Learning from Committee: Reasoning Distillation from a Mixture of Teachers with Peer-Review

Anonymous ACL submission

Abstract

While reasoning capabilities typically emerge in large language models (LLMs) with tens of billions of parameters, recent research focuses on improving smaller open-source models through knowledge distillation (KD) from commercial LLMs. However, many of these studies rely solely on responses from a single LLM as the gold rationale, unlike the natural human learning process, which involves understanding both the correct answers and the reasons behind mistakes. In this paper, we introduce a novel **F**ault-**A**ware Dist**I**llation via Peer-**R**eview (**FAIR**) approach: 1) Instead of merely obtaining rationales from teachers, our method asks teachers to identify and explain the student's mistakes, providing customized instruction learning data. 2) We design a simulated peer-review process between teacher LLMs, which selects only the generated rationales above the acceptance threshold. This reduces the chance of teachers guessing correctly with flawed rationale, improving instructional data quality. Comprehensive experiments and analysis on mathematical, commonsense, and logical reasoning tasks demonstrate the effectiveness of our method.

1 Introduction

Large Language Models (LLMs) have proven to be highly effective in addressing a wide range of complex tasks (Ni et al., 2024; Fan and Tao, 2024), including mathematical reasoning (Lewkowycz et al., 2022; Imani et al., 2023), commonsense reasoning (Zhao et al., 2024; Achiam et al., 2023), and logical reasoning (Liu et al., 2023; Xu et al., 2023b). However, these emergent reasoning abilities tend to manifest only in LLMs with more than 100 billion parameters, while smaller models struggle to exhibit such capabilities (Wei et al., 2022a). Despite this, related research (Touvron et al., 2023; Zeng et al., 2022) has shown that smaller language models, particularly those with fewer than 10 billion parameters, can perform similarly to larger

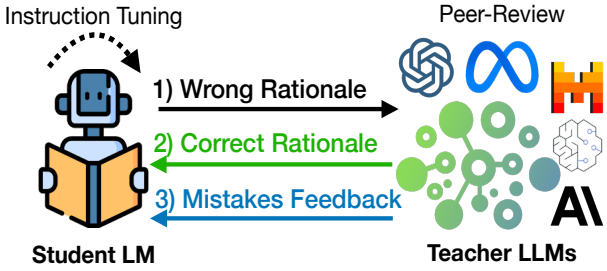

Figure 1: Student LM learns from multiple teacher LLMs via Peer-Review distillation.

models in terms of following human instructions. However, it is challenging to prompt smaller Language Models (LMs) to generate reasoning steps by Chain-of-Thought (CoT) prompts (Wang et al., 2023). Moreover, most existing reasoning datasets lack high-quality rationale (Gurrapu et al., 2023) due to the high cost of manual annotations.

To address these challenges, distilling the capabilities of LLMs emerges as a resource-friendly and effective strategy. DeepSeek-R1 (Guo et al., 2025) demonstrates that distilling reasoning patterns from larger models outperform RL-derived patterns on smaller models. Through collecting rationales generated by LLMs for instruction tuning, previous studies have been able to distill the private LLMs' reasoning abilities into smaller models (Wang et al., 2022; Ho et al., 2023; Magister et al., 2022; Fu et al., 2023). However, most of these efforts fall within the scope of Labeling Knowledge Distillation (Xu et al., 2024b), where LLMs are primarily used to annotate data for training smaller models, without utilizing smaller model's output as feedback to generate customized instruction data to improve the LM in return. As a result, LLMs remain unaware of the limitations of smaller models.

Furthermore, prior research typically employs only one LLM as the teacher, which can introduce more biased training data compared to using multiple teacher LLMs during distillation. Therefore, we propose using multiple LLMs from different organizations as teachers to provide more impartial

and diverse training data. Additionally, we design a simulated peer-review process between teacher LLMs, where the rationale generated by one LLM is reviewed by other LLMs. Only the rationales that pass this peer-review process are included in the training dataset. This method reduces the likelihood of flawed rationales, even when a correct answer is provided, thereby improving the overall quality of the training data for instruction tuning.

To this end, we propose a Fault-Aware Distillation via Peer-Review (FAIR) knowledge distillation method from multiple LLMs, as briefly shown in Figure 1. Inspired by the natural human learning process (Konold et al., 2004), we argue that students should not only know **what** is the correct answer but also learn **why** they made mistakes. Therefore, in addition to providing the correct rationale generated by the teacher LLMs, we also present the student model's mistakes to the teacher LLMs and return the mistake-specific feedback. Furthermore, inspired by the multi-agent evaluation framework of Nan et al. (2023), we employ multiple LLMs as teachers and ask them the same question. Each teacher LLM's answer is reviewed by the other teachers, and only the responses that pass this peer-review process are included in the instruction training dataset. We believe this peer-review mechanism between teacher LLMs can significantly reduce biased or flawed rationales, leading to improved distillation performance. In summary, the contributions of our work are as follows:

1. The **F**ault-**A**ware Dist**I**llation via Peer-**R**eview (**FAIR**) approach is introduced to help student LM learn not only from the correct rationale but also from feedback on their own mistakes provided by teacher LLMs, which builds a comprehensive instruction tuning method aimed at enhancing the student LM's general reasoning abilities.

2. We design a simulated Peer-Review mechanism between teacher LLMs to filter out flawed rationales and improve the confidence of instruction tuning data.

3. Our work provides a comprehensive benchmark on the mathematical, commonsense, and logical reasoning tasks. Experiments and comparisons with other concurrent works demonstrate the effectiveness of our method in distilling the reasoning ability of teacher LLMs.

2 Related Work

LLM Reasoning Recent studies focus on provoking the thought processes of LLMs, validating their effectiveness in reasoning tasks (Wei et al., 2022b; Imani et al., 2023; Fu et al., 2023). Various techniques have been developed to enhance LLM reasoning abilities (Chu et al., 2023; Xu et al., 2024a; Chen et al., 2023). Chain-of-Thought (CoT) (Wei et al., 2022b) improves reasoning by prompting LLMs to generate intermediate natural language thought processes. Huang et al. (2022) demonstrates that LLMs can self-improve through self-training on majority voting data. Chung et al. (2024) showed that smaller LMs can acquire CoT skills by training on rationales. The work s1 (Muennighoff et al., 2025) proves the significance of high-quality CoT data on the reasoning test performance. In this paper, we further show that the CoT performance of smaller LMs can be improved through integrated instruction learning using CoT data selected by majority voting from LLMs.

Knowledge Distillation from LLMs Distilling knowledge from LLMs by fine-tuning smaller language models using high-quality data collected from LLMs has become a prominent research direction (Xu et al., 2023a; Li et al., 2024; Guo et al., 2025). This approach serves as an effective method for transferring the emergent abilities of black-box LLMs to smaller open-source models. However, while recent works (Ho et al., 2023; Shridhar et al., 2022; Guo et al., 2024) use LLM-generated reasoning rationales as supervisory signals, they often overlook providing student models with feedback on their mistakes when their answers are incorrect. To address this, we collect both the correct rationale and mistake-specific feedback (Jiang et al., 2023) for student models' wrong answers from LLMs, integrating them into instruction tuning to enhance the overall reasoning capabilities of the student models. Moreover, unlike previous studies that depend on a single teacher LLM (Chenglin et al., 2023; Zhu et al., 2024) or intermediate roles such as mentors (Lee et al., 2024) and teaching assistant (TA) (Zhou and Ai, 2024), we employ multiple LLMs (Tian et al., 2024; Sun et al., 2023) as teachers to increase the diversity of generated data. Finally, compared to peer-review methods in LLMs for evaluation (Ning et al., 2024; Chu et al., 2024), we design a simulated peer-review process to ensure high-quality instruction training data, thereby improving the distillation performance.

