Optimizing Language Model's Reasoning Abilities with Weak Supervision

Anonymous ACL submission

Abstract

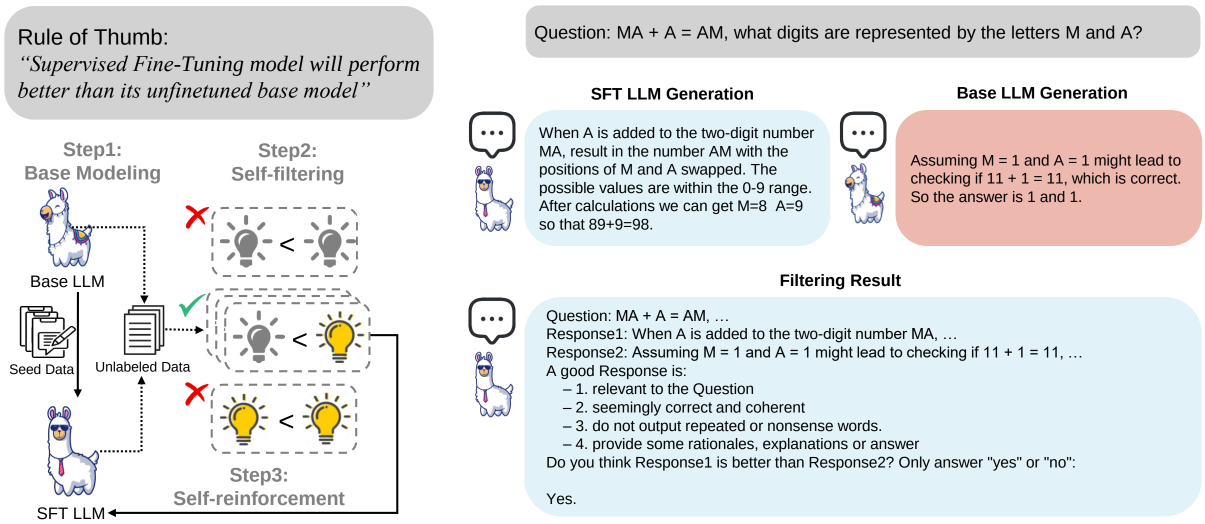
While Large Language Models (LLMs) have demonstrated proficiency in handling complex reasoning, much of the past work has depended on extensively annotated datasets by human experts. However, this reliance on fully-supervised annotations poses scalability challenges, particularly as models and data requirements grow. In this work, we begin by analyzing the limitations of existing data-efficient reinforcement learning (RL) methods in LLMs' reasoning enhancement. To mitigate this, we introduce self-reinforcement, an efficient weak-to-strong approach to optimize language models' reasoning abilities utilizing both annotated and unlabeled samples. Our method enhances the quality of synthetic feedback by fully harnessing annotated seed data and introducing a novel self-filtering mechanism to remove invalid pairs. We also present PUZZLEBEN, a weakly supervised benchmark for reasoning that comprises 25,147 complex questions, answers, and human-generated rationales across various domains, such as brainteasers, puzzles, riddles, parajumbles, and critical reasoning tasks. Our experiments underscore the significance of PUZZLEBEN, as well as the effectiveness of our methodology as a promising direction in future endeavors. Our dataset and code will be published soon on `Anonymity Link`.

1 Introduction

Large language models (LLMs) (Brown et al., 2020; Zhang et al., 2022a; Chowdhery et al., 2022; Touvron et al., 2023) with Chain-of-Thought (CoT)-based prompting (Wei et al., 2022; Wang et al., 2022; Yao et al., 2024; Besta et al., 2024) have demonstrated strong capabilities across various tasks and applications. To further enhance LLMs' reasoning capabilities, many previous work have relied on extensive datasets fully annotated by human experts (Longpre et al., 2023; Zhang et al., 2022b; Ranaldi and Freitas, 2024) or rationale distilled from larger models (Wang et al., 2023; Kim et al., 2023). This reliance, while beneficial for model training, presents significant scalability and availability challenges, particularly given the data requirement scale with the size of the LLMs.

Recent studies have demonstrated that Reinforcement Learning (RL), coupled with heuristic feedback, can bolster the reasoning capabilities of LLMs with only a few annotations (Luong et al., 2024; Feng et al., 2024; Tan et al., 2024). These approaches can be roughly categorized into two types: rule-based and self-construction. The rule-based (Luong et al., 2024) method devises a group of criteria to determine the reward assigned to the specific reasoning process. In contrast, the self-construction method (Feng et al., 2024) tends to construct the pair of reasoning samples in different qualities based on different assumptions about the factors influencing quality. While the abovementioned techniques alleviate the availability issue in reasoning enhancement, they still suffer from several limitations. Firstly, the **inflexibility and lack of comprehensiveness** in rule-based reward assignments can aggravate inherent problems in LLMs, such as bias (Casper et al., 2023) and reward hacking (Chen et al., 2024a; Jinnai et al., 2024). Secondly, while preference feedback produced by self-construction aligns with human intuition, there are still instances where **assumptions may not hold valid**. For example, while Feng et al. (2024) reasonably assume that reasoning samples leading to correct answers should be superior to those leading to incorrect ones, there exist scenarios where rigorous reasoning may only err in the final step, or a poor rationale may coincidentally yield a correct answer. Besides, both types of approaches **fail to fully exploit seed annotations** in datasets, which, though sparse, have been proven valuable and crucial in weakly-supervised scenarios(Zhou, 2018).

