) (please see the sections in §4.2 for all the benchmarks we employ).

Language Models. To verify the generality of our approach while considering the CoT prompting tends to reliably work in LLMs (Wei et al., 2022b; Suzgun et al., 2023), we evaluate PRECoT using two Transformer (Vaswani et al., 2017) based large LMs with varying architectures, pre-training corpora and objectives, and instruction tuning methods. We only use the public engines of the models with greedy decoding and zero temperature for reproducibility.

- **PaLM 2:** We use a public engine of PaLM 2 (Anil et al., 2023), *text-bison-001*, through PaLM API. It is based on the scaled transformer architecture and pre-trained with various objectives following UL2 (Tay et al., 2023). Also, it is instruction-tuned with scaled FLAN procedure (Wei et al., 2022a; Chung et al., 2022; Longpre et al., 2023). Please note that the details of *text-bison-001* (e.g., exact model size) are not publicly available.
- **GPT-3:** We use a public engine of GPT-3 (Brown et al., 2020) with 175B parameters, *text-davinci-003*, through OpenAI API. It is a decoder-only auto-regressive transformer that inherits the architecture of GPT-2 (Radford et al., 2019). It is pre-trained with causal language modeling objective and instruction-tuned with reinforcement learning from human feedbacks (RLHF) (Christiano et al., 2017; Ouyang et al., 2022) using Proximal Policy Optimization (PPO) (Schulman et al., 2017).

Baselines. We compare our PRECoT with two representative approaches, few-shot and zero-shot CoT (Wei et al., 2022c; Kojima et al., 2022). For a fair comparison, we use the same chain of thought demonstrations as those used in previous works (Wei et al., 2022b; Suzgun et al., 2023) in the few-shot CoT and the solution searching stage of PRECoT. We manually annotate demonstrations to extract the given information and objective from the problem for few-shot PRECoT. We append “Let’s think step by step” to the end of all few-shot prompts for generating reasoning steps to improve the performance (Kojima et al., 2022; Fu et al., 2023). Note that zero-shot PRECoT is an automated framework; any manually designed demonstration is not required, as in zero-shot CoT.

4.2 RESULTS

We observe the three main points in the results. (1) Our PRECoT outperforms CoT baselines on most benchmarks in both few-shot and zero-shot manners. (2) Our method enables qualitatively better reasoning in different aspects, such as robust reasoning against irrelevant information and context sensitivity. (3) Performance gains from applying our method are broadly consistent across both language models, demonstrating the generalizability of our approach. Please refer to the following sections for detailed evaluations on different domains.

Table 1: Accuracy (%) on arithmetic reasoning tasks of PRECOT compared to CoT (we highlight in green the cases where our method outperforms CoT). Few-shot PRECOT outperforms few-shot CoT on all tasks, and zero-shot PRECOT outperforms zero-shot CoT on most tasks.

Model	Method	GSM8K	GSM-IC	SVAMP	AQuA	
PaLM 2 (N/A)	Few-shot	CoT	56.56	60.50	74.30	42.91
		PRECOT	61.79 (+5.23)	74.05 (+13.55)	78.20 (+3.90)	44.49 (+1.58)
	Zero-shot	CoT	56.86	61.90	72.30	41.73
		PRECOT	54.74 (-2.12)	67.85 (+5.95)	73.90 (+1.60)	45.67 (+3.94)
GPT-3 (175B)	Few-shot	CoT	60.73	72.70	79.30	48.82
		PRECOT	61.26 (+0.53)	73.80 (+1.10)	81.00 (+1.70)	51.97 (+3.15)
	Zero-shot	CoT	62.93	74.55	77.10	46.06
		PRECOT	60.35 (-2.58)	76.05 (+1.50)	79.20 (+2.10)	44.49 (-1.57)

Table 2: Output comparison of few-shot PRECOT and CoT on GSM-IC in PaLM 2. While CoT gets distracted by irrelevant information and deviates from the correct reasoning trajectory, PRECOT keeps on track to reach the correct answer.

Question: A magazine costs \$3 each. Jewel bought 10 magazines to be sold at \$3.50 each. Jewel’s mother bought 5 newspapers. How much will Jewel gain from selling her magazines? (Ground truth: 5)	
CoT (incorrect)	PRECOT (correct)
<p>Answer: Let’s think step by step. 10 magazines at \$3 each is $10 \times \\$3 = \\30. Jewel will sell her magazines at \$3.50 each so she will make $10 \times \\$3.50 = \\35. Jewel’s mother bought 5 newspapers at \$1 each so she spent $5 \times \\$1 = \\5. Jewel will gain $\\$35 - \\$5 = \\$30$ from selling her magazines. So the answer is 30.</p>	<p>Given Information: The given information is 10 magazines were bought at \$3 each, and will be sold at \$3.50 each. Jewel’s mother bought 5 newspapers.</p> <p>Objective: The question is asking for the amount of money Jewel will gain from selling her magazines.</p> <p>Answer: Let’s think step by step. Jewel bought 10 magazines at \$3 each, so that’s $10 * 3 = \\$30$. She will sell them at \$3.50 each, so that’s $10 * 3.5 = \\$35$. So she will gain $\\$35 - \\$30 = \\$5$. So the answer is 5.</p>

4.2.1 ARITHMETIC REASONING

Tasks. We consider the following four tasks for evaluation: (1) GSM8K (Cobbe et al., 2021), human-crafted grade school math word problems (MWP) with linguistic diversity. (2) GSM-IC (Shi et al., 2023), a variant of GSM8K that introduces irrelevant information to the questions for investigating the distractibility of LMs. Since its size is around 60K, we use a random subset of size 2K for cost efficiency. (3) SVAMP (Patel et al., 2021), elementary-level MWPs designed to necessitate models to consider problem context instead of relying on simple heuristics. (4) AQuA-RAT (Ling et al., 2017), multiple-choice algebraic word problems augmented by crowdsourcing from a seed set including GRE (Graduate Record Examinations).

