Towards Faithful Multi-step Reasoning through Fine-Grained Causal-aware Attribution Reasoning Distillation

Zheng Chu[1], **Jingchang Chen**[1], **Zhongjie Wang**[1], **Guo Tang**[1]
Qianglong Chen[2], **Ming Liu**[1,3*], **Bing Qin**[1,3]
[1]Harbin Institute of Technology, Harbin, China
[2]Zhejiang University [3]Peng Cheng Laboratory
zchu@ir.hit.edu.cn

Abstract

Despite the remarkable reasoning capabilities demonstrated by large language models (LLM), the substantial computational overhead limits their practices. Some efforts have been directed toward distilling multi-step reasoning capabilities into smaller models through chain-of-thought (CoT). While CoT facilitates multi-step reasoning, the dependencies between reasoning steps are not always clearly discernible, which may lead to inconsistent reasoning. In this paper, we introduce fine-grained attribution reasoning distillation (FARD), which incorporates grounded citations to consolidate the relationships between reasoning steps. Specifically, FARD distills attribution reasoning rationales from LLMs to substitute CoT reasonings, which clarifies the dependencies among reasoning steps. Besides, we regularize the model's attention pattern by leveraging the causal dependencies between reasoning steps, thereby enhancing the consistency of reasoning. Grounded attribution reasoning also enhances interpretability and verifiability, thereby facilitating faithful reasoning. We evaluate FARD on mathematical and general reasoning benchmarks. The experimental results indicate that FARD outperforms CoT distillation methods in mathematical reasoning, demonstrating its effectiveness. Furthermore, the small models trained with FARD have shown outstanding performance in out-of-distribution reasoning, proving strong generalization capabilities.[1]

1 Introduction

Recent advancements have witnessed large language models (LLMs) achieving significant milestones across various domains of natural language processing. A noteworthy point is the application of chain-of-thought prompting (Wei et al., 2022a), which facilitates LLMs to conduct step-by-step reasoning before providing definitive responses, consequently improving their accuracy and reliability. Chain-of-thought has demonstrated remarkable performance in complex reasoning, such as mathematical reasoning and logical reasoning (Wei et al., 2022b; Cobbe et al., 2021; Tafjord et al., 2021).

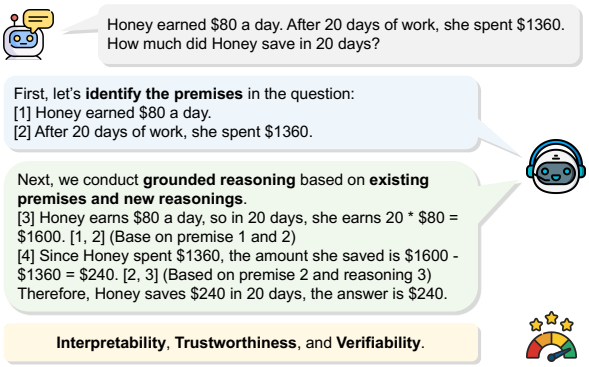

Figure 1: An example of attribution reasoning in mathematical tasks. Grounded attribution reasoning enhances interpretability, trustworthiness, and verifiability.

Despite the remarkable achievement, the vast number of parameters and high computational costs of LLMs limit their deployment, thereby constraining their availability. In response, some works attempt to transfer the capabilities of LLMs into smaller ones. One viable approach is distilling CoT reasoning trajectories from LLMs to fine-tune smaller models, thereby endowing them with the step-by-step reasoning capabilities (Fu et al., 2023; Li et al., 2023d; Wang et al., 2023a).

Although these methods can enhance the reasoning capabilities of smaller models, certain challenges still persist. Firstly, chain-of-thought reasoning fails to explicitly describe the fine-grained dependencies between reasoning steps, potentially leading to inconsistent and unfaithful reasoning. Besides, the linear structure of chain-of-thought reasoning poses difficulties for post-hoc verification and refinement of the reasoning trajectories. Secondly, these methods tend to induce overfitting to the teacher's outputs among students, con-

*Corresponding Author
[1]Resources link

Proceedings of the 31st International Conference on Computational Linguistics, pages 2291–2315
January 19–24, 2025. ©2025 Association for Computational Linguistics

sequently undermining their capacity for generalization (Gudibande et al., 2023; Chen et al., 2023).

To address these challenges, this paper aims to introduce attribution reasoning to facilitate faithful reasoning and enhance generalization capabilities. Accordingly, we propose Fine-grained Attribution Reasoning Distillation (FARD). Firstly, the grounded citations within attribution reasoning process offer enhanced interpretability and post-hoc verifiability, as shown in Figure 1. Secondly, we employ fine-grained causal dependencies between reasoning steps to regularize the attention pattern during reasoning, thereby enhancing consistency. Finally, we abstract the reasoning process into three aspects: premise extraction, attribution reasoning with self-questioning, and answer summarization, thereby enhancing generalization. FARD equips small models with strong multi-step reasoning capabilities, and the attribution reasoning trajectories enhance interpretability and verifiability, thereby facilitating faithful multi-step reasoning.

Concretely, we begin by distilling attribution reasoning rationales enriched with grounded citations from a strong teacher model. Following this, we employ a validity filter and diversity filter to screen high-quality and diverse rationales. These rationales are then parsed into fine-grained causal dependency graph based on attribution citations. Finally, we fine-tune the student model using the derived rationales, and employ the causal dependency to regularize the attention pattern.

We conduct extensive experiments on 3 models across 9 datasets including diverse reasoning aspects. Experimental results demonstrate that FARD outperforms CoT distillation in performance. For instance, FARD achieved an 3.0% improvement over the baselines on GSM8K (Cobbe et al., 2021), and it still maintains excellent performance when faced with perturbed questions (Li et al., 2024). Additionally, our experiments on 4 out-of-distribution mathematical datasets: SVAMP (Patel et al., 2021), ASDiv (Miao et al., 2020), MultiArith (Roy and Roth, 2015), and SingleEQ (Koncel-Kedziorski et al., 2015) also surpass the baselines, indicating that our attribution distillation method possesses stronger generalization capabilities. Moreover, to investigate the general reasoning capabilities, we conduct experiments on commonsense reasoning StrategyQA (Geva et al., 2021), temporal reasoning Date Understanding, and logical reasoning Logical Deduction (Srivastava et al., 2023). Experimental results indicate that FARD performs well in out-of-domain scenarios, demonstrating its cross-domain generalization capabilities.

Our contributions can be summarized as follows:

- We propose FARD for fine-grained attribution reasoning distillation to enhance the multi-step reasoning capabilities of small models.
- FARD uses fine-grained causal dependencies between reasoning steps to regularize the model's pattern, enhancing the reasoning consistency.
- Attribution reasoning trajectories provide improved interpretability and verifiability, thereby facilitating faithful multi-step reasoning.

2 Related Work

2.1 Chain-of-Thought Reasoning

Recent work has shown that prompting LLMs to provide step-by-step reasoning trajectories before final answers can significantly improve their reasoning capabilities and provide interpretability, which is known as chain-of-thought prompting (Wei et al., 2022a; Chu et al., 2024). Some studies design instructions to guild LLMs in zero-shot CoT to reduce annotation cost (Wei et al., 2022b; Zhang et al., 2023b), and ensemble learning is also proved an effective approach to enhance reasoning performance (Wang et al., 2023b; Aggarwal et al., 2023). Additionally, Dua et al. (2022); Zhou et al. (2023) breaks down complex questions into simpler subquestions and tackles them sequentially, thereby reducing the complexity of reasoning.

In this paper, we divide reasoning process into three components: key information extraction, self-questioning, and attribution reasoning, to enhance both performance and domain generalization.

2.2 Multi-step Reasoning Distillation

Knowledge distillation serves as an effective method to transfer the capabilities of large models to smaller ones (Hinton et al., 2015). However, the inaccessibility of advanced models, coupled with the high training costs, renders white-box distillation methods difficult to apply, such as response-based (Kim et al., 2021; Huang et al., 2022), feature-based (Zagoruyko and Komodakis, 2017; Sun et al., 2019) and relation-based techniques (Park et al., 2019; Zhang et al., 2024).