To address the challenges and foster LLMs' reasoning abilities with weak supervision, we intro-



(a). Overview Pipeline **(b). Self-filtering**

Figure 1: The overview pipeline of our methods, self-reinforcement and the detailed implementation of self-filtering in our methodology. This is an iterative weak-to-strong learning framework that intends to improve LLMs' reasoning under weak supervision. Blue indicates this response comes from strong models while red is from weaker models.

duce self-reinforcement in this work. Our methodology unfolds in three phases: initial base modeling, self-filtering, and self-reinforcement. In the base modeling stage, we hypothesize that the Supervised Fine-Tuned (SFT) model shows better performance compared to its unfinetuned counterpart when addressing unlabeled questions. Thus, we train the model in the seed annotation data and build comparisons using the response from the SFT LLM and base LLM. This tuning-based approach intuitively maximizes the utilization of seed annotations, thereby potentially yielding response pairs with more substantial quality distinctions compared to other self-construction methods. During the second phase, We borrow insights from recent self-judging methods (Yuan et al., 2024; Pang et al., 2024), proposing a self-filtering step where the LLM evaluates and eliminates undesirable response pairs to further ensure the quality of pairwise feedback. In the reinforcement learning phase, we use Direct Preference Optimization (DPO) (Rafailov et al., 2023) to refine the models by learning from the quality differences between their responses to the unlabeled question set. Noticable, our self-reinforcement allows iterative self-improvement while reducing the reliance on extensively annotated datasets.

Besides, we collect and introduce PUZZLEBEN, a weakly-supervised reasoning benchmark specifically designed to support and validate the effectiveness of weak-to-strong (Burns et al., 2023) learning paradigms. PUZZLEBEN encompasses a diverse collection of 25,147 labeled questions with answers and meticulously designed human rationale references, as well as 10,000 unlabeled questions. It consists of various problem types, including brainteasers, puzzles, riddles, parajumbles, and critical reasoning tasks. The presence of both annotated and unannotated questions within PUZZLEBEN enables the practical application of our self-reinforcement strategies. Additionally, the brainteaser subset in PUZZLEBEN features with human-labeled difficulty and fun scores, which could be used for further in-depth analysis.

Our experiments in PUZZLEBEN highlight the significant impact of human-annotated rationales and diverse problem types within PUZZLEBEN, as well as the efficacy of self-reinforcement in future reasoning work. There is also a significant observation that the current models' perception of difficulty in reasoning tasks does not always align with human perceptions, highlighting a potential area for further superalignment in the field of reasoning.

To sum up, our contribution can be summarized into the following aspects:

- We expose the limitations of previous RL-based data-efficient methods in enhancing the LLMs' reasoning abilities and propose our self-reinforcement tailored for weakly-supervised reasoning learning.
- We build PUZZLEBEN, a comprehensive weakly-supervised reasoning benchmark consisting of

various problem types.

- With extensive experiments conducted, we validate the effectiveness of our method and propose further hints and guidance on LLM's reasoning.

2 Related Work

LLMs' Reasonings CoT (Wei et al., 2022) equips LLMs with enhanced reasoning capabilities, leading to a series of subsequent studies (Wang et al., 2022; Zhou et al., 2022; Creswell and Shanahan, 2022; Besta et al., 2023; Li et al., 2023; Lightman et al., 2023) that simulate human logical processes. These methods are applied across various reasoning tasks, including commonsense (Geva et al., 2021; Zhao et al., 2024), logical (Pan et al., 2023; Lei et al., 2023), and mathematical reasoning (Cobbe et al., 2021; Hendrycks et al., 2021). To enhance LLMs' reasoning, training on annotated reasoning datasets (Longpre et al., 2023; Zhang et al., 2022b; Ranaldi and Freitas, 2024) and distilling from larger models (Wang et al., 2023; Kim et al., 2023) are two common ways. However, these two methods suffer from resource availability and that stimulates our motivation to explore data-efficient and self-powered methods to boost LLMs' reasoning abilities.

Reinforcement Learning Proximal Policy Optimization (PPO) (Schulman et al., 2017) is a key RL technique for aligning models with human preferences (Ouyang et al., 2022). They further lead to the development of Direct Preference Optimization (DPO) (Rafailov et al., 2023), which uses the LLM as an implicit reward model. Recent efforts are exploring the use of reinforcement learning in tasks that involve reasoning. For example, Luong et al. (2024) adopts PPO to differentiate between correct and incorrect reasoning explanations, requiring a large corpus of human-annotated golden references. Feng et al. (2024) propose self-motivated learning by training the reward model with synthetic feedback produced from the policy.

Self-training and Self-improvement Many previous works in this direction assign a pseudo label from a learned classifier to further improve the base model (Xie et al., 2020; RoyChowdhury et al., 2019; Chen et al., 2021). Huang et al. (2022) propose utilizing language models to self-improve without supervised data. Chen et al. (2024b) employing LLMs from earlier iterations along with human-annotated SFT data to refine the models. They contrast data decoded by the models with data supervised by humans and learn from this comparison, which still necessitates considerable human effort. Although our work shares similar insights with this direction, we intend to unveil the potential to supervise strong models with a weak model in the field of reasoning.

Weak-to-strong Learning and Generalizations Burns et al. (2023) introduces the potential of leveraging weak model supervision to elicit the full capabilities of much stronger models for superalignment in the future. Following this trend, our work tends to explore how to improve LLMs' reasoning abilities under weakly low-resource supervision. This direction is significant when humans cannot provide large-scale confident answers when the questions become too hard.

Weakly-supervised Learning Many previous works in this field concern about how to benefit from unreliable or noisy labels (Bach et al., 2017; Ratner et al., 2017; Guo et al., 2018; Song et al., 2022). Semi-supervised learning (Kingma et al., 2014; Laine and Aila, 2016; Berthelot et al., 2019), when only a subset of labels are available, is closely related to our methodology. We fine-tune a base model on a random seed dataset, then iteratively train it on unlabeled data in a semi-supervised manner to progressively improve the initially weak model without full supervision.

3 Our Methodology: Self-Reinforcement

In this section, we describe our method to elicit the potential of language models for weak-to-strong generalization in reasoning tasks aimed at minimizing human annotation effort.

Our methodology assumes access to a base language model, a small amount of seed data, and a collection of unlabelled questions. The key assumption is that **Supervised Fine-Tuning (SFT) models will perform better in some questions than its unfinetuned base model within the same training domain**.