Results. Table 1 shows the accuracy of our method (PRECOT) and the competing methods (CoT). Overall, PRECOT boosts the performance of CoT on most tasks. Particularly, few-shot PRECOT makes significant improvements on all tasks over both LLMs. This result strongly suggests that problem representation is effective for arithmetic reasoning. Additionally, zero-shot PRECOT also shows competitive results on most tasks, but its effectiveness is slightly inferior to few-shot PRECOT. We speculate that this is because the problem representation generated in a few-shot setting is qualitatively better than that in a zero-shot setting (further analysis can be found in §5.2).

Robustness to Irrelevant Information. As demonstrated in the results on GSM-IC and SVAMP, PRECOT makes LLMs robust to disturbances from irrelevant information. In other words, PRECOT is more capable of finding the desired answer by combining only the necessary information from

Table 3: Accuracy (%) on commonsense reasoning tasks of PRECoT compared to CoT. PRECoT outperforms CoT on most tasks, demonstrating notable improvement in more complex tasks.

Model	Method	StrategyQA	CSQA	SocialQA	Date	Causal Judg.	Ruin Names	
PaLM 2 (N/A)	Few-shot	CoT	72.53	81.65	77.23	58.00	54.01	67.20
		PRECoT	72.01 (-0.52)	79.12 (-2.53)	79.32 (+2.09)	66.80 (+8.80)	56.68 (+2.67)	68.80 (+1.60)
	Zero-shot	CoT	77.16	78.30	75.18	55.60	58.29	60.40
		PRECoT	73.32 (-3.84)	80.59 (+2.29)	77.07 (+1.89)	58.00 (+2.40)	60.43 (+2.14)	62.00 (+1.60)
GPT-3 (175B)	Few-shot	CoT	73.06	78.54	73.80	81.20	62.03	76.40
		PRECoT	71.97 (-1.09)	79.52 (+0.98)	74.05 (+0.25)	84.40 (+3.20)	62.57 (+0.54)	77.60 (+1.20)
	Zero-shot	CoT	64.02	69.37	71.80	70.80	61.50	47.20
		PRECoT	65.59 (+1.57)	70.93 (+1.56)	70.16 (-1.64)	69.20 (-1.60)	63.64 (+2.14)	59.60 (+12.40)

Table 4: Output comparison of few-shot PRECoT and CoT on Date in PaLM 2 (it is originally a multiple-choice task, but the options are omitted here for the demonstration). While CoT misses details of the context in the question, PRECoT is more sensitive to the details, enabling it to provide the correct answer.

Question: Jane scheduled 3 appointments with 5 people for tomorrow (Tue, 7/9/1972). What is the date one week ago from today in MM/DD/YYYY? (Ground truth: 07/01/1972)	
CoT (incorrect)	PRECoT (correct)
Answer: Let’s think step by step. 7/9/1972 - 7 days = 7/2/1972. So the answer is 07/02/1972.	Given Information: The given information is Jane scheduled 3 appointments with 5 people for tomorrow (Tue, 7/9/1972). Objective: The question is asking for a date one week ago from today in MM/DD/YYYY. Answer: Let’s think step by step. If tomorrow is 7/9/1972, then today is 7/8/1972. The date one week ago from today is 7/1/1972. So the answer is 07/01/1972.

distracting information. As depicted in Table 2, the problem representation enables the LLM to accurately trace the reasoning path from the initial state to the goal state without deviation.

4.2.2 COMMONSENSE REASONING

Tasks. We consider the following six tasks for evaluation: (1) StrategyQA (Geva et al., 2021), multi-hop open-domain yes-or-no questions. (2) CommonsenseQA (CSQA; Talmor et al. (2019)), multiple-choice questions that often require real-world prior knowledge. (3) SocialQA (Sap et al., 2019), multiple-choice questions about human behavior and emotional responses in a given situation. (4) Date Understanding (Date), multiple-choice questions that require a model to calculate the date using simple date arithmetic and real-world knowledge. (5) Causal Judgment, multiple-choice questions that ask typical answers to causal questions related to a given context. (6) Ruin Names, multiple-choice questions that ask a humorous edit of the artist or movie name. Date, Causal Judgment, and Ruin Names are a subset of Big-Bench Hard.

Results. Table 3 shows the accuracy of our methods (PRECoT) and the competing methods (CoT). Overall, PRECoT outperforms CoT in few-shot and zero-shot manners over both LLMs. As one of the notable points, few-shot PRECoT makes substantial performance improvement for all commonsense reasoning tasks in both LLMs except for StrategyQA and CSQA. Unlike the other tasks, they only rely on internal prior knowledge. Therefore, we manually investigate the outputs of the LLMs and observe that constructing a good problem representation for these tasks is challenging (please see Appendix A.2 for an example). The evaluation result and this finding suggest that problem representation benefits complex commonsense reasoning tasks with a specific context, such as SocialQA, Date, Causal Judgment, and Ruin Names.

Table 5: Accuracy (%) on symbolic reasoning tasks of PRECOT compared to CoT. PRECOT outperforms CoT on most tasks by a large margin.