Recent work treats LLMs as black boxes (Zhu et al., 2023), collecting their step-by-step reasoning trajectories as training corpus. Fu et al. (2023); Wang et al. (2023a); Li et al. (2023d) extract chain-of-thought reasoning data from LLMs as training

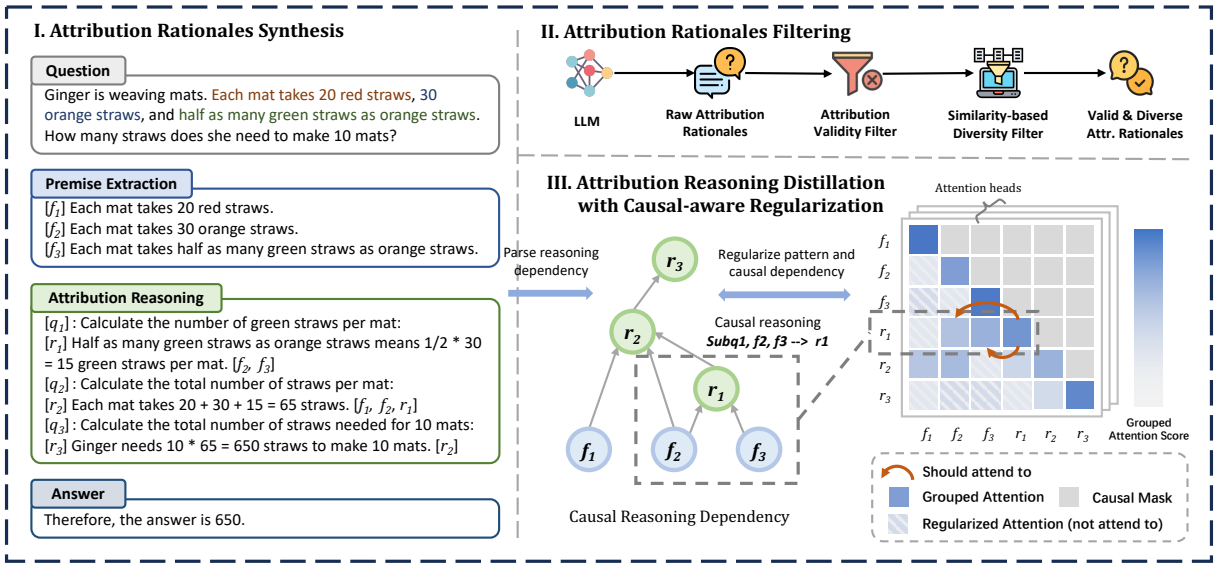

Figure 2: Overview of FARD, which includes: I. Attribution Rationales Synthesis: sample multiple attribution reasoning trajectories from LLMs. II. Attribution Rationales Filtering: obtain high-quality and diverse rationales through filtering. III. Causal-aware Attribution Reasoning Distillation: Fine-tune the small model using attribution rationales with causal-aware dependency regularization. For brevity, only premise and reasoning nodes are shown.

data, and Chen et al. (2023) introduces consistency distillation by optimizing KL divergence between reasoning pairs. Besides, Wang et al. (2023c) gets customized reasoning trajectories from LLMs based on feedback from students. Additionally, some research has explored extracting code format rationales from LLMs to enable code-assisted reasoning (Li et al., 2023a; Zhu et al., 2024a,b).

Compared to their approach, we obtain structured attribution reasoning rationales from the LLMs and attempt to further optimize the model using the structural reasoning dependency graphs.

2.3 Attributed Citation Generation

Hallucinations pose a significant challenge in the generation and reasoning in large language models (Zhang et al., 2023a; Huang et al., 2023). Recently, some studies have incorporated attribution citations of evidence into generation tasks to enhance verifiability and thereby mitigate hallucination issues (Li et al., 2023b; Bohnet et al., 2022; Li et al., 2023c; Yue et al., 2023; Berchansky et al., 2024). Gao et al. (2023b); Fierro et al. (2024) enables LLMs to generate citations concurrently with the text through in-context learning, whereas Gao et al. (2023a); Xu et al. (2023) explores post-hoc attribution, which involves LLMs seeking the most relevant information to support attribution after generating preliminary answers, and Huang et al. (2024) delves into finer-grained attribution citations at the sentence level.

In contrast to the aforementioned works that concentrate on generation tasks, our method incorporates attribution within multi-step reasoning tasks, obtaining detailed step-level causal dependencies relationships and offering better interpretability.

3 Methodology

We introduce a fine-grained attribution reasoning distillation (FARD) framework to enhance the robust and generalized reasoning capabilities of smaller language models, while also improving the interpretability and verifiability of the reasoning process. Specifically, we first distill structured attribution reasoning rationales from LLMs. Subsequently, we apply filters based on content validity and similarity to obtain high-quality and diverse reasoning chains, which are then parsed into reasoning causal dependency graphs. Finally, we employ fine-grained attribution reasoning rationales in conjunction with causal dependency graphs to enhance the reasoning capabilities of smaller models. The overview of FARD is illustrated in Figure 2.

3.1 Attribution Rationales Synthesis

FARD distills rationales in the form of attribution reasoning to enhance the interpretability and verifiability of the reasoning process. Concretely, we divide the reasoning process into three components: premise extraction, attribution reasoning, and an-

swer summarization, as illustrated in Figure 2(I).

Premise extraction entails extracting crucial information from the question and formatting it as numbered premises for reference in subsequent reasoning. In attribution reasoning, we introduce self-questioning to decompose complex questions into a series of simpler sub-questions. The solution to each sub-question relies on existing premises or reasoning, thereby enabling grounded attribution reasoning. Finally, the model derives a final answer based on the preceding reasoning.

We obtain attribution reasoning rationales from the teacher LLM through in-context learning. To ensure diversity in the rationales, we set the temperature τ=*1.3* to sample *N=32* instances for each question in our experiments. For demonstrations of in-context learning, please refer to Appendix C.

3.2 Rationales Filtering and Parsing

3.2.1 Attribution Reasoning Filtering

Even the strong teacher LLMs may produce erroneous reasoning and attributions, thereby filtering is necessary to maintain data quality. We compare the prediction with the ground truth to verify the correctness of reasoning. Subsequently, we validate the attribution structures and filter out samples that are structurally invalid. Besides, recent work has indicated that the diversity of training instances influences model performance (Li et al., 2023d; Chen et al., 2023). To enhance reasoning diversity within the limited training instances, we adopt an iterative similarity-based filtering approach to retain high-diverse examples (Chen et al., 2023). We describe detailed filtering algorithm in Appendix A.2.

3.2.2 Causal Dependency Parsing

The structured characteristic of attribution reasoning allows us to readily acquire fine-grained, step-level causal dependencies within the reasoning. Specifically, we transform the reasoning into a directed acyclic graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ based on the reasoning tags and their corresponding citation tags.

$$\mathcal{V} = \{v_p^i, \cdots, v_q^j, \cdots, v_r^k, \cdots\} \quad (1)$$

$$\mathcal{E} = \{(v^i, v^j) \mid v^i \in \mathcal{V}, v^j \in \mathcal{V}_r\} \quad (2)$$

where v_p, v_q, v_r denote the premise, sub-question, and reasoning nodes, respectively. $\mathcal{V}$ represents any node, and $\mathcal{V}_r$ denotes the reasoning nodes.

Figure 2 illustrates the reasoning graph. In the graph, each sub-graph of depth 2 represents the reasoning process of a sub-question. The sub-question reasoning is a deductive reasoning process involving causal relationships between premise nodes $v^i \in \mathcal{V}$ and conclusion nodes $v^j \in \mathcal{V}_r$.

3.3 Attribution Reasoning Distillation with Causal-aware Regularization

As previously mentioned, we have obtained the causal dependencies between reasoning steps, and we aim to utilize this prior information for optimization. Recent work on the on the mechanisms of multi-step reasoning has found that the attention pattern of LLMs aligns with the reasoning dependency tree (Hou et al., 2023). To this end, we seek to strengthen this pattern by employing the causal dependency relationships to improve reasoning.

Next, we will describe how to use the causal dependency through a specific example. The dashed box in Figure 2(III) depicts a reasoning process of a sub-question. Within the sub-question, reasoning r_1 depends on premises f_2 and f_3, forming a causal relationship $f_2 \wedge f_3 \rightarrow r_1$. Given this causal relationship, r_1 should attend more to content of the premises f_2 and f_3 it relies on, rather than to f_1, which is irrelevant to current reasoning step. Therefore, we should apply regularization to positions that are irrelevant to the current-step reasoning. Therefore, in this example, the attention flow from r_1 to f_1 should be penalized.

In practice, we calculate the attention distribution at group granularity, where each group corresponds to a text segment (premise, question, or reasoning), simplifying the computation. Specifically, supposing the reasoning involves *M* segments, we can obtain a grouped attention matrix $\boldsymbol{A}^{(g)}$.

$$a_{i,j}^{(g)} = \frac{1}{|i|} \sum_{p=i_s}^{i_e} \sum_{q=j_s}^{j_e} a_{p,q} \in \mathbb{R} \quad (3)$$

$$\boldsymbol{A}^{(g)} = [a_{i,j}^{(g)}] \in \mathbb{R}^{M \times M} \quad (4)$$

where $a_{p,q}$ is element of original attention matrix, i and j are two groups, i_s and i_e are the start and end indices of group i, and $|i|$ is the length of group i. $a_{i,j}^{(g)}$ denotes the degree of attention that the content of group i pays to that of group j.

We hypothesize that the reasoning node should focus on three components: the original question and current sub-question, the supporting premises, and the reasoning itself. Based on this hypothesis, we apply the following regularization term to

penalize the reasoning-irrelevant attention group.

$$\mathcal{L}_{reg} = \sum_{i=0}^{M} \sum_{j=0}^{M} a_{i,j}^{(g)} \tag{5}$$

where $v_j \in V_r$ and v_i is not predecessor of v_j.

For attribution reasoning rationales, we employ the vanilla causal language modeling objective for optimization. During the optimization, we mask the prompt tokens and only train on the generated content. The overall training objective can be expressed as the weighted combination of the language modeling loss and attribution regularization.

$$\mathcal{L} = \mathcal{L}_{lm} + \beta \mathcal{L}_{reg} \tag{6}$$

4 Experimental Setup

4.1 Benchmarks

We employ the mathematical reasoning dataset GSM8K (Cobbe et al., 2021) as the distillation training source. In evaluation, we conduct experiments on in-distribution and out-of-distribution mathematical reasoning, and out-of-domain general reasoning datasets to validate the effectiveness and generalization of our method.