Our overall pipeline would entail three core steps:

- *base modeling:* Access unfinetuned base pretrained model π_0. Sample a seed data set $\mathcal{A}^{(0)} = \{(x_g, r_g, y_g)\}$ from the training set in PuzzleBen to optimize an SFT model π_1 by maximizing $p(r_g, y_g \mid x_g)$, where x_g is the sampled question labeled with rationale r_g and answer y_{g}.

- *self-filtering:* Randomly sample a set of unlabeled questions $\{x_{\text{u}}\}$ to generate rationales $r_0 \sim \pi_0(y \mid x_{\text{u}})$ and $r_1 \sim \pi_1(y \mid x_{\text{u}})$. We then design a self-filtering prompt to select responses where r_1 is preferred over r_0 using criteria like relevance and coherence, enhancing the unlabeled dataset with pairs of annotations $\mathcal{A}^{(1)} = \{(x_u, r_1, y_1, r_0, y_0) \mid r_1 \succ r_0\}$.
- *reinforcement learning:* Then, we apply Differential Performance Optimization (DPO) to learn from the discrepancies between pairs of rationales, further fine-tuning π_1 on $\mathcal{A}^{(1)}$ to get π_2.

We will describe the procedures of our methodology in more detail below.

3.1 Step 1: Base Modeling

This initial step involves enhancing the reasoning ability of the unsupervised base model π_0 by fine-tuning it with a small, high-quality annotated seed data $\mathcal{A}^{(0)} = \{(x_{\text{g}}, r_{\text{g}}, y_{\text{g}})\}$, where x_{g} is a sampled question labeled with rationale r_{g} and answer y_{g}. This process is aimed at directly improving the model's basic reasoning ability with the supervised fine-tuning loss function:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\mathbb{E}_{\{(x_{\text{g}}, r_{\text{g}}, y_{\text{g}})\} \sim \mathcal{A}^{(0)}} \left[\sum_{t=1}^{|r_{\text{g}}|} \log(\pi_\theta(a_t|s_t)) \right] \tag{1}$$

where θ represents the model parameters, and $\pi_\theta(a_t|s_t)$ is the probability of taking action a_t at state s_t, given the policy parameterized by θ. After supervised fine-tuning, we could get $\pi_1 = \pi_{\text{SFT}}$.

3.2 Step 2: Self-Filtering

To select high-quality examples for the next step, we further prompt π_1 itself to evaluate the response pairs to unlabeled questions generated by itself and π_0. Then we get $r_0 \sim \pi_0(y \mid x_{\text{u}})$ and $r_1 \sim \pi_1(y \mid x_{\text{u}})$. We attach self-filtering prompting we designed in Table 9. We aim to identify instances where π_1 outperforms π_0 based on relevance, coherence, and the presence of detailed rationales. Only responses where π_1 demonstrates superior reasoning are retained.

$$\mathcal{A}^{(1)} = \{(x_u, r_1, y_1, r_0, y_0) \mid r_1 \succ r_0\} \tag{2}$$

This selective approach ensures the inclusion of only high-quality rationale pairs in the training process, thereby improving the overall effectiveness of our methods.

3.3 Step 3: Reinforcement Learning

The third step in our methodology employs an innovative RL approach to utilize the augmented response pairs. This step is based on the assumption that SFT models will exhibit superior rationale-generating capabilities compared to their unfine-tuned counterparts within the same training domain. This difference in capability is primarily manifested in the quality of rationales produced.

The score s_i for the output (r_i, y_i) from π_i and its reference base model π_{ref} is derived as in Equation 3.

$$s_i = \beta \log \frac{P_{\pi_i}(r_i, y_i|x_i)}{P_{\pi_{ref}}(r_i, y_i|x_i)} \tag{3}$$

According to our assumptions, more capable models will obtain higher scores in this phase. This output quality discrepancy can be directly learnt with DPO based on the ranking loss in Equation 4. This enables us to finetune the stronger SFT model π_1 in a way that systematically amplifies its strengths in rationale generation.

$$L = \sum_{i,j:s_i > s_j} \max(0, s_i - s_j) \tag{4}$$

3.4 Iterative Self-Reinforcement

Self-reinforcement provides a reasonable approach to continue to refine its own reasoning ability interactively. By repeating this process, we enhance the model's understanding and reasoning capabilities to learn from the comparisons between itself and weaker models.

In the iterative process, we leverage the improved model from the previous iteration, π_1, and compare its output against the base model, π_0, to obtain a new model π_2. This is formalized as follows:

$$\pi_t = \text{Self-Reinforcement}\left(\pi_{t-1}, \pi_{t-2}\right) \tag{5}$$

Notably, our experiments in Section 6 demonstrate that our approach can continually grow with the improvements in the SFT model's capabilities. With each iteration of training, the previously "strong" model can serve as the "weaker" model for the next cycle, since the new, stronger model is developed based on the comparison between the two models from the prior round.

$$L_{\text{iter}} = \sum_{i,j:i \neq j; s_i^{t-1} > s_j^{t-2}} \max(0, s_j^{t-1} - s_i^{t-2}) \tag{6}$$

Here, L_{iter} represents the iterative self-reinforcement learning loss, s_i^{t-1} and s_j^{t-2} represent the scores of the rationales produced by π_{t-1} and π_{t-2} respectively. This iterative process allows the model to improve upon itself, leveraging the comparative strengths of each iteration's outcome.

4 Data Collection for PuzzleBen

In this section, we introduce PuzzleBen, a diversified and challenging benchmark with 25,147 annotated questions and 10,000 unannotated queries designed to test and enhance the LLMs' reasoning abilities with weak supervision. Our dataset spans multiple domains and question styles, and to illustrate this diversity, we create an overview of questions from PuzzleBen in Table 1 and include the detailed questions, answers, and human-annotated rationales in Table 8.