Model	Method	Colors	Deduction	Tracking	Coin Flips	Last Letters	
PaLM 2 (N/A)	Few-shot	CoT	70.00	47.07	38.40	70.27	44.53
		PRECOT	74.80 (+4.80)	64.27 (+17.20)	42.93 (+4.53)	96.40 (+26.13)	43.86 (-0.67)
	Zero-shot	CoT	55.60	48.93	31.20	56.67	1.47
		PRECOT	64.40 (+8.80)	59.60 (+10.67)	36.27 (+5.07)	56.93 (+0.27)	16.53 (+15.06)
GPT-3 (175B)	Few-shot	CoT	86.40	66.40	78.53	85.20	60.80
		PRECOT	89.20 (+2.80)	64.27 (-2.13)	80.13 (+1.60)	98.67 (+13.47)	61.07 (+0.27)
	Zero-shot	CoT	69.20	54.40	36.80	98.40	51.07
		PRECOT	76.80 (+7.60)	60.53 (+6.13)	36.40 (-0.40)	92.27 (-6.13)	61.07 (+10.00)

Table 6: Output comparison of zero-shot PRECOT and CoT on Colors in PaLM 2 (it is originally a multiple-choice task, but the options are omitted here for the demonstration). While PRECOT follows the correct goal-oriented reasoning trajectory, CoT misunderstands the problem.

Question: On the nightstand, there are two green sheets of paper, two mauve cat toys, two green cat toys, and three mauve sheets of paper. If I remove all the cat toys from the nightstand, how many mauve things remain on it? (Ground truth: three)	
CoT (incorrect)	PRECOT (correct)
<p>Answer: Let’s think step by step. $2 + 2 = 4$ cat toys. $2 + 3 = 5$ mauve things. $5 - 4 = 1$ mauve thing remains on the nightstand. So the answer is two.</p>	<p>Given Information: The given information is 2 green sheets of paper, 2 mauve cat toys, 2 green cat toys, and 3 mauve sheets of paper on the nightstand.</p> <p>Objective: The question is asking how many mauve things remain on the nightstand after removing all the cat toys.</p> <p>Answer: Let’s think step by step. 2 mauve cat toys + 3 mauve sheets of paper = 5 mauve things. 5 mauve things - 2 mauve cat toys = 3 mauve things. So the answer is three.</p>

Context Sensitivity. Our method shows relatively higher context awareness in commonsense reasoning. For example, as shown in Table 4, CoT relies on simple heuristics to assemble information without regard to the context of the question, while PRECOT infers the appropriate answer based on the intent of the problem.

4.2.3 SYMBOLIC REASONING

Tasks. We consider the following five tasks for evaluation: (1) Reasoning about Colored Objects (Colors), multiple-choice questions about the consequence of taking a particular action on several portrayed objects. (2) Logical Deduction (Deduction), multiple-choice questions that ask for the order of objects based on the given relationship. (3) Tracking Shuffled Objects (Tracking), multiple-choice questions that ask for the final position of the object, given initial positions of objects and subsequent transformations (e.g., swaps). (4) Coin Flips, yes-or-no questions that ask for the final state of the coin after a few flips. (5) Last Letter Concatenation (Last Letters), questions that ask for the concatenation of the last letters from the given names. Colors, Deduction, and Tracking are a subset of Big-Bench Hard. Coin Flips and Last Letters are synthetically generated as described in Wei et al. (2022c).

Results. Table 5 shows the accuracy of our methods (PRECOT) and the competing methods (CoT). Overall, our method significantly boosts the performance of both LLMs by a large margin on most tasks in both few-shot and zero-shot settings. Notably, improvements in Colors are consistently observed across all models and settings. Table 6 shows how setting the initial and goal states helps. The reasoning steps in CoT are misaligned with the information embedded in the question, leading

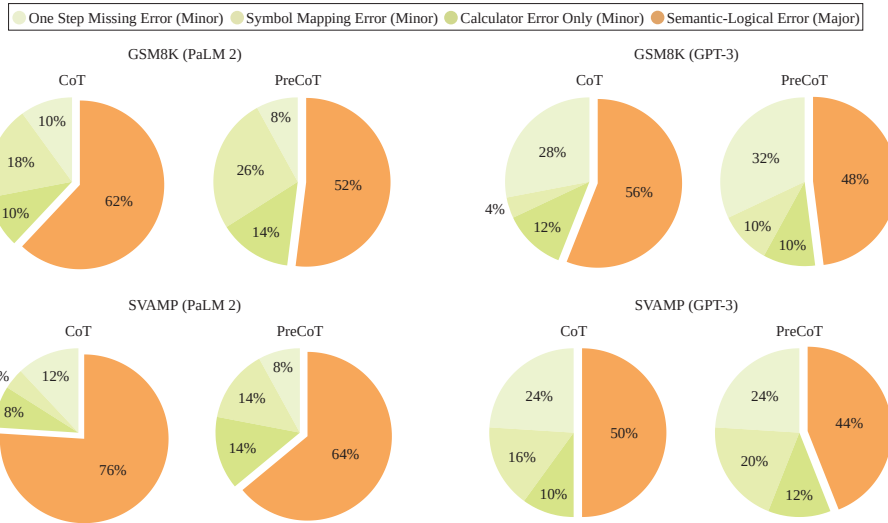


Figure 2: Distribution change in types of incorrect chain of thoughts evaluated on GSM8K and SVAMP in GPT-3 and PaLM 2. PRECoT reduces the major errors and makes it less severe.

them to deviate from the correct reasoning trajectory. In contrast, PRECoT demonstrates a proper reasoning process when presented with problem representation.

5 ANALYSIS

5.1 REASONING ERROR ANALYSIS

Evaluating the reasoning performance with accuracy might not be sufficient to measure the quality of the reasoning chains of LLMs when the generated answer is incorrect. Therefore, to explore the effectiveness of the problem representation in more detail, we conduct a manual analysis on the incorrect chain of thoughts on problems that a language model gets wrong. Specifically, for each LLM, we randomly sample 100 arithmetic reasoning problems (50 each from GSM8K and SVAMP), for which both CoT and PRECoT are incorrect. Then, we ask human evaluators to classify the types of incorrect chain of thoughts according to the criteria proposed by Wei et al. (2022c), which includes three minor error categories (one step missing, symbol mapping, and calculator error) and one major error category (semantic-logical error¹). As shown in Figure 2, introducing problem representation reduces the major errors and makes them less severe errors in both LLMs (please see the example in Appendix A.2). The trend suggests that problem representation substantially contributes to the reasoning in LLMs. It is consistent with our performance evaluation presented in earlier sections, revealing the potential of problem representation that accuracy could not reflect.