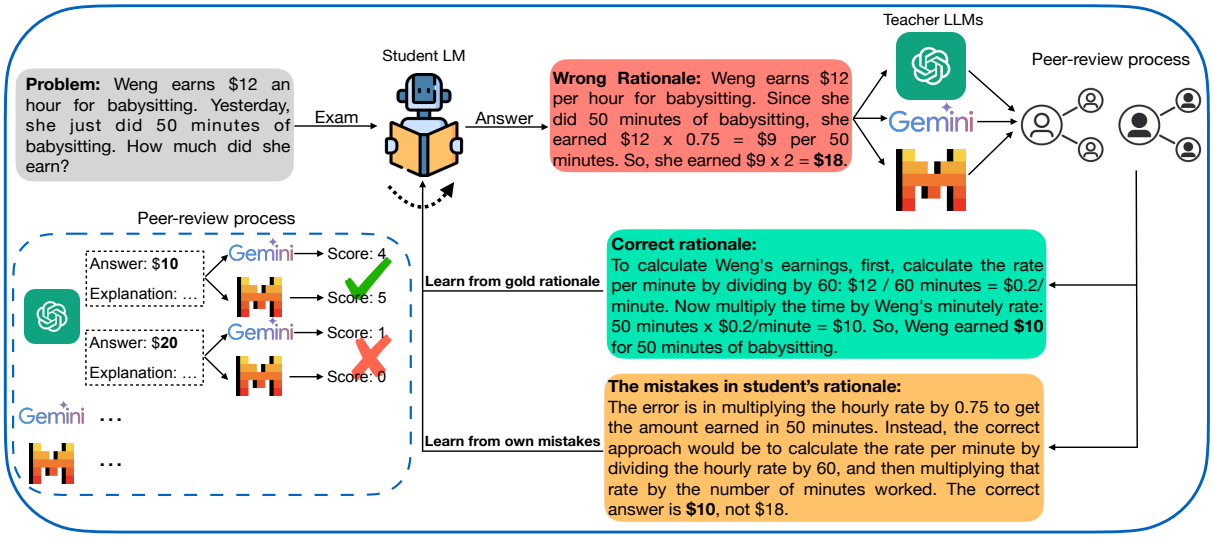

Figure 2: Overview of our **F**ault-**A**ware Dist**I**llation via Peer-**R**eview (**FAIR**) method. The specific structure of the peer-review process is explained in the left-bottom sub-figure.

3 Method

As illustrated in Figure 2, we introduce a **F**ault-**A**ware Dist**I**llation via Peer-**R**eview (**FAIR**) knowledge distillation method that empowers the student model to improve by learning from its own mistakes and the correct answers generated by multiple teacher models. Specifically, our instruction learning procedure involves four major steps: (1) The student LM takes an "exam" on the training set to identify mistakes that are incorrectly generated rationales. (2) We then craft various prompts that incorporate the question and the student's wrong rationale to prompt the teacher LLMs to generate correct answers and provide feedback on the student's errors respectively. (3) A simulated peer-review process is conducted among the teacher LLMs to produce highly confident instructional data. (4) Finally, the student model learns to reason through instruction learning based on the peer-reviewed correct answers and tailored corrections on its mistakes provided by the teacher LLMs.

3.1 Collecting Mistaks on Student Model

We aim to gather samples from reasoning benchmarks where the student model incorrectly answers questions. These samples will be used to create customized instructional data from the teacher models. To achieve this, the student model undergoes an "exam" on the training set D_{train} to assess its reasoning ability and collect the mistake set $D_{mistake}$, which are the samples containing incorrect rationales and answers. Specifically, given a dataset $D = \{x, y\}$, where x is the question and y is the gold answer, we propose to input the question x into the student model to generate the output $f(x) = [r', y']$. Here, the square brackets denote the concatenation of the student model's rationale r' and answer y', with the answer typically at the end of the output. Since the correct rationale r is often not provided in D_{train}, we follow Wang et al. (2023)'s work by considering r' as the wrong rationale if $y' \neq y$. Finally, the mistake set $D_{mistake}$ is collected as follows:

$$D_{mistake} = \{(x, r', y') \mid (x, y) \in D_{train},\ y' \neq y\} \quad (1)$$

where x is the question, r' is the wrong rationale, y and y' are correct and wrong final answer.

The collected mistake set $D_{mistake}$ highlights the student's reasoning weaknesses and will be utilized for the following purposes:

1) Providing the incorrectly answered questions for the teacher LLMs to generate correct rationales.

2) Using the student's incorrect rationales to prompt the teacher LLMs to identify errors and create customized mistakes feedback.

3.2 Inquiring Teacher LLMs with Student's Mistakes

We expect the teacher LLM to function as a reasoning instructor who can identify student's mistakes and provide tailored feedback, rather than merely an answer provider. Therefore, we query the teacher LLMs with the student's incorrectly answered questions, aiming for them to generate

Question: Weng earns $12…How much did she earn? Let's think step by step.
Teacher's rationale: …

Imagine you are a teacher, I will give you one student's incorrect answer to a question. You should point out the mistakes in the student's answer.
Question: Weng earns $12…How much did she earn? Let's think step by step.
Student's Answer: … So, she earned $9 x 2 = $18.
Hint: The correct answer should be 10.
Teacher's feedback: …

Figure 3: The prompt template P_{rt} (first) and P_{fb} (second) for asking teacher LLMs to generate rationale and mistakes feedback. The part colored in yellow is the teacher's output.

Imagine you are a reviewer, I will give you one submission about the rationale for a question. You should simulate the peer-review process by evaluating the rationale based on its correctness and soundness. Let's think step by step, but your final answer should only be one number, ranging from 1-5 (the higher the score is, the more possible you think the rationale is correct).
Question: Weng earns $12…How much did she earn? Let's think step by step.
Rationale**:** To calculate Weng's earnings…
Hint: The correct answer should be 10.
Teacher's score: …

Figure 4: The prompt template P_{pr} for asking teacher LLMs to perform peer-review process. The part colored in yellow is the teacher's output.

the correct rationale and identify specific errors in the student's mistakes. We believe that customized training data, which includes both "what" the correct answer is and "why" the mistakes were made, can effectively address the student's weaknesses. For prompt P_{fb} to gather feedback on the student's mistakes, we follow Zelikman et al. (2022) by adding a hint that explicitly provides the correct answer to the question, ensuring more accurate responses. The detailed prompt templates are shown in Figure 3. In detail, for each sample $(x, r', y') \in D_{mistake}$, we request each teacher $\mathcal{M}_T^k$ from the total of N teacher LLMs to generate its own feedback f_k, which will be collected as the mistakes feedback set $D_{feedback}$:

$$f_k = \mathcal{M}_T^k(P_{fd}(x, r', y))$$
$$D_{feedback} = \{(x, r', f_k) \mid (x, r', y') \in D_{mistake}, 1 \leq k \leq N\} \quad (2)$$

where $\mathcal{M}_T^k(x)$ represents the k-th teacher LLM's output when given x as the input. $P_{fb}(x)$ denotes the prompt template filled in with x to generate mistakes feedback.