Each question in the training set comes with a gold-standard rationale crafted by human experts. All the answers and references are well-examined by the websites' users. The unlabeled set serves as a special part of PuzzleBen that is pivotal for exploring unsupervised or weakly-supervised learning techniques in the future. As for the test set, it has been thoughtfully structured to include options and answers, streamlining the evaluation process for enhanced convenience.

Meanwhile, a distinct section of our PuzzleBen dataset has been enriched with both difficulty and fun scores, informed by user interactions online. This feature emerges as a crucial resource for examining the reasoning capabilities of LLMs and their alignment with human supervisory thought processes.

Parajumbles The four sentences below when properly sequenced, would yield a coherent paragraph. Decide on the proper sequence.
Puzzles A B C D E F G H I J K L M N O P Q R S T U V W X Y Z If the second half of the alphabet is reversed then which letter will be 4th to the right of 20th letter from the right?
Brain Teasers Pointing to a person, a man said to a woman, "His mother is the only daughter of your father." How was the woman related to the person?
Riddles I will disappear every time you say my name. What am I?
Critical Reasoning Passage: ... Question 1: In this context, which of the following most logically explains the paradox? Question 2: Which of the following is an assumption on which the argument depends?

Table 1: Question examples from PuzzleBen. The detailed Q&A and human-annotated rationales are attached to Table 8 in Appendix.

4.1 Brainteasers

The primary intent of collecting brainteasers in PuzzleBen is to promote LLMs' capabilities in tackling problems that require deep thought and creative solutions. We systematically collect those questions from a well-designed open-sourced website, Braingle[1]. Each question is accompanied by a solution that has garnered widespread acceptance among users, along with a difficulty rating and a human rationale reference.

A subset of our dataset is distinguished by an additional metric from the website – the success rate of individuals who have attempted. The inclusion of human-assigned difficulty levels and success rates in this subset offers invaluable insights for our subsequent exploration into enhancing the weak-to-strong learning capabilities of LLMs.

4.2 Riddles

The primary intent of collecting riddles in PuzzleBen is to compel LLMs to think beyond the immediate context. A riddle can describe commonsense knowledge in explicit or counterlogical methods (Lin et al., 2021). We collect those well-designed riddles from an open-sourced website famous for stimulating cognitive explosions, ahaPuzzles[2].

While Lin et al. (2021) initiated the conversation, our dataset goes a step further by incorporating human rationale, vividly showcasing the intricacies of human thought processes. This addition significantly enhances the potential for LLMs to evolve innovatively and critically weak-to-strong generalizations from human's step-by-step reasoning iterations.

4.3 Puzzles

Puzzles are designed to challenge our cognitive faculties, forcing us to tap into both learned knowledge and innate logic in real-world problems. Unlike

[1] https://www.braingle.com/

[2] https://www.ahapuzzles.com/

riddles, which play on linguistic ambiguities or reconstructing logically coherent narratives, Puzzles hinge on methodical, step-by-step deduction and inference of structured problems.

We collect puzzles from sawaal[3], a well-known public website. This aspect is meticulously reviewed and validated by the community, ensuring the dataset serves as a rigorous training ground to promote LLMs from weak and basic capabilities to generalize strong reasoning capabilities.

4.4 Parajumbles

Parajumbles involve reordering jumbled sentences into a logical sequence, requiring a deep understanding of the relationships within texts. Including parajumbles in our dataset helps transition LLMs from basic learning to advanced modeling, enabling sophisticated logical reasoning.

The inspiration for this task is drawn from two well-known tests - Common Admission Test(CAT)[4] and Pearson Test of English for Academic(PTE)[5]. Besides CAT and PTE, we also collect and shuffle those paragraphs from (Misra, 2022; Harinatha et al., 2021), two open-sourced news datasets collected from various corpora, such as HuffPost, Business Insider, and CNN.

4.5 Critical Reasoning

Critical Reasoning (CR) is essential for evaluating advanced human cognition (Tittle, 2011). Inspired by the reasoning questions from GRE[6] and GMAT[7], our CR dataset tests and enhances LLMs' abilities to handle complex logical tasks such as understanding paradoxes, assumptions, and conclusions. This helps LLMs reflect the complex nature of human logic.

While our CR question format is similar to ReClor (Yu et al., 2020), our dataset includes expert rationale from experienced educators and excludes any identical questions found in ReClor, enhancing our benchmark's distinctiveness and educational value.

Table 2 presents each subset's size in our PUZZLEBEN, and we put more statistics results in Appendix A.

[3] https://www.sawaal.com/
[4] https://cdn.digialm.com/EForms/configuredHtml/756/84433/Registration.html
[5] https://www.pearsonpte.com/
[6] https://www.ets.org/gre.html
[7] https://www.mba.com/exams/gmat-exam/

Subset	Size
Annotated Trainset	22,528
Unannotated Question Set	10,000
Testset	2,618

Table 2: Detailed Subset's Size in PUZZLEBEN.

5 Baseline Performance on PUZZLEBEN

In this section, we evaluate several baseline models' performance on PUZZLEBEN.

5.1 Performance on Five Subtasks

Table 3 shows standard prompting and zero-shot CoT's performance of GPT4 and PaLM2 on five categories of tasks in PUZZLEBEN.

As we can see, CoT struggles with the parajumble task. The reason might be that parajumble largely tests concurrent reasoning, where one hypothesizes a sequence and then thinks in reverse to verify its correctness. CoT's step-by-step thinking approach can easily introduce errors at the very beginning of the logic. This limitation underpins the necessity for the PUZZLEBEN dataset, which aims to enrich future research's landscape by focusing on diverse tasks that challenge current models in various novel ways.

5.2 Utility of Human Rationale Collected in PUZZLEBEN

To convince the utility of the human rationales in PUZZLEBEN, we conduct experiments to utilize those collected rationales both in prompting and fine-tuning directions. Table 4 represents the relations between In-Context Learning (ICL) accuracy and k-shot rationale examples.