5.2 PROBLEM REPRESENTATION QUALITY AND REASONING PERFORMANCE

In arithmetic reasoning tasks, we have observed cases in which poorly constructed problem representations lead to incorrect reasoning. Most of these issues arise from omissions while extracting given information, as shown in Table 7 (note that in many cases, the problem representation is well generated). Those tasks tend to have a more intricate context than other tasks, in which entities and their corresponding numeric values are heavily interweaved. This complexity poses challenges for zero-shot PRECoT in fully extracting the given information. This drawback could make it challenging to investigate the contribution of problem representation to reasoning in LLMs, which is our primary goal in this study. To determine if problem representation is effective on its own, independent of its construction process, we adjust the experiment protocol; we re-evaluate the solution searching performance of zero-shot PRECoT with the problem representation constructed in few-shot PRECoT.

¹We merge two original major error categories, incoherent chain of thought and semantic understanding error, into semantic-logical error since they are closely related. Please see Appendix A.2 for the categorization detail.

Table 7: Output comparison of zero-shot PRECOT and zero-shot PRECOT+ on GSM8K in PaLM 2. Zero-shot PRECOT omits critical context when extracting the given information and fails to recognize the final objective clearly. On the contrary, zero-shot PRECOT+ can reach the correct answer, supported by the better-constructed problem presentation.

Zero-shot PRECOT (incorrect)	Zero-shot PRECOT+ (correct)
<p>Question: Jessica is trying to figure out how much to pay on all her debts each month. Her student loans have a minimum payment of \$300/month, her credit card’s minimum is \$200/month, and her mortgage’s minimum is \$500/month. If Jessica wants to pay 50% more than the minimum, how much does she pay in a year? (Ground truth: 18000)</p> <p>Given Information: The given information is 300, 200, 500, 50%.</p> <p>Objective: The question is asking 1. how much Jessica pays on her debts each month 2. how much Jessica pays in a year.</p> <p>Answer: Let’s think step by step. 1. Jessica pays $300 + 200 + 500 = \\$1000/\text{month}$. 2. Jessica pays $1000 * 12 = \\$12,000/\text{year}$. Thus, the answer is 12000.</p>	<p>Given Information: The given information is that Jessica has three debts with minimum payments of \$300, \$200, and \$500, and she wants to pay 50% more than the minimum.</p> <p>Objective: The question is asking how much Jessica pays in a year on all her debts.</p> <p>Answer: Let’s think step by step. $300 + 200 + 500 = \\$1000$. $1000 * .5 = \\$500$. $1000 + 500 = \\$1500$. Jessica pays \$1500 a month on her debts. $1500 * 12 = \\$18,000$. Jessica pays \$18,000 a year on her debts. Thus, the answer is 18000.</p>

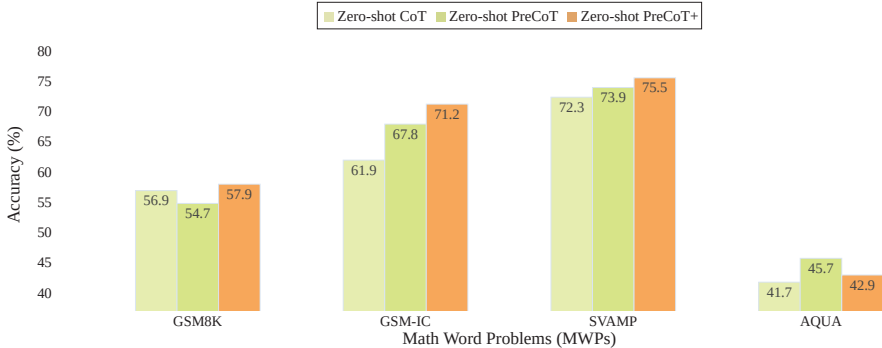


Figure 3: Zero-shot reasoning accuracy with problem representation generated with few-shot demonstrations (zero-shot PRECOT+). Zero-shot PRECOT+ tends to boost the performance of zero-shot PRECOT and outperforms zero-shot CoT on all tasks.

We refer to this as zero-shot PRECOT+. Figure 3 shows that the problem representation generated in the few-shot setting further enhances the overall zero-shot reasoning performance. Notably, in GSM8K, our zero-shot PRECOT+ now outperforms zero-shot CoT while it does not when problem representation is constructed in a zero-shot manner. These findings suggest that a better-constructed problem representation can more effectively contribute to the reasoning process.

6 CONCLUSION

We propose a novel approach called PRECOT that enhances the reasoning in LLMs by incorporating problem representation, a cornerstone in human problem-solving. The extensive evaluation on 15 benchmarks across three reasoning categories demonstrates PRECOT outperforms CoT on most tasks in both few-shot and zero-shot fashions. Furthermore, a qualitative error analysis indicates that PRECOT reduces major reasoning errors. Additionally, the improvement in zero-shot PRECOT+ shows that well-constructed problem representations enhances reasoning performance, implying that introducing problem representations is an effective approach. We hope our insights will inspire future work exploring the potential of language models for reasoning.

REPRODUCIBILITY STATEMENT

For reproducibility, we attach our implementation of PRECOT, evaluation code bases, and all prompts as supplementary materials. Our results can be reproduced using the attached codes and aforementioned public APIs.