3.3 Simulating Peer-Review Between Teacher Models

During our experiments, we observe that the rationales provided by teacher LLMs are not always accurate, even when the final answer matches the gold answer. This discrepancy is rare in mathematical tasks, where there is often a strict correlation between the correctness of the rationale and the final answer number due to the inherent nature of mathematics. However, for multiple-choice questions, such as those in the commonsense StrategyQA (Geva et al., 2021) (True or False) and logic LogiQA (Liu et al., 2020) (A, B, C, D) benchmarks, there are instances where a correct rationale may lead to an incorrect final choice, or a wrong rationale might result in a correct final choice. See Appendix B for more peer-review examples on different benchmarks.

To address this issue and avoid having teacher LLMs "**guess**" the correct answer without well-grounded reasoning steps, we propose a simulated peer-review process among teacher LLMs. Since most relevant datasets do not provide gold rationales, we assume that each LLM's rationale should be reviewed and scored by peer LLMs, which is inspired by the multi-agent evaluation framework of Nan et al. (2023). Only those rationales that pass this peer-review process with high confidence will be included in the final instructional tuning dataset. Figure 2 has explained the peer-review process. For the rationale generated by each teacher LLM, we incorporate it into the designed peer-review prompt P_{pr} shown in Figure 4 and request all other LLMs to score it. Specifically, assume we have N different teacher LLMs $\mathcal{M}_T^1, \mathcal{M}_T^2, ..., \mathcal{M}_T^N$. For the k-th teacher LLM $\mathcal{M}_T^k$, we obtain its generated rationale r_k by:

$$r_k = \mathcal{M}_T^k(P_{rt}(x)) \quad (3)$$

where $\mathcal{M}_T^k(x)$ represents the k-th teacher LLM's output when given x as the input. $P_{rt}(x)$ denotes the rationale prompt template filled in with x.

Subsequently, we ask each teacher except M_T^k to peer-review this rationale r_k and score it. The scores are collected to form the score set $Score(r_k)$ for rationale r_k. Only the rationale r_k with an average score $Avg(Score(r_k))$ exceeding the accep-

tance threshold Th will be included in the rationale set $D_{rationale}$:

$$Score(r_k) = \{\mathcal{M}_T^i(P_{pr}(x, r_k, y)) \mid 1 \leq i \leq N \; and \; i \neq k\}$$
$$D_{rationale} = \{(x, r_k) \mid if \; Avg(Score(r_k)) \geq Th, 1 \leq k \leq N\} \quad (4)$$

where $\mathcal{M}_T^i(x)$ represents the i-th teacher LLM's output with input x. $P_{pr}(x)$ denotes the peer-review prompt template filled in with x to generate score.

3.4 Instruction Tuning for Student Models

The reasoning ability of the student LM can be enhanced through instruction tuning (Wei et al., 2021), which incorporates both verified rationales and customized mistake corrections provided by the teacher models. See Appendix C for explicit instruction tuning templates on different benchmarks.

Learning from Teacher's Rationales The rationales generated by the teacher LLMs are specifically tailored to address the student's weaknesses, identified through the student's previous exam. According to Equation 4, these collected rationales are combined into the set $D_{rationale}$ as the correct rationales, which are then used to fine-tune the student LM. For the instruction tuning process, we aim for the student model, when given the question x as the instruction, to produce an answer that closely aligns with the corresponding rationale r in $D_{rationale}$. The loss function for learning from the teacher's rationale is defined as follows:

$$\mathcal{L}_{\text{rationale}} = \mathbb{CE}(\mathcal{M}_S(x), r), \; for \; r \in D_{\text{rationale}} \quad (5)$$

where $\mathbb{CE}$ denotes the Cross-Entropy function, and $\mathcal{M}_S(x)$ represents the student LM's output when given x as the input.

Learning from Student's Mistakes In addition to learning from correct rationales, we propose that the student model should also learn from its own mistakes, simulating the typical human learning process. This approach helps the student not only grasp the correct answers but also understand the reasons behind the errors. To facilitate this, we constructed the feedback set $D_{feedback}$, based on Equation 2, which provides feedback on the student's mistakes. Through this process, we expect the student LM to learn the teacher's reasoning capabilities and generate outputs that closely align with the teacher's feedback f when given instructions to identify its own mistakes. Finally, the loss function for learning from mistakes feedback is defined as follows:

$$\mathcal{L}_{\text{feedback}} = \mathbb{CE}\left(\mathcal{M}_S(x \oplus r'), f\right), \; for \; f \in D_{feedback} \quad (6)$$

where $\mathbb{CE}$ denotes the Cross-Entropy function, and $\oplus$ represent the string concatenation. $\mathcal{M}_S(x \oplus r')$ represents the student LM's output when given $x \oplus r'$ as the input.

Joint Learning The final optimization process integrates learning from both correct answers and the teachers' customized mistakes feedback. Therefore, the instruction learning losses from Equation 5 and Equation 6 are combined as follows:

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{\text{feedback}} + (1 - \alpha) \cdot \mathcal{L}_{\text{rationale}} \quad (7)$$

where α controls the impact of learning from mistakes, balancing the two learning objectives.

4 Experiments

4.1 Datasets

We focus on evaluating reasoning abilities with various datasets, including mathematical reasoning with GSM8K (Cobbe et al., 2021) and SVAMP (Patel et al., 2021), commonsense reasoning with StrategyQA (Geva et al., 2021), and logical reasoning with LogiQA (Liu et al., 2020). All datasets were downloaded from Huggingface, utilizing the standard train/test set split. Datasets statistics are shown in Appendix A.1.

4.2 Baselines

To demonstrate the effectiveness of our method, we include the following baselines: (1) the teacher LLMs and student LMs without fine-tuning, to highlight the impact of distilling reasoning abilities from the teachers; (2) sophisticated distillation methods applied to smaller models: CodeT5-Large (Zhu et al., 2024), Qwen2-1.5B (Adarsh et al., 2025), and GPT-J (Wang et al., 2023); (3) related works that utilize Llama-series: Llama-7B (Li et al., 2024), Llama2-7B (Guo et al., 2024; Mitra et al., 2023), and Llama3.1-8B (Hicham Badri, 2025); and (4) distillation approach on the larger model T5-XXL (Magister et al., 2022).