As the number of shots of the training examples increases, the performance across most tasks seems to improve. Specifically, for the Puzzles and Riddles tasks, there's a noticeable increase in performance from the 0-shot to the 8-shot learning. The Parajumble and Brainteasers task, though starting with a lower performance score, also shows a similar positive trend.

The evaluation showcases the utility of human reference in PUZZLEBEN. It is evident that increasing the number of shots or examples benefits the model's accuracy, especially in tasks like Puzzles, Riddles, Parajumble and Brainteasers. This analysis suggests that for tasks demanding a deeper understanding of complex reasoning, a higher number of shots might provide better guidance to the

Model	Method	Puzzles	Riddles	Parajumble	CR	Brainteasers
PaLM2	Standard Prompting (Brown et al., 2020)	49.45	61.90	25.54	58.39	34.89
	Zero-Shot CoT (Madaan et al., 2023)	53.24	63.03	20.08	51.98	41.96
GPT4	Standard Prompting (Brown et al., 2020)	64.37	67.70	52.17	65.32	52.58
	Zero-Shot CoT (Madaan et al., 2023)	81.22	81.92	45.96	63.01	53.53

Table 3: PaLM2 and GPT4's accuracy on the five tasks in PUZZLEBEN. CR stands for critical reasoning subset.

Shots	Puzzles	Riddles	Parajumble	CR	BT
0	81.22	81.92	45.96	63.01	53.53
1	82.92	80.53	46.27	65.97	53.02
8	84.90	85.63	51.42	68.73	55.62

Table 4: GPT4's k-shot ICL performance on PUZZLEBEN. BT stands for Brainteaser tasks.

model, leading to improved outcomes.

To further demonstrate the effectiveness of our PUZZLEBEN dataset, we have conducted a detailed analysis of the effectivenss of collected human rationales in PUZZLEBEN for SFT. The results, as shown in Table 5, highlights the substantial improvements in LLaMA-13b's performance when finetuned with our dataset. These improvements underscore the quality and relevance of the training data provided in our PUZZLEBEN. All of those results indicate how well our dataset is suited for enhancing LLMs' complex reasoning capabilities.

Model	Method	Accuracy
LLaMA2-13b	-	10.38
	after SFT	41.22

Table 5: LLaMA-13b's performance on PUZZLEBEN's testset before and after Supervised Finetuning (SFT).

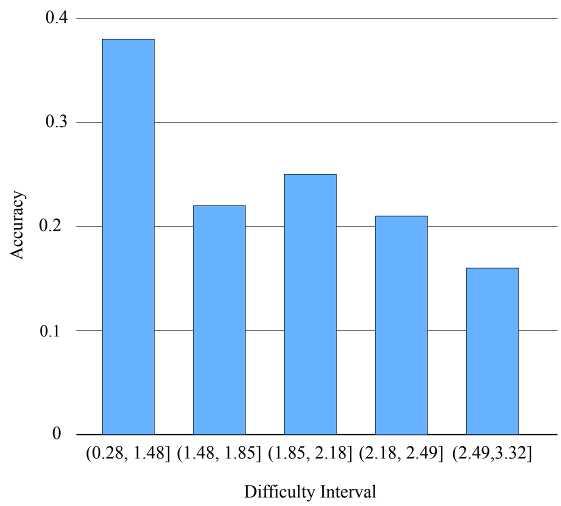


Figure 2: Accuracy of Llama2-13b across interval-based difficulty score ranges on the subset of PUZZLEBEN. The difficulty ratings represent the average of all user-assigned scores ranging from 1 to 4, with each category containing an equal number of items.

5.3 Correlation between Model Performance and Human Difficulty Perception

Our experiments Results depicted in Figure 2 illustrate a broad trend where Llama2-13b's accuracy on the PuzzleBen subset wanes as difficulty score intervals rise. This pattern shows that the model's challenges generally match the rising difficulty of tasks as humans perceive them, though not perfectly. Our research points to the possibility of improving model performance by tuning it to align more closely with human perceptions of task difficulty, rather than merely matching answers to questions. This approach could enhance the model's understanding of reasoning tasks.

6 Experiments about Self-Reinforcement

6.1 Initialization

Seed data & Unlabeled Questions We randomly select 6400 questions and its rationales from PUZZLEBEN. Considering the difficulty of our dataset, each question and answer has all been fully examined and discussed by annotators. We also randomly select 6400 unanswered questions for each iteration.

Training Details We choose the pretrained LLaMA2-13b (Touvron et al., 2023) as our base model. Throughout the training, we consistently apply standard hyperparameters: a learning rate of 5e-5, a batch size of 16 instances, and a total of 3 training epochs. Besides, we employ QLoRA (Dettmers et al., 2024) with a rank of 16, a LoRA alpha set to 32, and a LoRA dropout rate of 0.05.

Baselines As we discussed in Section 2, we introduced a novel method to improve LLM reasoning abilities with minimal human effort. Self-reinforcement's motivations and settings are different from traditional methods utilizing extensive prompting or heavy fine-tuning. Hence, we have few comparable baselines. However, a similar approach, ReFT (Luong et al., 2024), also uses minimal input and RL to enhance LLMs by learning from model-decoded rationales, specifically by sampling reasoning paths and then creating positive and negative pairs based on the final result. Although this method aligns with ours to some extent, it cannot be applied to unformatted human rationale texts or datasets lacking an exact answer.

6.2 Self-reinforcement Results on PuzzleBen

Methods	Iterations	Accuracy
Unfinetune	-	10.38
SFT	-	17.33
ReFT	-	22.47
self-reinforcement (ours)	t_1	28.11
self-reinforcement (ours)	t_2	37.82

Table 6: LLaMA2-13b self-reinforcement and the baselines' results on PuzzleBen with the same labeled seed data set.