REFERENCES

- Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Keane, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Mousaleem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. Palm 2 technical report, 2023.
- J.D. Bransford and B.S. Stein. *The Ideal Problem Solver: A Guide for Improving Thinking, Learning, and Creativity*. W.H. Freeman, 1993. ISBN 9780716722052. URL <https://books.google.co.kr/books?id=nnRxQgAACAAJ>.
- 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. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf.
- 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 Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben 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 Garcia, 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 Sankaranarayanan Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways, 2022.

- Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 4299–4307, 2017. URL <https://proceedings.neurips.cc/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html>.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416, 2022. doi: 10.48550/arXiv.2210.11416. URL <https://doi.org/10.48550/arXiv.2210.11416>.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021. URL <https://arxiv.org/abs/2110.14168>.
- J. E. Davidson. The role of metacognition in problem solving. In *Metacognition*. The MIT Press, 1994. doi: 10.7551/mitpress/4561.003.0012. URL <https://doi.org/10.7551/mitpress/4561.003.0012>.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL <https://aclanthology.org/N19-1423>.
- Ruilu Fu, Han Wang, Xuejun Zhang, Jun Zhou, and Yonghong Yan. Decomposing complex questions makes multi-hop QA easier and more interpretable. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 169–180, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.17. URL <https://aclanthology.org/2021.findings-emnlp.17>.
- Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. Complexity-based prompting for multi-step reasoning. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=yflicZHC-19>.
- 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. *Trans. Assoc. Comput. Linguistics*, 9:346–361, 2021. doi: 10.1162/tacl_a_00370. URL https://doi.org/10.1162/tacl_a_00370.
- Mary L. Gick. Problem-solving strategies. *Educational Psychologist*, 21(1-2):99–120, 1986. doi: 10.1080/00461520.1986.9653026.
- James G. Greeno. Process of understanding in problem solving. *Cognitive Theory*, 2:43–83, 1977.
- James G Greeno. Natures of problem-solving abilities. *Handbook of learning and cognitive processes*, 5:239–270, 1978.
- Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model, 2023.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020.

- Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. Decomposed prompting: A modular approach for solving complex tasks. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=_nGgzQjzaRy.
- Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 22199–22213. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/8bb0d291acd4acf06ef112099c16f326-Paper-Conference.pdf.
- 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)*, pp. 158–167, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1015. URL <https://aclanthology.org/P17-1015>.
- Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9):195:1–195:35, 2023. doi: 10.1145/3560815. URL <https://doi.org/10.1145/3560815>.
- Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. The flan collection: Designing data and methods for effective instruction tuning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 22631–22648. PMLR, 2023. URL <https://proceedings.mlr.press/v202/longpre23a.html>.
- Allen Newell. *Human Problem Solving*. Prentice-Hall, Inc., USA, 1972. ISBN 0134454030.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.
- Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2080–2094, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021-naacl-main.168. URL <https://aclanthology.org/2021.naacl-main.168>.
- G. Polya. *How to Solve It*. Princeton University Press, Princeton, 1945. ISBN 9781400828678. doi: doi:10.1515/9781400828678. URL <https://doi.org/10.1515/9781400828678>.
- Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models, 2023.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu,

- Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d’Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorraine Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. Scaling language models: Methods, analysis & insights from training gopher, 2022.
- Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4463–4473, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1454. URL <https://aclanthology.org/D19-1454>.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL <http://arxiv.org/abs/1707.06347>.
- Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H. Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 31210–31227. PMLR, 2023. URL <https://proceedings.mlr.press/v202/shi23a.html>.
- Herbert A Simon. Information-processing theory of human problem solving. *Handbook of learning and cognitive processes*, 5:271–295, 1978.
- Herbert A. Simon. *The Sciences of the Artificial (3rd Ed.)*. MIT Press, Cambridge, MA, USA, 1996. ISBN 0262691914.
- Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Johan Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinon, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, Cesar Ferri, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Christopher Waites, Christian Voigt, Christopher D Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, C. Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Chen, Dieuweke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodolà, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang,

Ethan A Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Xinyue Wang, Gonzalo Jaimovitch-Lopez, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Francis Anthony Shevlin, Hinrich Schuetze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B Simon, James Koppel, James Zheng, James Zou, Jan Kocon, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh Dhole, Kevin Gimpel, Kevin Omondi, Kory Wallace Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Ludheng He, Luis Oliveros-Colón, Luke Metz, Lütfi Kerem Senel, Maarten Bosma, Maarten Sap, Maartje Ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramirez-Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael Andrew Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swedrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimeo Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan Andrew Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nishit Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter W Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan Le Bras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Russ Salakhutdinov, Ryan Andrew Chi, Seungjae Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel Stern Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima Shammie Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven Piantadosi, Stuart Shieber, Summer Misherggi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsunori Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Venkatesh Ramasesh, vinay uday prabhu, Vishakh Padmakumar, Vivek Sriku-mar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Song, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Tong, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang, and Ziyi Wu. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL <https://openreview.net/forum?id=uyTL5Bvosj>.

- Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, Toronto, Canada, July 2023. Association for Computational Linguistics. URL <https://aclanthology.org/2023.findings-acl.824>.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4149–4158, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1421. URL <https://aclanthology.org/N19-1421>.
- Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. UL2: Unifying language learning paradigms. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=6ruVLB727MC>.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2609–2634, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.147. URL <https://aclanthology.org/2023.acl-long.147>.
- Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022a. URL <https://openreview.net/forum?id=gEZrGCozdqR>.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022b. ISSN 2835-8856. URL <https://openreview.net/forum?id=yzkSU5zdwD>. Survey Certification.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837. Curran Associates, Inc., 2022c. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models, 2023.
- Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=WZH7099tgfM>.

A APPENDIX

A.1 PRECOT IMPLEMENTATION DETAIL

Problem Representation Construction. We use *task demonstrations* (few-shot PRECOT) or *instructions* (zero-shot PRECOT) with generation *triggers* to extract the given information and objective from a problem. We format the prompt as shown in Table 8 with its components in Table 10.