4.3 Implementation Details

Models We selected GPT-3.5-Turbo[1], Gemini-1.0-Pro (Team et al., 2023), and Mixtral-8x7B-Instruct-v0.1 (Jiang et al., 2024) as the teacher LLMs. The selection motivations include the considerations of the expense and accessibility of the LLMs and their proved powerful NLP capabilities. Among

[1] https://platform.openai.com/docs/models/gpt-3-5-turbo

Method	# Params	Distillation Teachers	Mathematical		Commonsense	Logical
			GSM8K	SVAMP	StrategyQA	LogiQA
Teacher LLMs						
GPT-3.5-Turbo	175B	-	78.01*	82.30*	70.92*	40.55*
Gemini-1.0-Pro	-	-	76.42*	81.10*	67.03*	39.94
Mixtral-8x7B-Instruct-v0.1	46.7B	-	74.40*	81.60*	72.83*	34.19*
Student LM Baselines						
CodeT5-Large+PaD (Zhu et al., 2024)	770M	GPT-3.5-Turbo	44.90*	51.00*	-	-
Qwen2-1.5B+SIKeD (Adarsh et al., 2025)	1.5B	Llama3-70B	64.97*	75.40*	-	-
GPT-J+Self-Reflection (Wang et al., 2023)	6B	ChatGPT	33.10*	55.00*	65.90*	-
Llama-7B+NCE (Li et al., 2024)	7B	GPT-3.5-Turbo, GPT-4	41.93*	51.50*	-	-
Llama2-7B+ReversalMath (Guo et al., 2024)	7B	GPT-4	52.10*	59.20*	-	-
ORCA2-7B (Mitra et al., 2023)	7B	ChatGPT, GPT-4	47.23*	-	-	35.02*
Llama3.1-8B+ReDistill (Hicham Badri, 2025)	8B	DeepSeek-R1	75.66*	82.00	-	-
T5-XXL+CoT (Magister et al., 2022)	11B	PaLM, GPT-3	21.99*	-	63.77*	-
Peer-Reviewed Distillation (Ours)						
Llama2-7B-chat (Touvron et al., 2023)	7B	-	15.62	39.67	47.02	18.74
+Teacher-Mixtral	7B	Mixtral-8x7B-Instruct	22.67	47.33	62.70	32.10
+Teacher-Gemini	7B	Gemini-1.0-Pro	26.84	49.33	57.93	32.72
+Teacher-GPT	7B	GPT-3.5-Turbo	30.71	51.67	60.12	31.04
+Teacher-Multiple, w/o Peer-Review	7B	Multiple	29.65	52.67	56.62	29.65
+Teacher-Multiple	7B	Multiple	**36.24**	**59.50**	**67.69**	**36.25**
Qwen2.5-1.5B-Instruct (Yang et al., 2024)	1.5B	-	64.44	77.00	53.86	19.97
+Teacher-Mixtral	1.5B	Mixtral-8x7B-Instruct	65.81	77.67	63.32	32.10
+Teacher-Gemini	1.5B	Gemini-1.0-Pro	66.26	78.67	60.41	33.95
+Teacher-GPT	1.5B	GPT-3.5-Turbo	68.01	79.33	62.45	34.25
+Teacher-Multiple, w/o Peer-Review	1.5B	Multiple	67.48	77.67	61.43	33.03
+Teacher-Multiple	1.5B	Multiple	**72.48**	**81.00**	**68.12**	**38.71**
Llama3.1-8B-Instruct (Dubey et al., 2024)	8B	-	74.00	81.67	63.03	36.56
+Teacher-Mixtral	8B	Mixtral-8x7B-Instruct	74.83	82.00	71.62	37.02
+Teacher-Gemini	8B	Gemini-1.0-Pro	76.42	82.33	66.96	39.94
+Teacher-GPT	8B	GPT-3.5-Turbo	77.94	83.00	70.16	40.86
+Teacher-Multiple, w/o Peer-Review	8B	Multiple	76.57	82.67	70.89	38.40
+Teacher-Multiple	8B	Multiple	**79.30**	**84.33**	**73.07**	**43.16**

Table 1: Accuracy (%) across various reasoning tasks with different distillation methods. * denotes the results are from the original paper or official document. "Teacher-x" indicates the specific teacher LLM used in the distillation experiment. The best performance among different student LMs in each benchmark is marked in **bold**.

the three student models, we choose Llama2-7B-chat (Touvron et al., 2023) as the backbone for its active community to compare performance, and Qwen2.5-1.5B-Instruct (Yang et al., 2024) as well as Llama3.1-8B Instruct (Dubey et al., 2024) to test the generalizability of FAIR method. The threshold in Equation 4 was set to $Th = 4$ for high confident rationales. The parameter α in Equation 7 was set to $\alpha = 0.5$ to balance the impact of learning from mistakes. For data inference from teacher LLMs, we collect samples that have at least one peer-reviewed rationale and one feedback. During the training, we randomly select one feedback and one rationale for each sample. All evaluation results are based on the zero-shot test set. Primary experiments were conducted on four Nvidia A100-80GB GPUs. More implementation details are in Appendix A.

4.4 Main Results

Main results are shown in Table 1.

Advantage of Distillation The inference results of student LM Llama2-7B show significant improvement after applying knowledge distillation. Although it still has a noticeable gap between the distilled Llama2-7B and teacher LLMs in mathematical reasoning after distillation, the fine-tuned Llama2-7B outperforms the weakest teacher LLM in commonsense and logical tasks. As more updated and powerful student LMs, Qwen2.5-1.5B and Llama3.1-8B show steady improvements after distillation. Notably, the multiple-teacher distillation results on Llama3.1-8B even surpass all teacher LLMs. Considering that we only use the failed cases set as shown in Table 2, it demonstrates that FAIR method effectively integrates LLMs to enhance the reasoning abilities of student models.

Comparison with Baselines Compared to distillation methods on smaller models such as CodeT5, Qwen2-1.5B , and GPT-J, FAIR on Qwen2.5-1.5B consistently achieves superior performance on the available mathematical and commonsense tasks. Compared with other works based on Llama-series

	#Wrong / #Train			
Model	GSM8K	SVAMP	StrategyQA	LogiQA
Qwen2.5-1.5B	1705/7473	136/700	671/1603	5450/7376
Llama2-7B	6236/7473	387/700	825/1603	6159/7376
Llama3.1-8B	1422/7473	132/700	554/1603	4433/7376

Table 2: Exam results on original student models. The wrongly answered samples will be collected for generating the teacher responses and distillation training set.

models, on the GSM8K benchmark, our performance on Llama2-7B (36.24%) lags behind Llama-7B+NCE (41.93%) and ReversalMath (52.10%), likely because these models were exclusively fine-tuned on mathematical tasks, with GSM8K being a key and difficult benchmark in this domain. The other trained mathematical datasets may improve student LM's overall mathematical reasoning capability. In addition, we utilize only the failed cases set, which is significantly smaller compared to the training data in other studies. Nevertheless, our approach still yields better performance compared to ReversalMath on another easier and smaller mathematical benchmark, SVAMP (59.50%>59.20%). Additionally, our results on LogiQA (36.25%) also exceed the ORCA2-7B (35.02%). Finally, distillation results on Llama3.1-8B-Instruct surpass the same Llama3.1-8B-Instruct+ReDistill and the larger T5-XXL+CoT on mathematical and commonsense tasks.

5 Analysis

5.1 Analysis about Peer-Review Process

To assess the importance of the peer-review process further, we compared the evaluation results with and without peer-review, as shown in Table 1. When peer-review is absent, the average test accuracy across all benchmarks decreases by 7.84%, 5.18%, and 2.83% for Llama2-7B, Qwen2.5-1.5B, and Llama3.1-8B, respectively. This reinforces that noisy answers generated by multiple teachers, which could potentially confuse the student model during instruction tuning, can be effectively filtered through peer review, ultimately enhancing the student model's performance. In addition, for our backbone Llama2-7B, the experiments without peer-review even fall behind the best single teacher-GPT distillation outcomes on GSM8K (29.65%<30.71%). This pattern is particularly pronounced in commonsense and logical reasoning tasks. These findings align with our assumption that peer-review may have a smaller impact on mathematical reasoning tasks, where the rationale and final result are highly correlated, but significantly improves the quality of instruction data in commonsense and logical reasoning tasks. More results based on peer-review between only two teacher LLMs are displayed in Appendix D.