Our experimental results on the PuzzleBen dataset using our self-reinforcement approach highlight significant enhancements in model performance. Our method surpassed traditional strategies such as Unfinetuned, SFT, and ReFT, reflecting the efficacy of our iterative, weak-to-strong learning framework. From the base accuracy of 10.38%, our model's accuracy improved drastically to 37.82% by the second iteration (t_2), underscoring the potential of self-reinforcement in leveraging weak supervision for substantial gains in reasoning tasks.

These findings support the effectiveness of our self-reinforcement methodology in continuously refining the reasoning capabilities of language models under limited supervision. By iterating through cycles of self-filtering and differential performance optimization, our approach not only enhances the quality of rationale generation but also steadily increases the overall model accuracy.

6.3 Ablation Study

Iterations	Methods	Accuracy
-	SFT	17.33
t_1	w/o self-filtering	18.32
	w self-filtering	28.11
t_2	w/o self-filtering	18.28
	w self-filtering	37.82

Table 7: Our method's accuracy with and without self-filtering in each iteration.

In this ablation study, we further explore self-filtering's potential impacts on our method. The results in Table 9 distinctly illustrates the crucial role of self-filtering in enhancing the performance of our self-reinforcement methodology. By comparing the results of models trained with and without the self-filtering component, it becomes evident that self-filtering significantly boosts accuracy across multiple iterations.

For instance, at iteration t_1, the model incorporating self-filtering achieved an accuracy of 28.11%, which is a substantial increase compared to the 18.32% accuracy of the model without self-filtering. Similarly, at iteration t_2, the gap widened even further, with the self-filtering model reaching an accuracy of 37.82% compared to 18.28% for the model without this feature. This clear disparity underscores the effectiveness of self-filtering in refining the dataset and improving the model's reasoning capabilities, thus leading to better performance on complex reasoning tasks.

7 Conclusions and Future Work

In this work, we introduce PuzzleBen, a benchmark tailored to augment and assess LLMs' understanding of creative, comprehensive, and nonlinear reasoning tasks. Each question is designed with high-quality and well-designed rationale reference annotated by human experts. In this direction, we propose self-reinforcement, in order to unveil LLMs' weak-to-strong self-learning capabilities in reasoning tasks under weak human supervision. Our methodology only requires a small annotated dataset compared with previous work. To utilize DPO for learning from the quality differences between the rationales decoded by stronger models and those from weaker base models, self-reinforcement provides a possible solution to exploit minimal human supervision effectively.

In future work, we plan to improve the self-reinforcement framework by incorporating dynamic and adaptive self-filtering criteria to enhance the quality of model-decoded data. Furthermore, employing active learning strategies or collaborative human-in-the-loop interventions may help align the models with complex human reasoning techniques and guide the development of LLMs from weak to strong reasoning capabilities. These improvements will aid in creating more autonomous, efficient, and robust reasoning models.

Limitations

It is crucial to recognize that the self-reinforcement process could see improvements with further refinements in self-filtering. Specifically, choosing more impactful positive and negative pairs can greatly enhance the effectiveness of DPO training. This approach aligns with the strategy of leveraging highly capable models or human experts for alignment tasks. Moreover, there remains uncertainty regarding the stability of our model with extensive iterations; specifically, whether the model might experience collapse or increased hallucination phenomena as iterations progress. Introducing a certain proportion of human-annotated data in each iteration could serve as an alignment mechanism, potentially mitigating these issues and ensuring the model remains robust and accurate over long-term training.

Another notable limitation is the inherent challenge of tuning parameters to prevent outputs from becoming progressively longer or shorter. This issue is reminiscent of similar behaviors observed in many reinforcement learning scenarios. To address this, setting appropriate generation-related parameters (such as early stopping and max new tokens) is essential. Additionally, incorporating penalty terms during the training process can help regulate output length and maintain the desired balance.

References

Stephen H Bach, Bryan He, Alexander Ratner, and Christopher Ré. 2017. Learning the structure of generative models without labeled data. In *International Conference on Machine Learning*, pages 273–282. PMLR.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. 2019. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michal Podstawski, Hubert Niewiadomski, Piotr Nyczyk, et al. 2023. Graph of thoughts: Solving elaborate problems with large language models. *arXiv preprint arXiv:2308.09687*.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, et al. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17682–17690.

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

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, et al. 2023. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. *arXiv preprint arXiv:2312.09390*.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research*.

Lichang Chen, Chen Zhu, Davit Soselia, Jiuhai Chen, Tianyi Zhou, Tom Goldstein, Heng Huang, Mohammad Shoeybi, and Bryan Catanzaro. 2024a. Odin: Disentangled reward mitigates hacking in rlhf. *arXiv preprint arXiv:2402.07319*.

Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. 2021. Semi-supervised semantic segmentation with cross pseudo supervision.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024b. Self-play fine-tuning converts weak language models to strong language models.

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 Sankaranarayana 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. 2022. Palm: Scaling language modeling with pathways.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Antonia Creswell and Murray Shanahan. 2022. Faithful reasoning using large language models. *arXiv preprint arXiv:2208.14271*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36.

Yunlong Feng, Yang Xu, Libo Qin, Yasheng Wang, and Wanxiang Che. 2024. Improving language model reasoning with self-motivated learning. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8840–8852.

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

Sheng Guo, Weilin Huang, Haozhi Zhang, Chenfan Zhuang, Dengke Dong, Matthew R Scott, and Dinglong Huang. 2018. Curriculumnet: Weakly supervised learning from large-scale web images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 135–150.

Sreeya Reddy Kotrakona Harinatha, Beauty Tatenda Tasara, and Nunung Nurul Qomariyah. 2021. Evaluating extractive summarization techniques on news articles. In *2021 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, pages 88–94. IEEE.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2022. Large language models can self-improve.

Yuu Jinnai, Tetsuro Morimura, Kaito Ariu, and Kenshi Abe. 2024. Regularized best-of-n sampling to mitigate reward hacking for language model alignment. *arXiv preprint arXiv:2404.01054*.