Solution Searching. We employ few-shot and zero-shot CoT prompting (Wei et al., 2022c; Kojima et al., 2022) for generating reasoning steps. We format the prompts as shown in Table 9. Note that the task demonstrations are only used in few-shot PRECOT. We extract LLMs’ predictions (predicted label) from generated reasoning steps using a manually designed rule-based output cleanser. We attach evaluation code bases in the supplementary material. Only for zero-shot PRECOT, we additionally extract their predictions with LLMs and then apply the output cleanser following Kojima et al. (2022).

Table 8: Prompt format for generating problem representation in PRECOT.

<pre>{A task demonstration or an instruction} Question: {Question} {Trigger}</pre>

Table 9: Prompt format for generating reasoning steps in PRECOT.

<pre>{A task demonstration (few-shot PRECOT only)} Question: {Question} Given Information: {Extracted information in the previous stage.} Objective: {Extracted objective in the previous stage.} Answer: Let’s think step by step.</pre>
--

A.2 ADDITIONAL ERROR ANALYSIS

Arithmetic Reasoning. We manually investigate the incorrect chain of thoughts generated by both LLMs within few-shot CoT and PRECOT on GSM8K and SVAMP. We largely follow the criteria proposed by Wei et al. (2022c) for a fair comparison, but we slightly modify the major error categorization. Specifically, we merge two original major error categories, incoherent chain of thought and semantic understanding error, into semantic-logical error. This is because these two categories are not independent, which makes it very difficult to distinguish between them. The concise definitions of each error category are as follows. For a detailed description of these criteria and examples, please refer to the previous work (Wei et al., 2022c).

- **Calculator Error Only (minor):** In this category, attaining the correct answer is possible by rectifying all calculation errors present in the reasoning chain.
- **Symbol Mapping Error (minor):** In this category, rectifying the numeric symbols in the equation without modifying the words in the reasoning chain makes an utterly correct explanation.
- **One Step Missing Error (minor):** In this category, the path to the correct answer necessitates the inclusion of one logical step in the reasoning chain.
- **Semantic-Logical Error (major):** Unlike the minor errors above, this category represents a scenario where the reasoning chain contains severe semantic and logical flaws, making the attainment of the correct answer impossible through simple modifications.

As delineated in the analysis section, PRECOT successfully downgrades some major errors in the incorrect chain of thoughts to minor ones. In the example in Table 11, CoT exhibits a significant logical flaw, whereas PRECOT only makes a computational mistake.

Table 10: Details of the components that comprise the prompt for constructing the problem representation in PRECOT.

Task	Component	Content
Given Information Extraction	Task demonstration (few-shot PRECOT)	Please refer §A.4 to see full prompt for each tasks.
	Instruction (zero-shot PRECOT)	Extract what information is given in the following question to help answer it.
	Trigger	Given Information: The given information is
Objective Extraction	Task demonstration (few-shot PRECOT)	Please refer §A.4 to see full prompt for each tasks.
	Instruction (zero-shot PRECOT)	Write what the following question is asking for to help answer it.
	Trigger	Objective: The question is asking

Table 11: Output comparison of few-shot PRECOT and CoT on GSM8K in PaLM 2. PRECOT only exhibit a minor calculation error, while CoT has a logical flaw and misses a critical reasoning step.

Question: Hannah slips on a banana peel and breaks her arm. The doctor charges her \$200 for the cast, \$300/hour for a 30-minute visit, \$4/pill for 30 painkillers, and \$6/hour for a mandatory minimum of 2 hours of parking. How much does the doctor’s visit cost total? (Ground truth: 482)	
CoT (Semantic-Logical Error)	PRECOT (Calculation Error)
<p>Answer: Let’s think step by step. $300 / 60 = 5$ dollars per 30 minutes. $5 * 30 = \\$150$ for the visit. $30 * 4 = \\$120$ for the painkillers. $150 + 120 + 200 = \\$470$. So the answer is 470.</p>	<p>Given Information: The given information is that Hannah’s doctor charges \$200 for the cast, \$300/hour for a 30-minute visit, \$4/pill for 30 painkillers, and \$6/hour for a mandatory minimum of 2 hours of parking.</p> <p>Objective: The question is asking for the total cost of the doctor’s visit.</p> <p>Answer: Let’s think step by step. The cast costs \$200. The visit costs $300 / 60 = \\$5 / \text{minute}$. The visit costs $5 * 30 = \\$150$. The painkillers cost $4 * 30 = \\$120$. The parking costs $6 * 2 = \\$12$. The total cost is $200 + 150 + 120 + 12 = \\582. So the answer is 582.</p>

Commonsense Reasoning. Tasks like StrategyQA and CSQA, which have no in-question information to reference and only rely on internal prior knowledge, are less likely to have well-constructed problem representations. These problem representations are often observed in the cases where the language model reaches incorrect answers, suggesting that problem representation might be less helpful to this type of question, as shown in Table 12.

A.3 DATASET DETAILS

We list the following 15 benchmarks we use in the experiments to aid reproducibility. Coin Flips and Last Letters are synthetically generated as described in previous works (Wei et al., 2022c; Kojima et al., 2022). Since the size of GSM-IC is about 60K, we use random subsets of size 2K for cost efficiency.

A.4 PROMPTS

We provide all prompts we use in the main experiment to aid reproducibility. In the solution searching stage of few-shot PRECOT, we use the same chain of thoughts as those used in previous

Table 12: Output example of zero-shot PRECOT on StrategyQA in PaLM 2. StrategyQA has no information to extract and relies solely on internal prior knowledge, making it hard for PRECOT to construct problem representation.