5.2 Quality of Automated Peer-Review

To further evaluate the reliability of our automated peer-review process, we conducted a manual analysis to assess whether the teachers' reasoning process genuinely supports their answers. This is important because an answer may sometimes be correct by chance despite flawed reasoning. First, we randomly selected 100 samples from $D_{mistake}$ of the LogiQA dataset and collected the original "correct" responses, whenever a teacher model's predicted final answers matched the gold multiple-choice answers. We then manually examined these responses and removed those "guessed" correct answers with flawed rationales. Finally, we compared our gold-standard, human-annotated reasoning with those produced by the automated peer-review process. Table 3 revealed that the peer-review process achieved an average accuracy of **90.35%** when compared to human annotations, demonstrating its high reliability.

Model	Original	PR	Human
GPT-3.5-Turbo	40	38	34
Gemini-1.0-Pro	38	34	30
Mixtral-8x7B-Instruct	35	30	28

Table 3: Comparison of the number of responses verified by original model predictions, peer-review (PR), and human annotations for random 100 LogiQA samples.

5.3 Abalation of Learning from Mistakes

As a key component of our FAIR method, we initially set the proportion of learning from mistakes to 0.5 in previous experiments for simplicity. To explore the influence of balancing learning from rationales and learning from mistakes, we adjust the value of α in Equation 7. Specifically, α was varied from [0, 0.25, 0.5, 0.75, 1], and experiments were conducted on all benchmarks for 5 epochs on Llama2-7B-chat, while keeping other parameters constant. Figure 5 visualizes how learning from mistakes affects instruction-tuning. Our findings support the hypothesis that learning from mistakes positively impacts instruction tuning. However, the relationship is not uniformly positive across all α values on the four benchmarks.

For GSM8K and LogiQA, the benefits of learning from mistakes increase when $\alpha < 0.25$, but start to decrease when α exceeds 0.25. Conversely, for StrategyQA and SVAMP, the advantages of learning from mistakes consistently grow and reach their peak when $\alpha = 0.75$. These results suggest that placing too much emphasis on learning from mistakes (i.e., a higher α value) can lead to instability. Consequently, it is important to evaluate and optimize α value for different tasks to effectively balance the learning of "what" (correct answers) and "why" (own mistakes) during training.

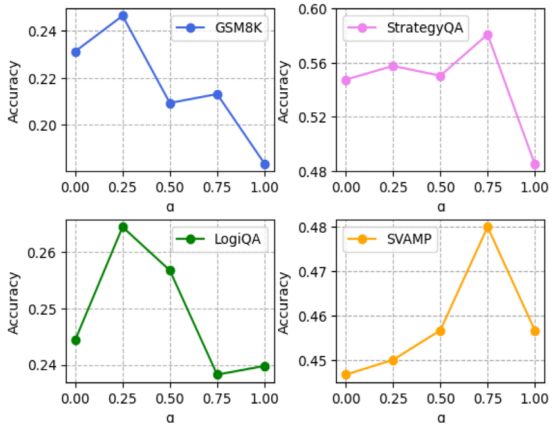

Figure 5: The effect of α of the tuning performance on Llama2-7B-chat. α=0 indicates the absence of learning from mistakes.

5.4 Effectiveness of Multiple Teachers

As shown in Table 1, our multiple-teacher distillation with peer-review method on Llama2-7B improves the average accuracy by 5.48% across four benchmarks compared to the single teacher distillation method with the highest accuracy. Although the performance gains on Qwen2.5-1.5B and Llama3.1-8B are slightly reduced, this is likely due to the strong baseline capabilities of the original student models, which are already competitive against teacher LLMs, and the limited size of the generated training set.

To ensure that all teacher LLMs contribute meaningfully to the final performance and prevent free-riding, Table 4 reports the number of responses utilized in the final multiple-teacher training tasks. They are generated by different LLMs and verified through the peer-review process. This comparison correlates with the distinct capabilities of each teacher model and underscores their collective contribution to enhancing the student model's performance after fine-tuning. Detailed comparisons of the student LM's output before and after distillation are provided in Appendix E.

Dataset	Qwen2.5-1.5B	Llama2-7B	Llama3.1-8B
GSM8K	595:486:589	2110:1801:2256	472:420:506
SVAMP	34:28:63	161:87:117	31:27:64
StrategyQA	333:143:123	396:192:160	293:106:78
LogiQA	1731:1884:1545	1706:2085:2049	1478:1403:1286

Table 4: The number of responses from various teacher LLMs used in the final multiple-teacher distillation process. The values represent the number of data points from Mixtral/Gemini/GPT respectively. This demonstrates that all teacher LLMs contribute significantly.

5.5 Assessment of Computational Overhead

To address concerns about the additional computational overhead introduced by FAIR, we evaluate the resources consumed during our experiments. Table 5 provides a comparison of the average number of tokens consumed for each sample with and without the peer-review. The selected teacher models are all entry-level LLMs that do not require subscriptions or high costs, ensuring accessibility for researchers with limited resources. Given the substantial improvement in the student model's performance and the fact that distillation is a one-time investment, the additional cost is highly justified. Moreover, the distilled model can even outperform certain teacher LLMs on specific benchmarks while maintaining significantly lower inference costs.

LLMs	Standard	PR	ΔCost ↑
GPT-3.5-Turbo	100.32	200.18	\$0.00005
Gemini-1.0-Pro	60.94	220.82	\$0.00008
Mixtral-8x7B-Instruct	131.12	214.45	\$0.00002

Table 5: The average number of tokens consumed for each sample with and without the peer-review (PR).

6 Conclusion

In this work, we introduce the Fault-Aware Distillation via Peer-Review (FAIR) approach. We implement a simulated peer-review process between multiple teacher LLMs to gather reliable outputs, which refines the quality of instruction tuning dataset. Additionally, we develop an integrated instruction tuning method that allows the student LM to learn from both the correct rationale and mistakes feedback. Comprehensive results on diverse reasoning tasks validate our efficient method for unlocking the reasoning potential of smaller open-source LMs through distillation, even with black-box LLMs and without dataset-provided rationales. We hope that our findings will encourage further investigations into reasoning distillation.

Limitations

Although our method demonstrates effectiveness in the reasoning ability distillation from teacher models to the student model, this technique has several limitations. First, our experiments primarily rely on GPT-3.5-Turbo, Gemini-1.0-Pro, and Mixtral-8x7B-Instruct-v0.1 as teacher LLMs due to considerations of availability and cost. The results in Table 1 suggest that as student models improve, the bottleneck in performance may shift to the capabilities of the teacher LLMs, highlighting the need for more advanced teacher models to further enhance student performance. Future research could benefit from using more powerful models like DeepSeek-R1, OpenAI-o3, and Claude-3 Opus. Secondly, future work could include more challenging benchmarks across different reasoning fields, such as FrontierMath (Glazer et al., 2024) and Humanity's Last Exam (Phan et al., 2025). Thirdly, due to time and cost constraints, our method does not collect the student LM's incorrect rationales and updates the instruction dataset after each epoch. The potential benefits of continuously incorporating fresh data throughout online training remain unexplored. Moreover, further research can regard teacher LLMs as agents, incorporating more sophisticated pipelines such as negotiation and decision-making during the peer-review process to enhance reliability. Lastly, we employ the default cross-entropy loss function for instruction tuning. It would be worthwhile to explore more sophisticated methods, such as the Group Relative Policy Optimization (GRPO) Reinforcement Learning method used in DeepSeek-R1, and to integrate additional techniques into the joint learning approach.