Seungone Kim, Se June Joo, Doyoung Kim, Joel Jang, Seonghyeon Ye, Jamin Shin, and Minjoon Seo. 2023. The cot collection: Improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning.

Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. 2014. Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27.

Samuli Laine and Timo Aila. 2016. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*.

Bin Lei, Chunhua Liao, Caiwen Ding, et al. 2023. Boosting logical reasoning in large language models through a new framework: The graph of thought. *arXiv preprint arXiv:2308.08614*.

Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. 2023. Making language models better reasoners with step-aware verifier. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5315–5333.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Bill Yuchen Lin, Ziyi Wu, Yichi Yang, Dong-Ho Lee, and Xiang Ren. 2021. Riddlesense: Reasoning about riddle questions featuring linguistic creativity and commonsense knowledge. *arXiv preprint arXiv:2101.00376*.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pages 22631–22648. PMLR.

Trung Quoc Luong, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. 2024. Reft: Reasoning with reinforced fine-tuning.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*.

Rishabh Misra. 2022. News category dataset. *arXiv preprint arXiv:2209.11429*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. 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 Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2023. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *arXiv preprint arXiv:2308.03188*.

Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. 2024. Iterative reasoning preference optimization. *arXiv preprint arXiv:2404.19733*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model.

Leonardo Ranaldi and Andre Freitas. 2024. Aligning large and small language models via chain-of-thought reasoning. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1812–1827, St. Julian's, Malta. Association for Computational Linguistics.

Alexander Ratner, Stephen H Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. 2017. Snorkel: Rapid training data creation with weak supervision. In *Proceedings of the VLDB endowment. International conference on very large data bases*, volume 11, page 269. NIH Public Access.

Aruni RoyChowdhury, Prithvijit Chakrabarty, Ashish Singh, SouYoung Jin, Huaizu Jiang, Liangliang Cao, and Erik Learned-Miller. 2019. Automatic adaptation of object detectors to new domains using self-training.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. 2022. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*.

Zhen Tan, Alimohammad Beigi, Song Wang, Ruocheng Guo, Amrita Bhattacharjee, Bohan Jiang, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. Large language models for data annotation: A survey. *arXiv preprint arXiv:2402.13446*.

Peg Tittle. 2011. *Critical thinking: An appeal to reason*. Routledge.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Peifeng Wang, Zhengyang Wang, Zheng Li, Yifan Gao, Bing Yin, and Xiang Ren. 2023. Scott: Self-consistent chain-of-thought distillation.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. 2020. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. *arXiv preprint arXiv:2002.04326*.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022a. Opt: Open pre-trained transformer language models.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022b. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

Zirui Zhao, Wee Sun Lee, and David Hsu. 2024. Large language models as commonsense knowledge for large-scale task planning. *Advances in Neural Information Processing Systems*, 36.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, et al. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

Zhi-Hua Zhou. 2018. A brief introduction to weakly supervised learning. *National science review*, 5(1):44–53.

A More Statistics about PUZZLEBEN

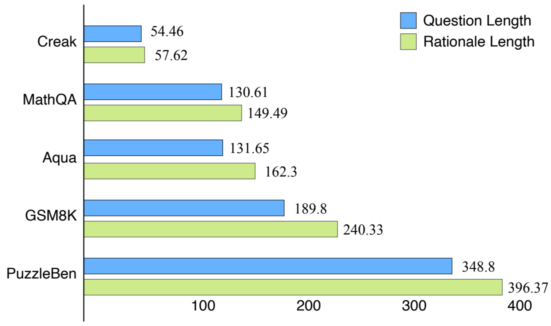


Figure 3: Average Length of Questions and Rationales designed in PUZZLEBEN and the other existing benchmarks designed with human rationales.

In this section, we provide several statistical analyses of our benchmark. As we can see in Figure 3, PUZZLEBEN distinguishes itself significantly in terms of the average length of questions and rationales when compared to other existing benchmarks. With questions averaging 348.80 characters and rationales at 396.37 characters, PuzzleBen's content not only exhibits a higher degree of complexity but also provides more elaborate explanations, which further proves PUZZLEBEN's uniqueness and necessity to the community.

A distinctive aspect of our PuzzleBen subset lies in its incorporation of difficulty scores for each brainteaser, derived from the pass rates of online users, offering a directional reflection of our collective grasp on reasoning tasks. The outcomes of our experiments, as detailed in Section 5.3, substantiate the effectiveness and necessity of this feature. This subset promises substantial relevance for future reasoning work, ensuring alignment with human cognitive perceptions from a novel direction.

B Detailed Examples in PUZZLEBEN

Part 1: Brainteasers

- *Question:* What characteristic do these three 12-digit numbers share with each other, but with no other 12-digit number? 100307124369, 111824028801, 433800063225.
- *Rationale:*
  - They are all square numbers: $100307124369 = 316713^2, 111824028801 = 334401^2, 433800063225 = 656635^2$
  - The sum of their digits are square numbers: $1+0+0+3+0+7+1+2+4+3+6+9 = 36 = 6^2, 1+1+1+1+8+2+4+0+2+8+8+0+1 = 36 = 6^2, 4+3+3+8+0+0+0+6+3+2+2+5 = 36 = 6^2$.
  - The sum of their digit pairs are square numbers: $10+03+07+12+43+69 = 144 = 12^2, 11+18+24+02+88+01 = 144 = 12^2, 43+38+00+06+32+25 = 144 = 12^2$.
  - The sum of their digit triplets are square numbers: $100+307+124+369 = 900 = 30^2, 111+824+028+801 = 1764 = 42^2, 433+800+063+225 = 1521 = 39^2$.
  - The sum of their digit quadruplets are square numbers: $1003+0712+4369 = 6084 = 78^2, 1118+2402+8801 = 12321 = 111^2, 4338+0006+3225 = 7569 = 87^2$.
  - The sum of their digit sextuplets are square numbers: $100307+124369 = 224676 = 474^2, 111824+028801 = 140625 = 375^2, 433800+063225 = 497025 = 705^2$.
- *Difficulty: 3.23, Fun: 2.45*

Part 2: Riddles

- *Question:* What has 13 hearts, but no other organs?
- *Rationale:* A deck of playing cards consists of 52 cards, divided into four suits: hearts, diamonds, clubs, and spades. Each suit contains one card for each rank from two to ten, plus a jack, queen, king, and ace. This means there are exactly 13 cards in the hearts suit, each metaphorically referred to as having a heart. However, these cards, being inanimate objects, do not possess any other organs, unlike living beings which have a heart along with other organs. This riddle plays on the word hearts as a suit in playing cards and the literal organ, making a deck of playing cards the correct answer since it metaphorically has 13 hearts but lacks any other organs.