Training (In-distribution) datasets We use the training data from GSM8K (Cobbe et al., 2021) as the training source for distillation and conduct in-distribution evaluation on its test set. Additionally, we employ GSM-Plus (Li et al., 2024), which includes human-crafted perturbed questions, as a challenging in-distribution dataset to assess the model's performance under question perturbation.

Out-of-distribution datasets In addition to grade school math series, we also use 4 mathematical reasoning datasets that were not present in the training data to evaluate the model's out-of-distribution generalization capabilities, including SVAMP (Patel et al., 2021), ASDiv (Miao et al., 2020), MultiArith (Roy and Roth, 2015), and SingleEQ (Koncel-Kedziorski et al., 2015).

Out-of-domain datasets Furthermore, to investigate the model's proficiency in general reasoning, we conduct experiments on 3 out-of-domain datasets beyond mathematics, including StrategyQA (Geva et al., 2021), Logical Deduction (Srivastava et al., 2023), and Date Understanding (Srivastava et al., 2023), which covers commonsense reasoning, logical reasoning, temporal reasoning, and symbolic reasoning. For statistics and examples of each dataset, please refer to Table 7.

4.2 Evaluation Metrics

Across all datasets, we employ Accuracy for evaluation purposes. In case of the GSM-Plus dataset, following (Li et al., 2024), we also take into account the performance drop rate (PDR) and accurately solved pairs (ASP) to evaluate the model's performance under question perturbations. Please refer to Appendix A.3 for details of metrics.

4.3 Backbone Large Language Models

We employ LLaMA3-70B-Instruct (Meta, 2024) as the teacher model and LLaMA2-7B-chat (Touvron et al., 2023), LLaMA2-13B-chat (Touvron et al., 2023), and Mistral-7B-Instruct-v0.2 (Jiang et al., 2023) as student models in our experiments.

4.4 Baselines

Few-shot CoT (Wei et al., 2022a) employs few-shot demonstrations with language models to generate step-by-step reasoning before the final answer.

Direct SFT fine-tunes on the GSM8K training set, which includes around 7k step-by-step reasoning rationales, without distillation from LLMs.

CoT Distillation leverages chain-of-thought reasoning trajectories (Wei et al., 2022a) distilled from larger models to fine-tune smaller models.

SCoTD (Li et al., 2023d) distills step-by-step reasoning data from larger models, incorporating self-consistency (Wang et al., 2023b) and correctness filtering, to fine-tune smaller models.

MCC-KD (Chen et al., 2023) distills natural language rationales from LLMs and employs KL-divergence to align diverse reasoning trajectories, thereby enhancing the generalization of reasoning.

Mathematical SFT LLMs have undergone domain-specific SFT in mathematical reasoning, including Abel (Chern et al., 2023), MammoTH (Yue et al., 2024) and MetaMath (Yu et al., 2024).

4.5 Implementation Details

In the experiment, we use AutoAWQ[2] (Lin et al., 2023) for int-4 quantization of LLaMA-3-70B-Instruct as the teacher model. During the training process, we employ Huggingface Accelerate[3] for mixed-precision training with bf16 precision. All experiments are conducted using 1 or 2 NVIDIA Tesla A100-80G GPUs. We use a linear warm-up

[2] https://github.com/casper-hansen/AutoAWQ
[3] https://github.com/huggingface/accelerate

Models	Methods	Corpus	*In-distribution*		*Out-of-distribution*				
			GSM8K	GSM-Plus	SVAMP	ASDiv	SingleEQ	MultiArith	Avg.
LLaMA3_{70b}	Fewshot CoT	-	89.98	77.04	88.60	90.46	85.82	97.16	90.51
LLaMA2_{7b}	MetaMath	NL 395K	48.33	32.83	55.20	64.98	72.19	83.83	69.05
	Abel	undisclosed	51.89	35.57	57.60	69.74	71.65	90.33	72.33
	MAmmoTH	NL 260K	50.15	33.61	53.60	65.28	72.99	86.83	69.68
	Fewshot CoT	-	24.58	17.16	46.40	57.32	62.83	69.83	59.10
	Direct SFT	CoT 7K	37.10	26.85	45.27	61.46	65.77	83.83	64.08
	CoT Distillation	CoT 74K	56.29	40.48	56.85	69.18	75.11	92.40	73.38
	SCoTD	CoT 70K	58.68	42.56	60.10	67.99	74.73	96.00	74.71
	MCC-KD	CoT 70K	58.85	43.85	57.13	70.16	**78.43**	95.27	76.31
	FARD (ours)	Attr. 68K	**61.58**	**44.48**	**64.61**	**71.82**	75.77	**98.61**	**77.70**
LLaMA2_{13b}	MetaMath	NL 395K	70.86	54.93	72.00	75.79	79.94	96.16	80.97
	Abel	undisclosed	60.47	44.60	66.00	75.47	78.61	96.33	79.10
	MAmmoTH	NL 260K	55.99	40.46	57.00	70.38	77.00	93.83	74.55
	Fewshot CoT	-	32.65	24.12	60.20	59.87	68.44	75.66	66.04
	Direct SFT	CoT 7K	48.45	35.31	55.67	67.30	73.53	90.94	71.86
	CoT Distillation	CoT 74K	66.38	50.16	67.70	74.43	78.61	96.11	79.21
	SCoTD	CoT 70K	67.08	50.79	70.40	75.89	79.31	95.88	80.37
	MCC-KD	CoT 70K	70.51	53.28	70.00	77.92	80.68	97.05	81.41
	FARD (ours)	Attr. 68K	**72.57**	**55.59**	**75.20**	**78.26**	**80.95**	**98.50**	**83.23**
Mistral_{7b}	MetaMath	NL 395K	77.54	65.52	81.60	81.84	82.62	97.50	85.89
	MAmmoTH	NL 260K	62.74	46.41	62.20	69.74	80.74	95.83	77.13
	Fewshot CoT	-	41.65	30.72	50.00	64.96	72.46	77.16	66.15
	Direct SFT	CoT 7K	56.80	43.40	61.60	69.09	77.71	93.49	75.47
	CoT Distillation	CoT 74K	70.99	55.06	74.00	75.79	80.83	96.16	81.70
	SCoTD	CoT 70K	73.82	58.15	77.60	79.77	83.55	97.33	84.56
	MCC-KD	CoT 70K	76.35	60.39	73.80	**82.59**	82.97	97.61	84.24
	FARD (ours)	Attr. 68K	**77.54**	**61.27**	**80.07**	81.46	**83.61**	**98.14**	**85.82**

Table 1: Experimental results on mathematical reasoning. Results are averaged over three independent runs, and results in gray are noted for reference purposes only. The best and second-best results are marked with **bold** and underlined. For a fair comparison, all distillation-based methods use the same teacher model, LLaMA3_{70b}-Instruct. The training starting point for student models is the aligned version (-chat or -instruct). Full results in Table 8.

for first 3% steps, followed by a cosine learning rate decay. For inference, we use VLLM[4]. The experimental results are the average of three runs. Hyperparameters details can be found in Appendix A.1. For the evaluation of mathematical reasoning and out-of-domain reasoning, we respectively employ zero-shot prompting and few-shot prompting. Details demonstrations can be found in Appendix C.

5 Experimental Results

In this section, we present the experimental results of in-distribution, out-of-distribution, as well as out-of-domain reasoning. Due to the use of additional training data in Math SFT baseline methods, which results in an unfair comparison, we only list their results for reference purposes.

5.1 In-distribution Results

Table 1 presents the in-distribution experimental results on GSM8K and GSM-Plus. It can be observed that our method significantly enhances the reasoning capabilities of smaller models, achieving performance improvements of 37.0/39.2/35.9, compared to prompt-based methods, respectively. While direct SFT also brings improvement, its performance is constrained by the limited amount and quality of data when compared to distillation-based approaches. Compared to prior baselines, our method has achieved an average improvement of 3.0% and 2.4% on GSM8K and GSM-Plus, respectively, across three student models. This indicates the effectiveness of FARD in boosting reasoning capabilities of smaller models. Meanwhile, as shown in Table 8, our method exhibits less performance drop rate under perturbation and is capable of solving both the original and perturbed questions more effectively, indicating its robustness against

[4] https://github.com/vllm-project/vllm

question perturbations. Moreover, FARD even outperforms the Mathematical SFT baselines that include additional training corpora in most cases.