Ethics Statement

The study offers a novel structure for knowledge distillation of the reasoning ability from LLMs to smaller LM, which could contribute to increased transparency and availability in AI systems. It underscores the fact that proprietary LLMs dominate reasoning tasks and weaken smaller open-source LMs. However, parts of the annotated data in this paper are collected from close-source GPT provided by OpenAI, and Gemini supplied by Google. The explainability and transparency of close-source models may raise risks for annotated data and decrease the trustworthiness.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Shivam Adarsh, Kumar Shridhar, Caglar Gulcehre, Nicholas Monath, and Mrinmaya Sachan. 2025. SIKed: Self-guided iterative knowledge distillation for mathematical reasoning.

Hongzhan Chen, Siyue Wu, Xiaojun Quan, Rui Wang, Ming Yan, and Ji Zhang. 2023. Mcc-kd: Multi-cot consistent knowledge distillation. *arXiv preprint arXiv:2310.14747*.

Li Chenglin, Chen Qianglong, Wang Caiyu, and Zhang Yin. 2023. Mixed distillation helps smaller language model better reasoning. *arXiv preprint arXiv:2312.10730*.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Tao He, Haotian Wang, Weihua Peng, Ming Liu, Bing Qin, and Ting Liu. 2023. A survey of chain of thought reasoning: Advances, frontiers and future. *arXiv preprint arXiv:2309.15402*.

Zhumin Chu, Qingyao Ai, Yiteng Tu, Haitao Li, and Yiqun Liu. 2024. Pre: A peer review based large language model evaluator. *arXiv preprint arXiv:2401.15641*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

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*.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Xiaojing Fan and Chunliang Tao. 2024. Towards resilient and efficient llms: A comparative study of efficiency, performance, and adversarial robustness. *arXiv preprint arXiv:2408.04585*.

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*, pages 10421–10430. PMLR.

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.

Elliot Glazer, Ege Erdil, Tamay Besiroglu, Diego Chicharro, Evan Chen, Alex Gunning, Caroline Falkman Olsson, Jean-Stanislas Denain, Anson Ho, Emily de Oliveira Santos, et al. 2024. Frontiermath: A benchmark for evaluating advanced mathematical reasoning in ai. *arXiv preprint arXiv:2411.04872*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Pei Guo, Wangjie You, Juntao Li, Yan Bowen, and Min Zhang. 2024. Exploring reversal mathematical reasoning ability for large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 13671–13685.

Sai Gurrapu, Ajay Kulkarni, Lifu Huang, Ismini Lourentzou, and Feras A Batarseh. 2023. Rationalization for explainable nlp: a survey. *Frontiers in Artificial Intelligence*, 6:1225093.

Appu Shaji Hicham Badri. 2025. Re-distilling smaller deepseek r1 models for better performance.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14852–14882.

Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2022. Large language models can self-improve. *arXiv preprint arXiv:2210.11610*.

Shima Imani, Liang Du, and Harsh Shrivastava. 2023. Mathprompter: Mathematical reasoning using large language models. *arXiv preprint arXiv:2303.05398*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Yuxin Jiang, Chunkit Chan, Mingyang Chen, and Wei Wang. 2023. Lion: Adversarial distillation of proprietary large language models. *arXiv preprint arXiv:2305.12870*.

Kathryn E Konold, Susan P Miller, and Kyle B Konold. 2004. Using teacher feedback to enhance student learning. *Teaching Exceptional Children*, 36(6):64–69.

Hojae Lee, Junho Kim, and SangKeun Lee. 2024. Mentor-kd: Making small language models better multi-step reasoners. *arXiv preprint arXiv:2410.09037*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. 2022. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857.

Yiwei Li, Peiwen Yuan, Shaoxiong Feng, Boyuan Pan, Bin Sun, Xinglin Wang, Heda Wang, and Kan Li. 2024. Turning dust into gold: Distilling complex reasoning capabilities from llms by leveraging negative data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18591–18599.

Hanmeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. 2023. Evaluating the logical reasoning ability of chatgpt and gpt-4. *arXiv preprint arXiv:2304.03439*.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2022. Teaching small language models to reason. *arXiv preprint arXiv:2212.08410*.

Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andres Codas, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, et al. 2023. Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv:2311.11045*.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*.

Linyong Nan, Ellen Zhang, Weijin Zou, Yilun Zhao, Wenfei Zhou, and Arman Cohan. 2023. On evaluating the integration of reasoning and action in llm agents with database question answering. *arXiv preprint arXiv:2311.09721*.

Haowei Ni, Shuchen Meng, Xupeng Chen, Ziqing Zhao, Andi Chen, Panfeng Li, Shiyao Zhang, Qifu Yin, Yuanqing Wang, and Yuxi Chan. 2024. Harnessing earnings reports for stock predictions: A qlora-enhanced llm approach. *arXiv preprint arXiv:2408.06634*.

Kun-Peng Ning, Shuo Yang, Yu-Yang Liu, Jia-Yu Yao, Zhen-Hui Liu, Yu Wang, Ming Pang, and Li Yuan. 2024. Peer-review-in-llms: Automatic evaluation method for llms in open-environment. *arXiv preprint arXiv:2402.01830*.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? *arXiv preprint arXiv:2103.07191*.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Sean Shi, Michael Choi, Anish Agrawal, Arnav Chopra, et al. 2025. Humanity's last exam. *arXiv preprint arXiv:2501.14249*.

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. 2022. Distilling reasoning capabilities into smaller language models. *arXiv preprint arXiv:2212.00193*.

Qiushi Sun, Zhangyue Yin, Xiang Li, Zhiyong Wu, Xipeng Qiu, and Lingpeng Kong. 2023. Corex: Pushing the boundaries of complex reasoning through multi-model collaboration. *arXiv preprint arXiv:2310.00280*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Yijun Tian, Yikun Han, Xiusi Chen, Wei Wang, and Nitesh V Chawla. 2024. Tinyllm: Learning a small student from multiple large language models. *arXiv preprint arXiv:2402.04616*.

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*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Zhaoyang Wang, Shaohan Huang, Yuxuan Liu, Jiahai Wang, Minghui Song, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, et al. 2023. Democratizing reasoning ability: Tailored learning from large language model. *arXiv preprint arXiv:2310.13332*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

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

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

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023a. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. *arXiv preprint arXiv:2304.01196*.

Fangzhi Xu, Qika Lin, Jiawei Han, Tianzhe Zhao, Jun Liu, and Erik Cambria. 2023b. Are large language models really good logical reasoners? a comprehensive evaluation from deductive, inductive and abductive views. *arXiv preprint arXiv:2306.09841*.

Han Xu, Jingyang Ye, Yutong Li, and Haipeng Chen. 2024a. Can speculative sampling accelerate react without compromising reasoning quality? In *The Second Tiny Papers Track at ICLR 2024*.

Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. 2024b. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. STar: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

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.

Yuhang Zhou and Wei Ai. 2024. Teaching-assistant-in-the-loop: Improving knowledge distillation from imperfect teacher models in low-budget scenarios. *arXiv preprint arXiv:2406.05322*.

Xuekai Zhu, Biqing Qi, Kaiyan Zhang, Xinwei Long, Zhouhan Lin, and Bowen Zhou. 2024. 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.