Part 3: Puzzles

- *Question:* A, B, C, D and E are sitting in a row. B is between A and K Who among them is in the middle ? I. A is left of 13 and right of D. II.C is at the right end. [Options] A. If the data in statement I alone are sufficient to answer the question B. If the data in statement II alone are sufficient answer the question C. If the data either in I or II alone are sufficient to answer the question; D. If the data in both the statements together are needed.
- *Rationale:* Clearly, we have the order : A. a E. From I, we have the order : D, A, B. E. From II, we get the complete sequence as D, A, B. E, C. Clearly. B is in the middle. So, both I and II are required.

Part 4: Critical Reasoning

- *Question:* In the shallow end of Lake Tomwa, there are remains of numerous Jeffery pine trees that grew there during a lengthy drought. Researchers had believed that this drought lasted at least 150 years, but carbon dating reveals that pines were growing in the lake bed for only 120 years, from 1200 until 1320. Since the Jeffrey pines, which cannot survive in water, must have died at the end of the drought, the dating shows that the drought lasted less than 150 years. **The argument given relies on which of the following as an assumption?** [Options] A. No other species of tree started growing in the bed of Lake Tomwa after 1200. B. No tree remains of any kind are present at the bottom of deeper parts of Lake Tomwa. C. There was at least one tree in the lake bed that was alive for the entire period from 1200 to 1320. D. There has not been a more recent drought that caused a drying up of the shallow end of the lake. E. The shallow end of the lake had been dry for less than 30 years by the time Jeffrey pines started growing in the lake bed.
- *Rationale:* The reasoning process in this article can be summarized as follows: (1) Pine trees cannot survive in water (they can only survive during dry periods) $\rightarrow$ after the dry period ends, J pine trees will inevitably die; (2) J pine trees only lived for 120 years: (1)+(2) $\rightarrow$ the duration of the drought was less than 150 years. The problem with this reasoning process is that it cannot determine when the drought began, as the drought could have started well before the J pine trees began to grow. Option A is incorrect because whether other species of trees began to grow 1200 years later does not affect the inference in the text, as the dating method mentioned is specific to J pine trees and is not influenced by other species of trees. Even if other water-resistant species of trees survived, it is irrelevant to the discussion at hand. Option B is incorrect, as whether trees existed at the deeper bottom of the lake does not affect the inference in the text. The depth of the lakebed where trees grew at most could only indicate the extent of the drought, not the existence of the drought itself. Option C is incorrect because whether any trees lived through the entire 120 years does not affect the inference in the text, as the dating method mentioned has already proven that J pine trees grew from 1200 to 1320. Even if each tree lived only one year, it does not affect the deduction that "J pine trees survived between 1200 and 1320." Option D is incorrect because whether a drought occurred again later does not affect the inference in the text, as whether there was a drought later is irrelevant to the study of this period. Additionally, the dating method has already proven that pine trees only survived during the consecutive 120 years between 1200 and 1320, which indicates that the specific drought period mentioned ended in 1320. Option E is correct because the text does not provide evidence on when the drought began. If the drought had already lasted for more than 30 years by the time J pine trees began to grow, then adding the 120 years of J pine trees' growth period, the total duration of the drought would exceed 150 years, contradicting the conclusion in the text.

Part 5: Parajumble

- *Question:* Reorder the following sentences to form a coherent paragraph. Sentence A) For example, if I am a group member, I can choose group -sending. Sentence B) About what an email list is. Sentence C) What the use of email list is. You can arrange contacts into a particular group in the email list. Sentence D) Further explanation for the example. No new words, and very easy.
- *Rationale:* To solve this, we shall analyze the given sentences closely to understand their logical and thematic connections. Sentence B serves as a general introduction by talking about what an email list is. It sets the stage for further discussion on the specifics of an email list, making it the natural starting point. Following the introduction of the email list, Sentence C delves into What the use of email list is by explaining that You can arrange contacts into a particular group in the email list. This explanation directly builds upon the introductory concept provided in sentence B, expanding the readerś understanding of the functionality and purpose of an email list. Sentence A presents a specific example For example, if I am a group member, I can choose group-sending. This sentence illustrates a practical application of the concept introduced in sentences B and C, showing how an individual might utilize the email listś functionality. Finally, Sentence D offers Further explanation for the example. No new words, and very easy. Since it aims to elaborate on the example given in sentence A, it logically follows that sentence, rounding off the explanation and providing clarity. Thus, the coherent sequence is B (introduction to the topic), followed by C (explanation of usage), leading into A (specific example of usage), and concluded with D (further elucidation of the example). Therefore, the correct order is BCAD, creating a logical flow from a general introduction to a specific example and its explanation.

Table 8: Examples collected from our PUZZLEBEN.

C Self-Filtering's Prompt Design

- Question: {}
- Response1: {}
- Response2: {}
- A good Response is:
  - 1. relevant to the Question
  - 2. seemingly correct and coherent
  - 3. do not output repeated or nonsense words.
  - 4. provide some rationales, explanations or answer
- Do you think Response1 is better than Response2? Only answer "yes" or "no":

Table 9: Prompting we designed in the stage of self-filtering. Response1 is generated from M_1 while Response2 is from M_0. We filter out the samples which Response1 is obviously worse than Response0.