Question: Can vitamin C rich fruits be bad for health? (Ground truth: yes)
PRECOT (incorrect)
Given Information: The given information is ‘vitamin C rich fruits’.
Objective: The question is asking whether vitamin C rich fruits can be bad for health.
Answer: Let’s think step by step. Vitamin C rich fruits are good for health. So the final answer is no.

Table 13: Details about the benchmark tasks employed in our experiments.

Category	Dataset	Size	Split
Arithmetic Reasoning	GSM8K (Cobbe et al., 2021)	1,319	test
	GSM-IC (Shi et al., 2023)	2,000	2/m-step
	SVAMP (Patel et al., 2021)	1,000	test
	AQuA (Ling et al., 2017)	254	test
Commonsense Reasoning	CommonsenseQA (Talmor et al., 2019)	1,221	dev (rand)
	StrategyQA (Geva et al., 2021)	2,289	Big-Bench (Srivastava et al., 2023)
	SocialIQA (Sap et al., 2019)	1,953	dev
	Date Understanding (Srivastava et al., 2023)	250	Big-Bench Hard (Suzgun et al., 2023)
	Causal Judgement (Srivastava et al., 2023)	250	
	Ruin Names (Srivastava et al., 2023)	250	
Reasoning about Colored Objects (Srivastava et al., 2023)	250		
Symbolic Reasoning	Logical Deduction (Srivastava et al., 2023)	750	Big-Bench Hard (Suzgun et al., 2023)
	Tracking Shuffled Objects (Srivastava et al., 2023)	750	
	Coin Flips (Wei et al., 2022c)	750	
	Last Letters (Wei et al., 2022c)	750	3/5/7

works (Wei et al., 2022c; Suzgun et al., 2023) for a fair comparison². In this paper, we only attach the prompts we use for arithmetic reasoning tasks, including GSM8K, GSM-IC, and SVAMP³. Please see Table 14- 21 and refer to the supplementary material for all other prompts.

²SocialIQA is an exception, as there is no CoT prompt available. Therefore, we manually annotated few-shot prompts, including the solution searching stage.

³For AQuA-RAT, please refer to the supplementary materials.

Table 14: Prompt for generating reasoning steps in zero-shot CoT.

Question: {Question} Answer: Let's think step by step.

Table 15: Prompt for extracting given information from a question in zero-shot PRECOT.

Extract what information is given in the following question to help answer it. Question: {Question} Given Information: The given information is

Table 16: Prompt for extracting an objective from a question in zero-shot PRECOT.

Write what the following question is asking for to help answer it. Question: {Question} Objective: The question is asking

Table 17: Prompt for generating reasoning steps in zero-shot PRECOT.

Question: {Question} Given Information: {Extracted information in the previous stage.} Objective: {Extracted objective in the previous stage.} Answer: Let's think step by step.

Table 18: Prompt for generating reasoning steps in arithmetic reasoning tasks in few-shot CoT. We use the same questions and the chain of thoughts as Wei et al. (2022c) for a fair comparison. Also, to improve performance, we add “Let’s think step by step” to the beginning of the chain of thoughts.

<p>Question: 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 will the grove workers plant today?</p> <p>Answer: Let’s think step by step. There are 15 trees originally. Then there were 21 trees after some more were planted. So there must have been $21 - 15 = 6$. So the answer is 6.</p> <p>Question: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?</p> <p>Answer: Let’s think step by step. There are originally 3 cars. 2 more cars arrive. $3 + 2 = 5$. So the answer is 5.</p> <p>Question: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?</p> <p>Answer: Let’s think step by step. Originally, Leah had 32 chocolates. Her sister had 42. So in total they had $32 + 42 = 74$. After eating 35, they had $74 - 35 = 39$. So the answer is 39.</p> <p>Question: Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How many lollipops did Jason give to Denny?</p> <p>Answer: Let’s think step by step. Jason started with 20 lollipops. Then he had 12 after giving some to Denny. So he gave Denny $20 - 12 = 8$. So the answer is 8.</p> <p>Question: Shawn has five toys. For Christmas, he got two toys each from his mom and dad. How many toys does he have now?</p> <p>Answer: Let’s think step by step. Shawn started with 5 toys. If he got 2 toys each from his mom and dad, then that is 4 more toys. $5 + 4 = 9$. So the answer is 9.</p> <p>Question: 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?</p> <p>Answer: Let’s think step by step. There were originally 9 computers. For each of 4 days, 5 more computers were added. So $5 * 4 = 20$ computers were added. $9 + 20$ is 29. So the answer is 29.</p> <p>Question: 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?</p> <p>Answer: Let’s think step by step. Michael started with 58 golf balls. After losing 23 on tuesday, he had $58 - 23 = 35$. After losing 2 more, he had $35 - 2 = 33$ golf balls. So the answer is 33.</p> <p>Question: Olivia has \$23. She bought five bagels for \$3 each. How much money does she have left?</p> <p>Answer: Let’s think step by step. Olivia had 23 dollars. 5 bagels for 3 dollars each will be $5 * 3 = 15$ dollars. So she has $23 - 15$ dollars left. $23 - 15$ is 8. So the answer is 8.</p>
--

Table 19: Prompt for extracting given information from the problems in arithmetic reasoning tasks in few-shot PRECOT.