A Experimental Setup Details

A.1 Datasets Statistics

We download datasets GSM8K, SVAMP, StrategyQA, and LogiQA from Huggingface. All datasets are split according to the official original split ratio. Table 6 shows the dataset statistics.

Dataset	Type	#Train	#Test
GSM8K	Mathematical	7473	1319
SVAMP	Mathematical	700	300
StrategyQA	Commonsense	1603	687
LogiQA	Logical	7376	651

Table 6: Dataset statistics.

A.2 Teacher LLMs Parameters

Table 7 shows the unified parameters setting for GPT-3.5-Turbo, Gemini-1.0-Pro, and Mixtral-8x7B-Instruct-v0.1 LLMs to generate answers for the student LM. GPT-3.5-Turbo and Gemini-1.0-Pro are required by their official APIs. Mixtral-8x7B-Instruct-v0.1 is required by the API hosted on Deepinfra: `https://deepinfra.com/mistralai/Mixtral-8x7B-Instruct-v0.1`.

Parameter	Value
Temperature	0.8
Max tokens	512
Top p	1
Presence penalty	0
Frequency penalty	0

Table 7: Teacher LLMs parameter settings.

A.3 Student LM Parameters

Experiments are performed with the Huggingface Trainer framework and Flash Attention (Dao et al., 2022). We use four Nvidia A100-80GB GPUs with FP16 for training and evaluation. The inference parameter settings across all datasets are shown in Table 8. The training hyperparameter settings across all datasets are shown in Table 9.

Parameter	Value
Temperature	0.3
Max new tokens	512
Top p	0.9
Top k	50
Do sample	True

Table 8: Student LM inference parameter settings.

Hyperparameter	Value
Epoch	10
Batch size	8
Learning rate	1e-5
Warmup ratio	0.03
Max seq length	512
Optimizer	AdamW
Gradient accumulation steps	2
Max grad norm	0.3

Table 9: Student LM training hyperparameter settings.

B Peer-Review Examples

Table 12 provides detailed examples of the peer-review process on GSM8K and StrategyQA. It highlights instances where the causality between the teacher LLM's rationale and the final answer may be insufficient, and demonstrates how our peer-review mechanism effectively identifies the most confident rationales.

C Instruction Tuning Templates

- Instruction tuning templates for learning from mistakes.
  - For all benchmarks:
    "### Instruction: Imagine you are a teacher, I will give you one student's incorrect answer to a question. You should point out the mistakes in the student's answer.
    ### Input: {}
    ### Response: {}"
- Instruction tuning templates for learning from rationale.
  - For benchmarks GSM8K and SVAMP:
    "### Instruction: Answer the following question. Let's think step by step.
    ### Input: {}
    ### Response: {}"
  - For benchmark strategyQA:
    "### Instruction: Answer the following question. Let's think step by step. First, you should answer "true" or "false". Then, you should explain how you draw this conclusion.
    ### Input: {}
    ### Response: {}"
  - For benchmark logiQA:
    "### Instruction: Answer the following question based on the given context, query, and options. Let's think step by step.
    ### Input: {}
    ### Response: {}"

D The Performance of Peer-Review between Two Teacher LLMs

To explore the cooperation between teacher LLMs further, we conduct experiments on the same student model Llama2-7B-chat based on combinations of two different teacher LLMs. The results are shown in Table 10. It is found that the performance improvement still correlates to the teacher LLMs' abilities on benchmarks. However, the performance of combinations for two teacher LLMs lags behind the three-teacher distillation, which proves the necessity of choosing three teacher LLMs as reviewers.

	Mathematical		**Commonsense**	**Logical**
Student LMs	GSM8K	SVAMP	StrategyQA	LogiQA
Llama2-7B-chat	15.62	39.67	47.02	18.74
+Mixtral, Gemini	27.37	49.67	63.32	34.56
+Mixtral, GPT	29.11	52.67	64.92	33.95
+Gemini, GPT	32.37	55.00	62.01	34.25
+Multiple	**36.24**	**59.50**	**67.69**	**36.25**

Table 10: Results of peer-review between two teacher LLMs.

E Case Study of Distillation Impact on Student LM's Output

Table 13 provides the comparisons of student LM's behavior difference before and after the instruction tuning across four benchmarks.

F The Performance of Out-of-Distribution (OOD) Scenarios

To evaluate the generalization abilities of different methods on out-of-distribution (OOD) data, we conducted experiments using one mathematical reasoning dataset as the training set and another dataset as the test set. Table 11 highlights the performance of the FAIR method on Llama2-7B-chat in OOD scenarios.

The results indicate a decrease in performance improvement compared to the original in-distribution scenarios in Table 1. Specifically, the accuracy gains on GSM8K were smaller than those on SVAMP, likely due to the greater complexity of GSM8K. Despite this, our multiple-teacher distillation approach consistently outperforms all single-teacher methods under OOD conditions, demonstrating its robustness and generalizability.

	Mathematical	
Student LMs	GSM8K	SVAMP
Llama2-7B-chat+Mixtral	8.72	36.67
Llama2-7B-chat+Gemini	10.54	46.00
Llama2-7B-chat+GPT	12.36	47.67
Llama2-7B-chat+Multiple	**13.43**	**55.00**

Table 11: The performance of FAIR on Llama2-7B-chat in out-of-distribution (OOD) scenarios. Specifically, we conducted experiments by training on SVAMP and testing on GSM8K, as well as training on GSM8K and testing on SVAMP.

GSM8K Example	
Question	"Josh is saving up for a box of cookies. To raise the money, he is going to make bracelets and sell them. It costs $1 for supplies for each bracelet and he sells each one for $1.5. If he makes 12 bracelets and after buying the cookies still has $3, how much did the box of cookies cost?"
Gold answer	"He makes $.5 profit on each bracelet because 1.5 - 1 = .5 He earns $6 because 12 x .5 = 6 The cookies cost $3 because 6 - 3 = 3 #### **3**"
GPT's answer	"Let's first calculate Josh's profit per bracelet: Profit per bracelet = Selling price - Cost price Profit per bracelet = $1.5 - $1 Profit per bracelet = $0.5 Now, let's calculate Josh's total profit from making and selling 12 bracelets: Total profit = Profit per bracelet * Number of bracelets Total profit = $0.5 * 12 Total profit = $6 We know that after buying the cookies, Josh still has $3 left. This means he spent $6 - $3 = $3 on the box of cookies. Therefore, the box of cookies cost **$3**." **Gemini's score: 5, Mixtral's score: 5**
Gemini's answer	"**15**" **GPT's score: 1, Mixtral's score: 2**
Mixtral's answer	"The box of cookies cost $13. Here's the reasoning: 1. Josh spends $1 on supplies for each bracelet, and he makes 12 bracelets. So, the total cost of supplies is 12 × $1 = $12. 2. He sells each bracelet for $1.5, and he makes 12 bracelets. So, the total revenue from selling the bracelets is 12 × $1.5 = $18. 3. After buying the cookies, he still has $3 left. This means that his total revenue ($18) minus the cost of the cookies ($X) equals the amount of money he has left ($3). In other words, $18 - $X = $3. 4. Solving for the cost of the cookies ($X), we get $X = $18 - $3 = $15. However, the problem states that after buying the cookies and still having $3, the cost of the cookies should be $13, not $15. **This might be a mistake in the question itself**." **GPT's score: 2, Gemini's score: 2**
Peer-reviewed answers	**GPT's answer**