Methods	*Out-of-Domain*			
	Date	Logic	SQA	Average
Student: LLaMA2$_{7b}$				
MetaMath	35.83	42.34	52.05	43.41
Abel	19.13	42.08	53.84	38.35
MAmmoTH	28.39	45.66	39.42	37.82
Fewshot CoT	38.51	51.66	57.38	49.18
Direct SFT	37.33	50.44	58.32	48.70
CoT Distillation	43.21	49.33	58.54	50.36
SCoTD	42.58	44.64	61.66	49.63
MCC-KD	39.84	50.44	59.09	49.79
Ours	**43.82**	**54.51**	**62.74**	**53.69**
Student: LLaMA2$_{13b}$				
MetaMath	42.73	-	61.35	-
Abel	47.58	40.58	57.33	48.50
MAmmoTH	55.74	51.67	52.09	53.17
Fewshot CoT	48.03	54.33	58.07	53.48
Direct SFT	53.25	53.89	57.43	54.86
CoT Distillation	55.63	54.00	57.90	55.84
SCoTD	55.37	53.55	60.20	56.37
MCC-KD	50.76	53.11	58.83	54.23
Ours	**55.82**	**55.79**	**60.95**	**57.52**
Student: Mistral$_{7b}$				
MetaMath	58.06	64.66	63.49	62.07
MAmmoTH	39.87	59.66	64.58	54.70
Fewshot CoT	52.46	63.67	58.03	58.05
Direct SFT	49.45	59.44	60.00	56.30
CoT Distillation	52.49	68.92	61.71	61.04
SCoTD	**55.32**	67.82	62.68	61.94
MCC-KD	48.66	66.55	62.70	59.30
Ours	50.51	**72.69**	**64.69**	**62.63**

Table 2: Experimental results on out-of-domain general purposed reasoning datasets. Results are averaged over three independent runs, and results in gray are noted for reference purposes only. The best and second-best results are marked with **bold** and underlined.

5.2 Out-of-distribution Results

To verify the capability of the distilled model to address similar mathematical reasoning problems, we conduct experiments on 4 out-of-distribution datasets: SVAMP, ASDiv, MultiArith, and SingleEQ. The experimental results are shown in Table 1. As distillation approaches advance, it is evident that the model's performance on out-of-distribution datasets has been steadily enhanced, though the extent of this improvement is less pronounced than that in the distribution. Despite this, our method surpasses the baselines across all three student models, with a relative improvement of 1.8%/2.2%/1.5% over the strongest baseline. Moreover, the student models empowered by FARD, even outperform the teacher model LLaMA3$_{70b}$ on SingleEQ and closely match its performance on MultiArith, demonstrating that FARD exhibits strong generalized reasoning capabilities.

5.3 Out-of-domain Results

Recent studies have found that improving a model's capabilities in specific domain can result in a decline in performance across general domains (Fu et al., 2023). This paper aims to explore how models specialized in the mathematical domain perform in out-of-domain generalized reasoning. To this end, we select 3 general purposed reasoning datasets: Date Understanding, Logical Deduction, and StrategyQA, which encompasses reasoning across logical, commonsense, temporal, and symbolic aspects. Table 2 presents the experimental results. It can be observed that direct SFT leads to a decrease in out-of-domain generalization, whereas distillation methods show an increase in such generalization. We attribute this issue to the low data diversity of direct SFT, which makes the model prone to overfitting to the distribution of mathematical reasoning. While distillation methods offer higher diversity in reasoning trajectories, making them less susceptible to overfitting and thus possessing stronger out-of-domain generalization.

Additionally, FARD consistently demonstrates stronger out-of-domain generalization capabilities, achieving performance improvements of 7.8%, 6.4%, and 5.6%. In our view, the key to improving out-of-domain generalization lies in our abstraction of the reasoning process into three components, which allows this general reasoning structure to be applicable to a wide range of reasoning tasks.

6 Analysis

6.1 Ablation Study

We conduct ablation experiments on FARD to investigate the impact of different components, as shown in Table 3. We first investigate the effect the filter (w/o filter). To ensure the correctness of the attribution structure, we perform ablation on the similarity-based diversity filter. Specifically, we randomly sample K instances from the synthetic data instead of using similarity-based filter, and use the randomly sampled rationales to train the student

Setting	GSM8K	OO-Dist.	OO-Dom.
LLaMA2$_{7b}$ (ours)	**61.6**	**77.7**	**53.7**
(a) w/o filter	59.5 (↓2.1)	76.1 (↓1.6)	51.9 (↓1.8)
(b) w/o alignment	61.2 (↓0.4)	77.3 (↓0.4)	52.9 (↓0.8)
(c) w/o self-ques	60.9 (↓0.7)	76.1 (↓1.6)	50.9 (↓2.8)
LLaMA2$_{13b}$ (ours)	**72.6**	**83.2**	**57.5**
(a) w/o filter	70.9 (↓1.7)	82.3 (↓0.9)	55.4 (↓2.1)
(b) w/o alignment	72.0 (↓0.6)	82.8 (↓0.4)	56.4 (↓1.1)
(c) w/o self-ques	72.2 (↓0.4)	81.9 (↓1.3)	56.1 (↓1.4)
Mistral$_{7b}$ (ours)	**77.5**	**85.8**	**62.6**
(a) w/o filter	76.4 (↓1.1)	84.4 (↓1.4)	58.2 (↓4.4)
(b) w/o alignment	77.1 (↓0.4)	85.3 (↓0.5)	60.1 (↓2.5)
(c) w/o self-ques	76.8 (↓0.7)	84.7 (↓1.1)	60.2 (↓2.4)

Table 3: Experimental results of the ablation study on filter, self-question, and attention regularization.

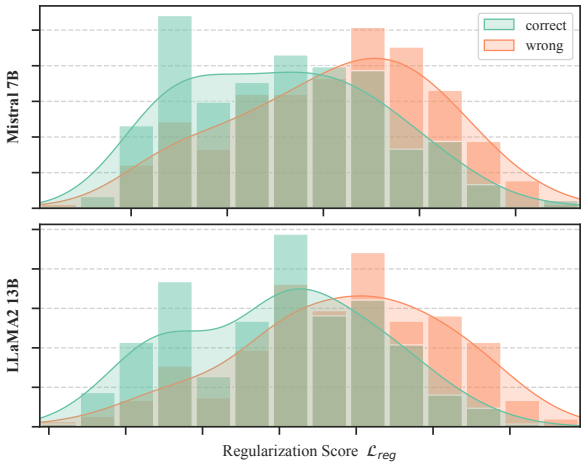

Figure 3: KDE distribution of attention regularization term in LLaMA2$_{13b}$ and Mistral$_{7b}$. Green and orange represent correct and incorrect samples, respectively.

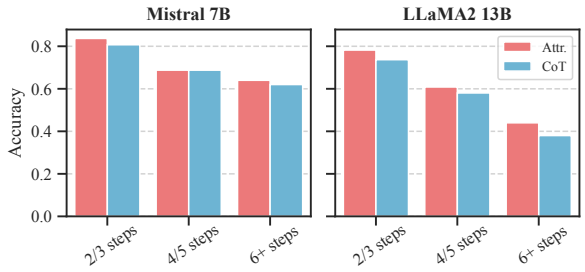

Figure 4: Performance of LLaMA2$_{13b}$ and Mistral$_{7b}$ at different reasoning steps on the GSM8K dataset.

model. As shown in Table 3(a), the removal of the filter results in a significant performance decline, which corroborates the critical role of the diversity of reasoning trajectories in the performance of small models in black-box knowledge distillation.

Afterward, we remove the attribution-based attention regularization (w/o alignment), as presented in Table 3(b). This causes a certain degree of performance decline across all tasks, indicating the positive role of reinforcing internal patterns within model reasoning. Additionally, the removal of self-questioning (w/o self-ques) within attribution reasoning also has a negative effect on model performance across all tasks, as shown in Table 3(c). This showcases the effectiveness of question decomposition when confronting complex questions.

6.2 Attention Regularization Terms

To explore the correlation between attention patterns and reasoning performance, we analyze the distribution of attention regularization terms during reasoning, as shown in Figure 3. As observed, both LLaMA2 and Mistral show different distributions between correct and incorrect samples. The regularization term is smaller in correct reasoning while larger in incorrect reasoning. Larger regularization term indicates that the model is focusing more on content unrelated to the current reasoning step. Therefore, this phenomenon might imply that irregular attention pattern is associated with a higher probability of incorrect reasoning.

6.3 Number of Reasoning Steps

We investigate the relationships between number of reasoning steps and performance, as shown in Figure 4. It is apparent that the model's performance worsens progressively with more reasoning steps. Notably, the performance decline trends vary between models. Mistral shows a smaller performance drop compared to LLaMA2. Even when performing equally on simple questions, the initially stronger model can maintain its performance on more difficult questions, while the weaker model experiences more decline. This suggests that the model's basic capabilities also play a key role in reasoning, particularly in more complex reasoning.

6.4 Error Analysis

To explore the reasons behind mistakes in attribution reasoning, we manually analyze 100 incorrect predictions by LLaMA2$_{13b}$ fine-tuned with FARD. The error distribution is shown in Table 4.

We categorize the errors into five classes: (a) Question Misunderstanding (QM): The model fails to consider or incorrectly understands critical conditions within the question. (b) Reasoning Error (RE): The model conducts flawed deductive reasoning, such as employing incorrect premises or drawing erroneous conclusions. (c) Calculation Error (CE): Arithmetic mistakes in calculations or

Error Type	Percentage
Question Misunderstanding	35%
Reasoning Error	40%
Calculation Error	19%
Decomposition Error	5%
Extraction Error	1%

Table 4: Proportion of each error type in attribution reasoning. The statistics is sourced from 100 instances of incorrect predictions that are manually analyzed.