<p>Question: 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 will the grove workers plant today?</p> <p>Given Information: The given information is that the workers will add trees to the grove today. There were originally 15 trees in the grove, and there should be 21 after they are done.</p>
<p>Question: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?</p> <p>Given Information: The given information is that two more cars arrive at the parking lot with three cars.</p>
<p>Question: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?</p> <p>Given Information: The given information is that Leah and her sister initially had 32 and 42 chocolates, respectively, and together ate 35.</p>
<p>Question: Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How many lollipops did Jason give to Denny?</p> <p>Given Information: The given information is that Jason initially had 20 lollipops, and after giving some to Denny, it was reduced to 12.</p>
<p>Question: Shawn has five toys. For Christmas, he got two toys each from his mom and dad. How many toys does he have now?</p> <p>Given Information: The given information is that Shawn initially had 5 toys and received 2 from each of his parents.</p>
<p>Question: 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?</p> <p>Given Information: The given information is that the server room with nine computers had five additional computers installed daily, Monday through Thursday.</p>
<p>Question: 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?</p> <p>Given Information: The given information is that Michael initially had 57 golf balls, but he lost 23 on Tuesday and two more on Wednesday.</p>
<p>Question: Olivia has \$23. She bought five bagels for \$3 each. How much money does she have left?</p> <p>Given Information: The given information is that Olivia initially had \$23, and she bought five bagels for \$3 each.</p>

Table 20: Prompt for extracting objective from the problems in arithmetic reasoning tasks in few-shot PRECOT.

<p>Question: 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 will the grove workers plant today?</p> <p>Objective: The question is asking for the number of trees the workers must add to the grove today.</p>
<p>Question: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?</p> <p>Objective: The question is asking for the number of cars will be in the parking lot.</p>
<p>Question: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?</p> <p>Objective: The question is asking for the number of chocolates left after they eat.</p>
<p>Question: Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How many lollipops did Jason give to Denny?</p> <p>Objective: The question is asking for the number of lollipops Jason gave to Denny.</p>
<p>Question: Shawn has five toys. For Christmas, he got two toys each from his mom and dad. How many toys does he have now?</p> <p>Objective: The question is asking for the number of toys Shawn owns after receiving the gift.</p>
<p>Question: 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?</p> <p>Objective: The question is asking for the total number of computers in the server room after installing additional computers.</p>
<p>Question: 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?</p> <p>Objective: The question is asking for the number of balls Michael has at the end of Wednesday.</p>
<p>Question: Olivia has \$23. She bought five bagels for \$3 each. How much money does she have left?</p> <p>Objective: The question is asking how much money Olivia has left after buying bagels.</p>

Table 21: Prompt for generating reasoning steps for arithmetic reasoning tasks in few-shot PRECoT. This prompt is a combination of the three prompts presented earlier (Table 18, 19, 20).

<p>Question: 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 will the grove workers plant today?</p> <p>Given Information: The given information is that the workers will add trees to the grove today. There were originally 15 trees in the grove, and there should be 21 after they are done.</p> <p>Objective: The question is asking for the number of trees the workers must add to the grove today.</p> <p>Answer: Let's think step by step. There are 15 trees originally. Then there were 21 trees after some more were planted. So there must have been $21 - 15 = 6$. So the answer is 6.</p> <p>Question: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?</p> <p>Given Information: The given information is that two more cars arrive at the parking lot with three cars.</p> <p>Objective: The question is asking for the number of cars will be in the parking lot.</p> <p>Answer: Let's think step by step. There are originally 3 cars. 2 more cars arrive. $3 + 2 = 5$. So the answer is 5.</p> <p>Question: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?</p> <p>Given Information: The given information is that Leah and her sister initially had 32 and 42 chocolates, respectively, and together ate 35.</p> <p>Objective: The question is asking for the number of chocolates left after they eat.</p> <p>Answer: Let's think step by step. Originally, Leah had 32 chocolates. Her sister had 42. So in total they had $32 + 42 = 74$. After eating 35, they had $74 - 35 = 39$. So the answer is 39.</p> <p>Question: Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How many lollipops did Jason give to Denny?</p> <p>Given Information: The given information is that Jason initially had 20 lollipops, and after giving some to Denny, it was reduced to 12.</p> <p>Objective: The question is asking for the number of lollipops Jason gave to Denny.</p> <p>Answer: Let's think step by step. Jason started with 20 lollipops. Then he had 12 after giving some to Denny. So he gave Denny $20 - 12 = 8$. So the answer is 8.</p> <p>Question: Shawn has five toys. For Christmas, he got two toys each from his mom and dad. How many toys does he have now?</p> <p>Given Information: The given information is that Shawn initially had 5 toys and received 2 from each of his parents.</p> <p>Objective: The question is asking for the number of toys Shawn owns after receiving the gift.</p> <p>Answer: Let's think step by step. Shawn started with 5 toys. If he got 2 toys each from his mom and dad, then that is 4 more toys. $5 + 4 = 9$. So the answer is 9.</p> <p>Question: 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?</p> <p>Given Information: The given information is that the server room with nine computers had five additional computers installed daily, Monday through Thursday.</p> <p>Objective: The question is asking for the total number of computers in the server room after installing additional computers.</p> <p>Answer: Let's think step by step. There were originally 9 computers. For each of 4 days, 5 more computers were added. So $5 * 4 = 20$ computers were added. $9 + 20$ is 29. So the answer is 29.</p> <p>Question: 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?</p> <p>Given Information: The given information is that Michael initially had 57 golf balls, but he lost 23 on Tuesday and two more on Wednesday.</p> <p>Objective: The question is asking for the number of balls Michael has at the end of Wednesday.</p> <p>Answer: Let's think step by step. Michael started with 58 golf balls. After losing 23 on tuesday, he had $58 - 23 = 35$. After losing 2 more, he had $35 - 2 = 33$ golf balls. So the answer is 33.</p> <p>Question: Olivia has \$23. She bought five bagels for \$3 each. How much money does she have left?</p> <p>Given Information: The given information is that Olivia initially had \$23, and she bought five bagels for \$3 each.</p> <p>Objective: The question is asking how much money Olivia has left after buying bagels.</p> <p>Answer: Let's think step by step. Olivia had 23 dollars. 5 bagels for 3 dollars each will be $5 * 3 = 15$ dollars. So she has $23 - 15$ dollars left. $23 - 15$ is 8. So the answer is 8.</p>
--