StrategyQA Example	
Question	"Would 2020 Toyota Supra lag behind at a Nascar rally?"
Gold answer	"**True**. The 2020 Toyota Supra has a top speed of 155 MPH. Nascar stock cars routinely exceed 200 MPH."
GPT's answer	"**False**. Explanation: 1. The 2020 Toyota Supra is not a car typically used in NASCAR racing. NASCAR races primarily feature specially designed stock cars that meet specific performance and safety regulations. 2. The 2020 Toyota Supra is a production sports car designed for road use, not for NASCAR track racing. While it may have impressive performance capabilities on the road, it is not optimized for NASCAR oval track racing conditions. 3. NASCAR races involve high speeds, close competition, and specific handling characteristics that are different from what the 2020 Toyota Supra is designed for. **The Supra would likely struggle to keep up with dedicated NASCAR race cars** in a NASCAR rally due to these differences in design and performance characteristics." **Gemini's score: 4, Mixtral's score: 2**
Gemini's answer	"**True**. The 2020 Toyota Supra is a sports car, while NASCAR vehicles are purpose-built race cars. The Supra would not be competitive in a NASCAR race because it is not designed for that type of racing. NASCAR vehicles are much more powerful and have a much better suspension system than the Supra. The Supra would also be at a disadvantage because it is not as aerodynamic as a NASCAR vehicle." **GPT's score: 4, Mixtral's score: 5**
Mixtral's answer	"**True**. NASCAR race cars are designed for high-speed oval track racing, with powerful V8 engines, aerodynamic bodies, and specialized suspension systems. The Supra, on the other hand, is a street-legal sports car, with a turbocharged inline-six engine and a more versatile suspension setup. If the Supra rebuilds the engine and improve other systems, **it is likely suited for the unique demands of NASCAR racing and keep up with NASCAR vehicles.**" **GPT's score: 2, Gemini's score: 3**
Peer-reviewed answers	**Gemini's answer**

Table 12: Detailed examples of peer-review process on different benchmarks

GSM8K Example	
Question	Weng earns $12 an hour for babysitting. Yesterday, she just did 50 minutes of babysitting. How much did she earn? **Gold answer:** $10
Student's output before tuning	Answer: $18. Explanation:Weng earns $12 per hour for babysitting. Since she did 50 minutes of babysitting, she earned $12 x 0.75 = $9 per 50 minutes. So, she earned $9 x 2 = $18.
Teachers' correct rationale	Weng earned $10 for 50 minutes of babysitting. Here's the logic behind this: 1. We know that Weng earns $12 per hour. 2. To find out how much she earns in 50 minutes, we first need to convert the time into hours. 3. We know that 1 hour equals 60 minutes, so 50 minutes is equal to 50/60 hours, which simplifies to 0.833 hours. 4. Now, we can calculate how much Weng earns for 0.833 hours by multiplying her hourly rate by the number of hours: $12 x 0.833 = $10. So, Weng earned $10 for 50 minutes of babysitting.
Teachers' mistakes feedback	The student's answer contains a mistake in calculating how much Weng earned for 50 minutes of babysitting. The error is in multiplying the hourly rate by 0.75 to get the amount earned in 50 minutes. Instead, the correct approach would be to calculate the rate per minute by dividing the hourly rate by 60, and then multiplying that rate by the number of minutes worked. The correct answer is $10, not $18.
Student's output after tuning	Weng earns $12 per hour, so she earns $12 / 60 = $0.20 per minute. Since she did 50 minutes of babysitting, she earned $0.20 x 50 = $10.
SVAMP Example	
Question	Paul got a box of 110 crayons for his birthday. During the school year he gave 90 crayons to his friends while he lost 412 crayons. How many more crayons did he lose than those he gave to his friends? **Gold answer:** 322
Student's output before tuning	Answer: 392. Explanation: Paul gave 90 crayons to his friends, so he had 110 - 90 = 20 crayons left. So he lost 412 - 20 = 392.
Teachers' correct rationale	Paul lost 412 crayons and gave away 90 crayons. To find out how many more crayons he lost than those he gave to his friends, we can subtract the number of crayons he gave away from the number of crayons he lost: 412 - 90 = 322 Therefore, Paul lost 322 more crayons than those he gave to his friends.
Teachers' mistakes feedback	The mistakes in the student's answer are as follows: The student calculated the number of crayons Paul lost by subtracting the remaining crayons (20) from the total number of crayons he lost (412), which is incorrect. The correct calculation should be 412 - 90 = 322. Therefore, the correct answer should be 412 - 90 = 322, indicating that Paul lost 322 more crayons than he gave to his friends.
Student's output after tuning	Paul lost 412 crayons - 90 crayons = 322 crayons more than those he gave to his friends.
StrategyQA Example	
Question	Would someone in Mumbai refer to Solanum melongena as an eggplant? **Gold answer:** False
Student's output before tuning	True: In Mumbai, Solanum melongena is commonly referred to as an eggplant.
Teachers' correct rationale	False. In Mumbai, people would more commonly refer to "Solanum melongena" as "brinjal" or "baingan" rather than "eggplant." While "eggplant" is widely used in American English, in India, especially in regions like Mumbai, the term "brinjal" is more common. "Baingan" is a Hindi word used across many parts of India.
Teachers' mistakes feedback	The student's answer contains a mistake in the explanation provided. The correct answer to the question is False, but the student mistakenly provided a hint that supports the correct answer. The correct information should be that in Mumbai, Solanum melongena is commonly referred to as a brinjal, not an eggplant.
Student's output after tuning	Answer: False. Solanum melongena is not referred to as "eggplant" in Mumbai. The term "eggplant" is specifically used in American English and Solanum melongena is known as "brinjal" in Mumbai. So, the answer is false.
LogiQA Example	
Question	Some Cantonese don't like chili, so some southerners don't like chili. Which of the following can guarantee the above argument? Options: ["Some Cantonese love chili.", "Some people who like peppers are southerners.", "All Cantonese are southerners.", "Some Cantonese like neither peppers nor sweets."]. **Gold answer:** "All Cantonese are southerners."
Student's output before tuning	Answer: ["Some Cantonese love chili."] Rationale: The argument states that "Some Cantonese don't like chili," which implies that not all Cantonese people dislike chili. Therefore, the statement "Some Cantonese love chili" is the only option that can guarantee the argument.
Teachers' correct rationale	The correct option is: "All Cantonese are southerners." Explanation: The argument states that some Cantonese don't like chili, and to conclude from this that some southerners don't like chili, we need to ensure that Cantonese are included within the broader category of southerners. If all Cantonese are southerners, then the statement about "some Cantonese not liking chili" can be applied to southerners as well, making the argument valid.
Teachers' mistakes feedback	The student's answer is incorrect. The statement "Some Cantonese love chili" does not guarantee the argument about "southerners" provided in the question. The correct answer should be "All Cantonese are southerners." This option directly connects the dislike for chili among some Cantonese to the fact that some southerners don't like chili, making it the most relevant choice to support the argument.
Student's output after tuning	Answer: ["All Cantonese are southerners."] Explanation: The argument states: "Some Cantonese don't like chili, so some southerners don't like chili." To guarantee this argument, we need to establish that some southerners (specifically Cantonese) don't like chili.

Table 13: Case study of distillation impact on student LM's output