Lora Rank	GSM8K	GSM-Plus	Math OOD
r=32 (1.17%)	59.2 (↓2.4)	42.4 (↓2.0)	76.5 (↓1.2)
***r=64* (2.31%)**	61.6	44.5	77.7
r=128 (4.53%)	63.0 (↑1.4)	45.8 (↑1.4)	79.4 (↑1.7)
r=256 (8.66%)	63.8 (↑2.2)	46.7 (↑2.2)	79.7 (↑2.0)
r=512 (15.95%)	65.2 (↑3.6)	47.2 (↑2.8)	79.6 (↑1.9)

Table 5: Experimental results in mathematical reasoning with different trainable parameters with LLaMA2$_{7b}$-FARD. The run with ***r=64*** serves as the baseline for cross comparison. Numbers in parentheses represent the percentage of trainable parameters.

solving equations. (d) Decomposition Error (DE): Erroneous question decomposition or the decomposition is unrelated to the solution. (e) Extraction Errors (EE): The model makes errors and omissions in extracting premises from the question.

As illustrated, the primary errors stem from inadequate understanding of the question and flaws in single-step deductive reasoning, which is constrained by the model's foundational capabilities. Additionally, calculation errors also pose a hindrance to mathematical reasoning performance, particularly when solving equations involving variables. The model makes relatively few mistakes in question decomposition and premise extraction.

6.5 Post Verification

Intuitively, attribution reasoning trajectories with grounded citations would provide better verifiability. We adopt an LLM [5] to post-verify the erroneous attribution reasoning trajectories, with examples shown in Table 9. It can be observed that, benefiting from self-question and grounded citations, the LLM can clearly pinpoint the location and cause of the errors during the verification process. This offers a more transparent verification process, and facilitates subsequent human-involved verification.

[5]Here we use deepseek-chat-v2

6.6 Impact of Trainable Parameters on Out-of-distribution Generalization

We conduct experiments with different lora rank r to investigate the impact of trainable parameter count on in-distribution and out-of-distribution performance, as shown in Table 5. Evidently, with the increase in the number of trainable parameters, the model's performance in mathematical reasoning consistently improves. Additionally, within the training distribution (GSM8K and GSM-Plus), the improvement continues to be substantial as the count of trainable parameters grows. However, in out-of-distribution reasoning, the performance improvement becomes insignificant as the trainable parameter scale increases. This phenomenon implies that the model's ability to generalize outside the training distribution still faces challenges.

7 Conclusion

This paper introduces Fine-Grained Attribution Reasoning Distillation (FARD) for improving the faithful reasoning capabilities of small models. FARD empowers models to generate grounded attribution citation during reasoning, thereby improving the trustworthiness and interpretability, facilitating faithful reasoning. Furthermore, FARD enhances the reasoning capabilities of models by regularizing their attention pattern based on the causal relationships between reasoning steps. Extensive experiments across nine reasoning datasets demonstrate the effectiveness of FARD. Further analysis also reveals the potential connections between attention pattern and reasoning correctness.

Limitations

FARD employs attribution reasoning to enhance the capabilities of faithful reasoning in language models, which has been validated through extensive experiments. Nevertheless, there are still some limitations. Firstly, we only use LLaMA-3-70b as the teacher model due to the computational resource constraints. Secondly, FARD provides step-level reasoning trajectories with sub-question and grounded citations, offering improved verifiability. This work has not yet conducted further reasoning verification and correction based on this structure, and we intend to address this in future work.

Acknowledgements

The research in this article is supported by the National Science Foundation of China (U22B2059,

62276083), the Human-Machine Integrated Consultation System for Cardiovascular Diseases (2023A003). We also appreciate the support from China Mobile Group Heilongjiang Co., Ltd. on our research,the research is jointly completed by both parties. Ming Liu is the corresponding author.

References

Aman Madaan Pranjal Aggarwal, Yiming Yang, and Mausam. 2023. Let's sample step by step: Adaptive-consistency for efficient reasoning and coding with llms. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 12375–12396. Association for Computational Linguistics.

Moshe Berchansky, Daniel Fleischer, Moshe Wasserblat, and Peter Izsak. 2024. Cotar: Chain-of-thought attribution reasoning with multi-level granularity. *CoRR*, abs/2404.10513.

Bernd Bohnet, Vinh Q. Tran, Pat Verga, Roee Aharoni, Daniel Andor, Livio Baldini Soares, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, Kai Hui, Tom Kwiatkowski, Ji Ma, Jianmo Ni, Tal Schuster, William W. Cohen, Michael Collins, Dipanjan Das, Donald Metzler, Slav Petrov, and Kellie Webster. 2022. Attributed question answering: Evaluation and modeling for attributed large language models. *CoRR*, abs/2212.08037.

Hongzhan Chen, Siyue Wu, Xiaojun Quan, Rui Wang, Ming Yan, and Ji Zhang. 2023. MCC-KD: multi-cot consistent knowledge distillation. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 6805–6820. Association for Computational Linguistics.

Ethan Chern, Haoyang Zou, Xuefeng Li, Jiewen Hu, Kehua Feng, Junlong Li, and Pengfei Liu. 2023. Generative ai for math: Abel. `https://github.com/GAIR-NLP/abel`.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Tao He, Haotian Wang, Weihua Peng, Ming Liu, Bing Qin, and Ting Liu. 2024. Navigate through enigmatic labyrinth a survey of chain of thought reasoning: Advances, frontiers and future. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1173–1203, Bangkok, Thailand. Association for Computational Linguistics.

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. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Dheeru Dua, Shivanshu Gupta, Sameer Singh, and Matt Gardner. 2022. Successive prompting for decomposing complex questions. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 1251–1265. Association for Computational Linguistics.

Constanza Fierro, Reinald Kim Amplayo, Fantine Huot, Nicola De Cao, Joshua Maynez, Shashi Narayan, and Mirella Lapata. 2024. Learning to plan and generate text with citations. *CoRR*, abs/2404.03381.

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023. Specializing smaller language models towards multi-step reasoning. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 10421–10430. PMLR.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y. Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023a. RARR: researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 16477–16508. Association for Computational Linguistics.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023b. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 6465–6488. Association for Computational Linguistics.

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. *Trans. Assoc. Comput. Linguistics*, 9:346–361.

Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. 2023. The false promise of imitating proprietary llms. *CoRR*, abs/2305.15717.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531.

Yifan Hou, Jiaoda Li, Yu Fei, Alessandro Stolfo, Wangchunshu Zhou, Guangtao Zeng, Antoine Bosselut, and Mrinmaya Sachan. 2023. Towards a mechanistic interpretation of multi-step reasoning capabilities of language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 4902–4919. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Lei Huang, Xiaocheng Feng, Weitao Ma, Yuxuan Gu, Weihong Zhong, Xiachong Feng, Weijiang Yu, Weihua Peng, Duyu Tang, Dandan Tu, and Bing Qin. 2024. Learning fine-grained grounded citations for attributed large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 14095–14113, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *CoRR*, abs/2311.05232.

Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. 2022. Knowledge distillation from A stronger teacher. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Taehyeon Kim, Jaehoon Oh, Nakyil Kim, Sangwook Cho, and Se-Young Yun. 2021. Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 2628–2635. ijcai.org.

Rik Koncel-Kedziorski, Hannaneh Hajishirzi, Ashish Sabharwal, Oren Etzioni, and Siena Dumas Ang. 2015. Parsing algebraic word problems into equations. volume 3, pages 585–597.

Chenglin Li, Qianglong Chen, Caiyu Wang, and Yin Zhang. 2023a. Mixed distillation helps smaller language model better reasoning. *CoRR*, abs/2312.10730.

Dongfang Li, Zetian Sun, Xinshuo Hu, Zhenyu Liu, Ziyang Chen, Baotian Hu, Aiguo Wu, and Min Zhang. 2023b. A survey of large language models attribution. *CoRR*, abs/2311.03731.

Dongfang Li, Zetian Sun, Xinshuo Hu, Zhenyu Liu, Ziyang Chen, Baotian Hu, Aiguo Wu, and Min Zhang. 2023c. A survey of large language models attribution. *CoRR*, abs/2311.03731.

Liunian Harold Li, Jack Hessel, Youngjae Yu, Xiang Ren, Kai-Wei Chang, and Yejin Choi. 2023d. Symbolic chain-of-thought distillation: Small models can also "think" step-by-step. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 2665–2679. Association for Computational Linguistics.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. 2024. Gsm-plus: A comprehensive benchmark for evaluating the robustness of llms as mathematical problem solvers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 2961–2984. Association for Computational Linguistics.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Xingyu Dang, and Song Han. 2023. Awq: Activation-aware weight quantization for llm compression and acceleration. *arXiv*.

Meta. 2024. Introducing meta llama 3.

Shen-Yun Miao, Chao-Chun Liang, and Keh-Yih Su. 2020. A diverse corpus for evaluating and developing english math word problem solvers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 975–984. Association for Computational Linguistics.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. 2019. Relational knowledge distillation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 3967–3976. Computer Vision Foundation / IEEE.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. 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, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2080–2094. Association for Computational Linguistics.

Subhro Roy and Dan Roth. 2015. Solving general arithmetic word problems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1743–1752, Lisbon, Portugal. Association for Computational Linguistics.

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 Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew K. 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 Karakas, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartlomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, Cèsar Ferri Ramírez, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris 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, 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 Tam, Dieuwke 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 J. Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, François Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Wang, Gonzalo Jaimovitch-López, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, Hinrich Schütze, 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 H. 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, José Hernández-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 D. Dhole, Kevin Gimpel, Kevin Omondi, Kory 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, Luheng 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, María José Ramírez-Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L. Leavitt, Matthias Hagen, Mátyás Schubert, Medina Baitemirova, Melody Arnaud, Melvin McElrath, Michael A. Yee, Michael Cohen, Michael Gu, Michael I. Ivanitskiy, Michael Starritt, Michael Strube, Michal Swedrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T., Nanyun Peng, Nathan A. Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha 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 Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Milkowski, 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 LeBras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Ruslan Salakhutdinov, Ryan Chi, 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 S. 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 T. Piantadosi, Stuart M. Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsu 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 V. Ramasesh, Vinay Uday Prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, 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. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Trans. Mach. Learn. Res.*, 2023.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for BERT model compression. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 4322–4331. Association for Computational Linguistics.

Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. 2021. ProofWriter: Generating implications, proofs, and abductive statements over natural language. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3621–3634, Online. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Peifeng Wang, Zhengyang Wang, Zheng Li, Yifan Gao, Bing Yin, and Xiang Ren. 2023a. SCOTT: self-consistent chain-of-thought distillation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 5546–5558. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Zhaoyang Wang, Shaohan Huang, Yuxuan Liu, Jiahai Wang, Minghui Song, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. 2023c. Democratizing reasoning ability: Tailored learning from large language model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1948–1966, Singapore. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022a. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022b. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Shicheng Xu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. 2023. Search-in-the-chain: Towards the accurate, credible and traceable content generation for complex knowledge-intensive tasks. *CoRR*, abs/2304.14732.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2024. Metamath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2024. Mammoth: Building math generalist models through hybrid instruction tuning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Xiang Yue, Boshi Wang, Ziru Chen, Kai Zhang, Yu Su, and Huan Sun. 2023. Automatic evaluation of attribution by large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 4615–4635. Association for Computational Linguistics.

Sergey Zagoruyko and Nikos Komodakis. 2017. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023a. Siren's song in the AI ocean: A survey on hallucination in large language models. *CoRR*, abs/2309.01219.

Zhiheng Zhang, Daojian Zeng, and Xue Bai. 2024. Improving continual few-shot relation extraction through relational knowledge distillation and prototype augmentation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 8756–8767. ELRA and ICCL.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2023b. Automatic chain of thought prompting in large language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

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. 2023. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Xuekai Zhu, Biqing Qi, Kaiyan Zhang, Xinwei Long, Zhouhan Lin, and Bowen Zhou. 2024a. PaD: Program-aided distillation can teach small models reasoning better than chain-of-thought fine-tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2571–2597, Mexico City, Mexico. Association for Computational Linguistics.

Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. 2023. A survey on model compression for large language models. *CoRR*, abs/2308.07633.

Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. 2024b. Distilling mathematical reasoning capabilities into small language models. *Preprint*, arXiv:2401.11864.

A Appendix

A.1 Hyperparameters

In this section, we will provide a detailed the hyperparameters used in our experiments, as shown in Table 6. During the data synthesis process, we set the temperature to 1.3, sampling *N=32* instances for each training instance, and use filter to retain up to *K=10* samples. We train 4 epochs for direct SFT baselines, 10 epochs for MCC-KD and 2 epochs for other methods. The total batch size during training process is set to 32. During training, we employ LoRA (Hu et al., 2022) for parameter-efficient fine-tuning with huggingface PEFT[6]. We add LoRA adapters to all linear modules. For 7B models, the lora_rank/lora_alpha is set to 64/128, with learning rate of 4e-5. For 13B models, the lora_rank/lora_alpha is set to 128/256, with learning rate of 8e-5. We linearly warm up the learning rate from 0 during the first 3% of the steps, after which we decay it to 0 using cosine annealing. The checkpoint at the end of the training will be used for subsequent evaluation. In the inference stage, we set the temperature to 0 for greedy decoding and do not use beam search.

Hyperparameters	Values
Epochs	2 or 4 or 10
Learning Rate	4e-5 or 8e-5
Batch Size	32
β	1e-2
Warmup Ratio	0.03
LR Decay	Cosine Annealing
LoRA Rank	64 or 128
LoRA Alpha	128 or 256
LoRA Targets	q, k, v, o
τ (Data Generation)	1.3
τ (Inference)	0.0
N	32
K	10

Table 6: Hyperparameters in experiments.

A.2 Rationale Filter

Rationale Validity Filter We first remove any unnecessary line breaks from the reasoning, and then determine whether the reasoning adheres to the three-part structure (Facts, Reasoning, and Answer). Following this, each reasoning step should feature a numerical tag for itself and tags of the premise it depends on, and should be associated to one sub-question. Finally, we parse the reasoning dependency to ensure the coherence between reasoning nodes, removing any instance that do not form a DAG or contain incorrect citation tags. Through the above procedures, we are able to derive attribution reasoning rationales that are both structurally valid and capable of being parsed.

Similarity-based Filter We iteratively remove the most similar samples to others from the rationales set until the number of remaining samples reaches our target. For computational efficiency, we adopt a sparse similarity metric, Jaccard Similarity, which gauges the degree of overlap between two sets, as referenced from (Chen et al., 2023).

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{7}$$

Specifically, we transform the rationales into N-gram, and consider Jaccard Similarity between two N-gram sets as the similarity between the two rationales. In each iteration, we remove one of the two rationales with the highest similarity, continuing this process until the sample size reach *K=10*.

A.3 Evaluation Metrics

Following (Li et al., 2024), we use two metrics, ASP and PDR, to assess the model's performance in the presence of perturbed questions.

Performance Drop Rate (PDR) measures the reduction in a model's performance when it is presented with perturbed questions relative to the performance on the original questions (lower is better).

$$PDR = 1 - \frac{\sum_{(x,y)\in\mathcal{D}_a} \mathbb{I}[LM(x), y]/|\mathcal{D}_a|}{\sum_{(x,y)\in\mathcal{D}} \mathbb{I}[LM(x), y]/|\mathcal{D}|}, \tag{8}$$

where $\mathcal{D}$ and $\mathcal{D}_a$ denote GSM8K and GSM-Plus.

Accurately Solved Pairs (ASP) serves as a metric to evaluate the percentage of cases in which the model successfully addresses both the original question and its perturbed version (higher is better).

$$ASP = \frac{\sum_{x,y;x',y'} \mathbb{I}[LM(x), y] \cdot \mathbb{I}[LM(x'), y']}{N \cdot |\mathcal{D}|}, \tag{9}$$

The overall results in Table 8 are calculated as: $overall = Acc_{gsm8k} + Acc_{gp} + ASP - PDR$.

[6] https://github.com/huggingface/peft

B Datasets

This section will provide a detailed introduction to the datasets used in this paper. We employ GSM8K as the source of synthetic data for training, and subsequently evaluate our method on in-distribution mathematical reasoning, out-of-distribution mathematical reasoning, and out-of-domain general reasoning tasks. The statistics and examples of the datasets can be found in Table 7.

For mathematical reasoning, we only used the GSM8K training set to generate reasoning rationales, with the remaining datasets being used for evaluation and not included in the training data.

- GSM8K (Cobbe et al., 2021) comprises a collection of high-quality, linguistically diverse grade school math word problems, developed by human authors. We use its training set to generate training data and employ the test set for evaluation.[7]
- GSM-Plus (Li et al., 2024) is an enhanced version of GSM, designed to test the robustness of large language models' mathematical reasoning by introducing various perturbations. We use the dataset in the evaluation.[8]
- SVAMP (Patel et al., 2021) is a challenging dataset that includes math word problems (MWP) targeted at students below grades four. We use the dataset for evaluation.[9]
- ASDiv (Miao et al., 2020) is a English math word problem corpus, which entails diverse language patterns and problem types. We use the dataset in the evaluation.[10]
- MultiArith (Roy and Roth, 2015) includes simple MWPs involving the four basic operations. We use the dataset in the evaluation.[11]
- SingleEQ (Koncel-Kedziorski et al., 2015) features basic arithmetic problems. We use the dataset in the evalution.[12]

For out-of-domain reasoning, we do not use any related data for training and directly conduct evaluation under few-shot setting.

- Date Understanding is a task in BIG-bench (Srivastava et al., 2023). It assesses large language models' ability to comprehend dates with implicit temporal connections and knowledge, covering temporal, mathematical, and commonsense reasoning.[13]
- Object Logical Deduction is a task in BIG-bench (Srivastava et al., 2023). It requires models to determine the order of objects based on deductive logical reasoning. This dataset involves reasoning and symbolic reasoning.[14]
- StrategyQA was introduce by Geva et al. (2021). It requires models to address implicit, multi-step questions, involving multi-step commonsense reasoning. We use the test set provided in BIG-bench in our evaluation.[15]

C Prompts

In this section, we will present the prompts used for rationales generation and zero-shot/few-shot reasoning in our experiments.

During rationale generation process, we distill attribution reasoning rationales from the teacher large language model using 4-shot demonstrations, as shown in Table 10 and Table 11.

In the inference stage, we evaluate mathematical reasoning tasks (in-distribution and out-of-distribution) under zero-shot setting, and adopting the few-shot approach for out-of-domain reasoning tasks. Table 12 presents the prompt templates of the SFT models, which are also their zero-shot instructions for mathematical reasoning. Tables 13, 14, and 15 respectively show the few-shot examples used for out-of-domain reasoning in Date Understanding, Logical Deduction, and StrategyQA. Substituting these examples into the corresponding prompt templates in Table 12 yields the few-shot prompts used during inference.

[7] https://github.com/openai/grade-school-math

[8] https://github.com/qtli/GSM-Plus

[9] https://github.com/arkilpatel/SVAMP/blob/main/SVAMP.json

[10] https://github.com/chaochun/nlu-asdiv-dataset

[11] https://github.com/wangxr14/Algebraic-Word-Problem-Solver/blob/master/data/MultiArith.json

[12] https://gitlab.cs.washington.edu/ALGES/TACL2015/-/blob/master/questions.json?ref_type=heads

[13] https://github.com/google/BIG-bench/blob/main/bigbench/benchmark_tasks/date_understanding/task.json

[14] https://github.com/google/BIG-bench/blob/main/bigbench/benchmark_tasks/logical_deduction/three_objects/task.json

[15] https://github.com/google/BIG-bench/blob/main/bigbench/benchmark_tasks/strategyqa

Datasets	#Examples	Question Example
(a) Mathematical Reasoning (In-distribution)		
GSM8K	1319	Eliza's rate per hour for the first 40 hours she works each week is $10. She also receives an overtime pay of 1.2 times her regular hourly rate. If Eliza worked for 45 hours this week, how much are her earnings for this week?
GSM-Plus	1319	Eliza earns $10 per hour for the first 40 hours she works each week and 1.2 times her regular hourly rate for any overtime. If Eliza earned $470 this week, how many hours did she work?
(b) Mathematical Reasoning (Out-of-distribution)		
SVAMP	500	There were 13 roses in the vase. Jessica cut some more roses from her flower garden which had a total of 12 roses. There are now 21 roses in the vase. How many roses are left in the garden?
ASDiv	314	Jennifer has an 87 cm long ribbon. She uses 24 cm of the ribbon to tie a present for her friend and 45 cm of the ribbon to make a bow. How much of the ribbon is left in the end?
MultiArith	600	Paige had 11 songs on her mp3 player. If she deleted 9 old songs from it and then added 8 new songs, how many songs does she have on her mp3 player?
SingleEQ	374	Mary is baking a cake. The recipe wants 8 cups of flour. She has put in 2 cups. How many more cups does she need to add?
(c) General Reasoning (Out-of-domain)		
Date Understanding	369	In the US, Thanksgiving is on the fourth Thursday of November. Today is the US Thanksgiving of 2001. What is the date one year ago from today in MM/DD/YYYY?
Logical Deduction	300	In a golf tournament, there were three golfers: Eve, Rob, and Mel. Rob finished below Mel. Mel finished below Eve. Who finished first? A. Eve B. Rob C. Mel
StrategyQA	2290	Was Aristotle a member of the House of Lords?

Table 7: Statistics and examples of datasets in in-distribution, out-of-distribution, and out-of-domain evaluations.

Models	Methods	Corpus	GSM8K	GSM-Plus	△ PDR (%) ↓	ASP (%) ↑	Overall
LLaMA3_{70b}	Fewshot CoT	-	89.98	77.04	14.38	73.36	226.00
LLaMA2_{7b}	MetaMath	NL 395K	48.33	32.83	32.06	24.51	73.61
	Abel	undisclosed	51.89	35.57	31.45	26.31	82.32
	MAmmoTH	NL 260K	50.15	33.61	32.98	25.94	76.72
	Fewshot CoT	-	24.58	17.16	30.15	9.33	20.92
	Direct SFT	CoT 7K	37.10	26.85	27.70	17.62	53.87
	CoT Distillation	CoT 74K	56.29	40.48	32.87	28.08	101.57
	SCoTD	CoT 70K	58.68	42.56	35.56	27.46	109.35
	MCC-KD	CoT 70K	58.85	43.85	**25.47**	35.89	113.12
	ARD (ours)	Attr. 68K	**61.58**	**44.48**	27.77	**37.40**	**115.69**
LLaMA2_{13b}	MetaMath	NL 395K	70.86	54.93	22.43	47.97	151.30
	Abel	undisclosed	60.47	44.60	35.79	26.24	114.62
	MAmmoTH	NL 260K	55.99	40.46	31.38	27.73	100.10
	Fewshot CoT	-	32.65	24.12	26.04	14.66	45.39
	Direct SFT	CoT 7K	48.45	35.31	27.11	25.79	82.44
	CoT Distillation	CoT 74K	66.38	50.16	24.39	42.52	134.67
	SCoTD	CoT 70K	67.08	50.79	24.27	43.56	137.16
	MCC-KD	CoT 70K	70.51	53.28	24.43	46.13	145.49
	ARD (ours)	Attr. 68K	**72.57**	**55.59**	**23.40**	**49.13**	**153.89**
Mistral_{7b}	MetaMath	NL 395K	77.54	65.52	19.39	56.08	176.75
	MAmmoTH	NL 260K	62.74	46.41	38.30	26.03	121.42
	Fewshot CoT	-	41.65	30.72	26.22	20.44	66.59
	Direct SFT	CoT 7K	56.80	43.40	23.58	32.90	109.51
	CoT Distillation	CoT 74K	70.99	55.06	22.42	47.97	151.57
	SCoTD	CoT 70K	73.82	58.15	21.22	51.55	162.31
	MCC-KD	CoT 70K	76.35	60.39	**20.89**	53.82	169.67
	ARD (ours)	Attr. 68K	**77.54**	**61.27**	20.98	**55.35**	**173.18**

Table 8: Experimental results on in-distribution datasets, GSM8K and GSM-Plus, including question perturbation metrics PDR and ASP. Results are averaged over three runs, and results in gray are noted for reference purposes only. For a fair comparison, all distillation methods use the same teacher, LLaMA3_{70b}. The starting point for student models is the aligned version (-chat or -instruct). Best and second results are highlighted by **bold** and underline.

Post Verification on Incorrect Attribution Reasoning Trajectory

Question: There are 96 fourth-graders at Small Tree School. 43 of them are girls. On Friday, 5 fourth-grade girls and 4 fourth-grade boys were absent.
How many fourth grade boys were at Small Tree School on Friday? (Ground truth: 49)

✗

Incorrect reasoning before verification:
[Facts]
[1] There are 96 fourth-graders at Small Tree School.
[2] 43 of them are girls.
[3] 5 fourth-grade girls were absent.
[4] 4 fourth-grade boys were absent.
[Reasoning]
Calculate the number of fourth-grade boys initially:
[5] There are 96 fourth-graders, and 43 are girls, so there are 96 - 43 = 53 fourth-grade boys initially. [1, 2]
Calculate the number of fourth-grade boys present on Friday:
[6] 5 fourth-grade girls were absent, and 4 fourth-grade boys were absent, so there are 53 - 5 - 4 = 44 fourth-grade boys present on Friday. [3, 4, 5]
[Answer]
The answer is 44.

✓

Reasoning after post-verification:
The reasoning provided contains an error in the calculation of the number of fourth-grade boys present on Friday. The error is in step **[6]**, where the number of absent girls is incorrectly subtracted from the number of boys. Here is the corrected reasoning:
[Facts]
[1] There are 96 fourth-graders at Small Tree School.
[2] 43 of them are girls.
[3] 5 fourth-grade girls were absent.
[4] 4 fourth-grade boys were absent.
[Reasoning]
Calculate the number of fourth-grade boys initially:
[5] There are 96 fourth-graders, and 43 are girls, so there are 96 - 43 = 53 fourth-grade boys initially. [1, 2]
Calculate the number of fourth-grade boys present on Friday:
[6] 4 fourth-grade boys were absent, so there are 53 - 4 = 49 fourth-grade boys present on Friday. [4, 5]
[Answer]
The answer is **49**.

Table 9: Post-verification of incorrect attribution reasoning trajectories. Benefiting from the attribution structure, the LLM can clearly pinpoint the location, step, and cause of errors in the verification process.

Few-shot Demonstrations for Attribution Reasoning Generation (Part I)

1

[Question]
A bag of jellybeans contains 8 black beans and 2 more green beans. It also contains 1 less orange bean than green beans. How many jellybeans are there in the bag?
[Facts]
[1] A bag of jellybeans contains 8 black beans and 2 more green beans.
[2] A bag of jellybeans also contains 1 less orange bean than green beans.
[Reasoning]
Calculate the number of green beans:
[3] There are 8 black beans and 2 more green beans, therefore there are 8 + 2 = 10 green beans. [1]
Calculate the number of orange beans:
[4] The bag contains 1 less orange bean than green bean, therefore there are 10 - 1 = 9 orange beans. [2, 3]
Calculate the total number of beans in the bag:
[5] Now there are 8 black beans, 10 green beans and 9 orange beans, therefore there are a total of 8 + 10 + 9 = 27 beans. [1, 3, 4]
[Answer]
The answer is 27.

2

[Question]
Rebecca runs a hair salon. She charges $30 for haircuts, $40 for perms, and $60 for dye jobs, but she has to buy a box of hair dye for $10 to dye every head of hair. Today, she has four haircuts, one perm, and two dye jobs scheduled. If she makes $50 in tips, how much money will she have in dollars at the end of the day?
If she makes $50 in tips, how much money will she have in dollars at the end of the day?
[Facts]
[1] Rebecca charges $30 for haircuts.
[2] Rebecca charges $40 for perms.
[3] Rebecca charges $60 for dye jobs.
[4] She has to buy a box of hair dye for $10 to dye every head of hair.
[5] Today, she has four haircuts, one perm, and two dye jobs scheduled.
[6] she makes $50 in tips.
[Reasoning]
Calculate how much money Rebecca made from haircuts:
[7] Rebecca makes $30 * 4 = $120 from haircuts. [1, 5]
Calculate how much money Rebecca made from perms:
[8] Rebecca makes $40 * 1 = $40 from perms. [2, 5]
Calculate how much money Rebecca made from dye jobs:
[9] Rebecca makes $60 * 2 = $120 from dye jobs. [3, 5]
Calculate how much money Rebecca cost on buying hair dye:
[10] Rebecca cost $10 * 2 = $20 to buy hair dye. [4, 5]
Calculate how much money Rebecca made:
[11] She makes $120 + $40 + $120 + $50 = $330 in total. [6, 7, 8, 9]
Calculate how much money Rebecca cost:
[12] She cost $20 in total. [10]
Calculate the money Rebecca earned in total:
[13] The money she earns is equal to the income minus the cost, therefore, she makes $330 - $20 = $310 today. [11, 12]
[Answer]
The answer is $310.

Table 10: Few-shot (4-shot) demonstrations for attribution reasoning rationales generation. (Part I)

Few-shot Demonstrations for Attribution Reasoning Generation (Part II)

[Question]
There are four members in one household. Each member consumes 3 slices of bread during breakfast and 2 slices of bread for snacks. A loaf of bread has 12 slices. How many days will five loaves of bread last in this family?
[Facts]
[1] There are 4 members in one household.
[2] Each member consumes 3 slices of bread during breakfast.
[3] Each member consumes 2 slices of bread for snacks.
[4] A loaf of bread has 12 slices.
[5] There are 5 loaves of bread.
[Reasoning]
3 Calculate how many slices of bread each family member consumes per day:
[6] Each member consumes 3 + 2 = 5 slices of bread in total per day. [2, 3]
Calculate how many slices of bread the family consumes per day:
[7] There are 4 members, so the family consumes 4 * 5 = 20 slices of bread in total per day. [1, 6]
Calculate how many slices of bread are there in 5 loaves of bread:
[8] There are 5 loaves of bread, with each loaf containing 12 slices. Therefore, there are 5 * 12 = 60 slices of bread in total. [4, 5]
Calculate how many days these loaves of bread will last a family:
[9] The family consumes 20 slices of bread per day, therefore the bread last 60 / 20 = 3 days in this family. [7, 8]
[Answer]
The answer is 3 days.

[Question]
At the salad bar, Grandma put three mushrooms on her salad. She also added twice as many cherry tomatoes as mushrooms, 4 times as many pickles as cherry tomatoes, and 4 times as many bacon bits as pickles. If one third of the bacon bits were red, than how many red bacon bits did Grandma put on her salad?
[Facts]
[1] Grandma put 3 mushrooms on her salad.
[2] She also added twice as many cherry tomatoes as mushrooms.
[3] She also added 4 times as many pickles as cherry tomatoes.
[4] She also added 4 times as many bacon bits as pickles.
[5] one third of the bacon bits were red.
4 [Reasoning]
Calculate the number of cherry tomatoes:
[6] She added twice as many cherry tomatoes as mushrooms, so there are 2 * 3 = 6 cherry tomatoes. [1, 2]
Calculate the number of pickles:
[7] She added 4 times as many pickles as cherry tomatoes, so there are 4 * 6 = 24 pickles. [2, 3]
Calculate the number of bacon bits:
[8] She added 4 times as many bacon bits as pickles, so there are 4 * 24 = 96 bacon bits. [3, 4]
Calculate the number of red bacon bits:
[9] One third of the bacon bits were red, so there are 1/3 * 96 = 32 red bacon bits. [5, 8]
[Answer]
The answer is 32.

Table 11: Few-shot (4-shot) demonstrations for attribution reasoning rationales generation. (Part II)

Methods	Prompt Templates
MetaMath	Below is an instruction that describes a task. Write a response that appropriately completes the request. ### Instruction: {Instruction} ### Response: Let's think step by step.
Abel	Question {Instruction} Answer: Let's think step by step.
MAmmoTH	Below is an instruction that describes a task. Write a response that appropriately completes the request. ### Instruction: {Instruction} ### Response:
CoT	Question: {Instruction}
Attribution	[Question]: {Instruction}

Table 12: Prompt templates and zero-shot instructions for SFT LLMs.

Few-shot Examples of Date Understanding Tasks

Input:
[Question]
Yesterday was April 30, 2021. What is the date one week ago from today in MM/DD/YYYY?
Output:
[Facts]
[1] Yesterday was April 30, 2021.
1 [Reasoning]
Calculate the date of today:
[2] If yesterday was April 30, 2021, then today would be May 1, 2021 (05/01/2021). [1]
Calculate the date one week ago from today:
[3] If today is May 1, 2021, then one week ago from today would be April 24, 2021. [2]
[Answer]
Therefore, the answer is 04/24/2021.

Input:
[Question]
Jane and John married on Jan 2, 1958. Today is their golden wedding anniversary. What is the date tomorrow in MM/DD/YYYY?
Output:
[Facts]
[1] Jane and John married on Jan 2, 1958.
[2] Today is their golden wedding anniversary.
2 [Reasoning]
Calculate the date today:
[3] Since today is is their golden wedding anniversary, which is the 50th anniversary of a marriage and Jane and John married on Jan 2, 1958. Adding 50 years to Jan 2, 1958 get Jan 2, 2008. So today is Jan 2, 2008 (01/02/2008). [1, 2]
Calculate the date tomorrow:
[4] Today is 01/02/2008, so tomorrow is 01/03/2008. [3]
[Answer]
Therefore, the answer is 01/03/2008.

Input:
[Question]
The day before yesterday was 11/23/1933. What is the date one week from today in MM/DD/YYYY?
Output:
[Facts]
[1] The day before yesterday was 11/23/1933.
3 [Reasoning]
Calculate the date today:
[2] The day before yesterday was 11/23/1933, today is 2 days after 11/23/1933, so today is 11/25/1933. [1]
Calculate the date one week from today:
[3] Today is 11/25/1933, and one week is 7 days. So 7 days from today is 12/02/1933. [2]
[Answer]
Therefore, the answer is 12/02/1933.

Table 13: Few-shot examples used in out-of-domain date understanding tasks.

	Few-shot Examples of Object Logical Deduction Tasks
1	**Input:** [Question] On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book. What is the leftmost? A. The black book B. The orange book C. The blue book **Output:** [Facts] [1] The blue book is to the right of the orange book. [2] The orange book is to the right of the black book. [Reasoning] [3] The blue book is to the right of the orange book, so the order from left to right is [orange book] [blue book] [1] [4] The orange book is to the right of the black book, so the black book is to the left of the orange book. The order from left to right is [black book] [orange book] [blue book] [2, 3] [5] In this order, the leftmost is the black book. [4] [Answer] Therefore, the answer is A. The black book.
2	**Input:** [Question] On a branch, there are three birds: a quail, an owl, and a hummingbird. The quail is to the left of the owl. The owl is to the left of the hummingbird. What is the second from the left? A. The quail B. The owl C. The hummingbird **Output:** [Facts] [1] The quail is to the left of the owl. [2] The owl is to the left of the hummingbird. [Reasoning] [3] The quail is to the left of the owl, so the order from left to right is [quail] [owl] [1] [4] The owl is to the left of the hummingbird, so the hummingbird is to the right of the owl, the order from left to right is [quail] [owl] [hummingbird]. [2, 3] [5] In this order, the second from the left is the owl. [4] [Answer] Therefore, the answer is B. the owl.
3	**Input:** [Question] In an antique car show, there are three vehicles: a sedan, a tractor, and a bus. The sedan is older than the tractor. The bus is older than the sedan. What is the second-newest? A. The sedan B. The tractor C. The bus **Output:** [Facts] [1] The sedan is older than the tractor. [2] The bus is older than the sedan. [Reasoning] [3] The sedan is older than the tractor, so the order from old to new is [sedan] [tractor] [1] [4] The bus is older than the sedan, so the order from old to new is [bus] [sedan] [tractor] [2, 3] [5] In this order, the second-newest is the sedan. [Answer] Therefore, the answer is A. The sedan.

Table 14: Few-shot examples used in out-of-domain object logical deduction task.

	Few-shot Examples of StrategyQA Tasks
1	**Input:** [Question] Would a block of iron sink in water? **Output:** [Facts] [1] The density of iron is 7.86 g/cm³ [2] The density of water is 1 g/cm³. [Reasoning] [3] Since the density of iron (7.86 g/cm³) is greater than the density of water (1 g/cm³), the block of iron would sink in water. [1, 2] [Answer] Therefore, the answer is Yes.
2	**Input:** [Question] Did Aristotle use a laptop? **Output:** [Facts] [1] Aristotle lived from 384 to 322 BCE. [2] Laptops and modern computing devices were not invented until thousands of years later. [Reasoning] [3] Given the time period in which Aristotle lived, it is impossible for him to have used a laptop. [1, 2] [Answer] Therefore, the answer is No

Table 15: Few-shot examples used in out-of-domain StrategyQA tasks.