MoDE-CoTD: Chain-of-Thought Distillation for Complex Reasoning Tasks with Mixture of Decoupled LoRA-Experts

Xiang Li[1,2], **Shizhu He**[1,2*], **Jiayu Wu**[1,2], **Zhao Yang**[1,2], **Yao Xu**[1,2],
Jun Yang[3], **Haifeng Liu**[3], **Kang Liu**[1,2,4], **Jun Zhao**[1,2]
[1]The Laboratory of Cognition and Decision Intelligence for Complex Systems,
Institute of Automation, Chinese Academy of Sciences
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences
[3] Guangdong OPPO Mobile Telecommunications Corp.,Ltd.
[4] Shanghai Artificial Intelligence Laboratory
{lixiang2022, wujiayu2022}@ia.ac.cn
{shizhu.he, zhao.yang, yao.xu, kliu, jzhao}@nlpr.ia.ac.cn
{yangjun2, blade}@oppo.com

Abstract

Chain-of-thought Distillation (CoTD) aims at distilling Chain-of-thought (CoT) reasoning ability of large language models (LLMs) to much smaller student models. The core of CoTD is using a large teacher model to generate rationales and fine-tune smaller student models. However, current Chain-of-thought Distillation works have the following limitations: 1) Student models are separately distilled from specific reasoning tasks and lack a collaboration mechanism, hindering the enhancement of reasoning performance through collaboration among various reasoning tasks. 2) The parameter update of student models severely harms the CoT reasoning ability on other unseen reasoning tasks not included in the distillation process. In this work, we introduce a novel CoT Distillation method, MoDE-CoTD, which decouples the CoT reasoning abilities out of the student model by distilling multiple LoRA-Experts and freezing the parameters of the student model. Sequentially, LoRA-Experts are combined and adapted to handle both seen and unseen reasoning tasks, enabling collaboration among diverse reasoning tasks to further enhance CoT reasoning performance. Experimental results on 14 datasets (including 4 unseen datasets) demonstrate the strength of MoDE-CoTD, with an average accuracy gain of 6.3% on seen datasets and 7.8% on unseen datasets.

Keywords: Chain-of-Thought, Distillation, LoRA

1. Introduction

Chain-of-thought (CoT) prompting successfully improves the reasoning capabilities of large language models (LLMs) (Wei et al., 2022). By eliciting language models to break down a reasoning task into a series of intermediate steps described by natural language, CoT achieves remarkable performances on various complex tasks such as arithmetic reasoning and commonsense reasoning, even for unseen tasks. However, the ability to solve complex reasoning tasks through CoT Prompting is considered an emergence that appears in very large models with at least tens of billions of parameters (Wei et al., 2022), such as PaLM of 540B (Chowdhery et al., 2022), GPT-3 of 175B (Brown et al., 2020), and LLaMA-2 of 70B (Touvron et al., 2023).

	Seen task	**Uneen tasks**	
Model	**GSM8K**	**Common-SenseQA**	**Reclor**
Random	0.0	20.0	25.0
Flan-t5-large	6.0	82.8	53.4
→ Distillation	7.1 +1.1	39.6 -43.2	25.0 -28.4

Table 1: **Student models suffer from *Catastrophic Degradation on Unseen Tasks*.** Distilling student models with an arithmetic dataset (GSM8K) severely impairs its CoT reasoning ability beyond arithmetic reasoning, such as commonsense reasoning and logical reasoning.

Due to the enormous computational resources or expensive API calls required to utilize CoT-capable LLMs (e.g., LLaMA-2-70B), it is significant to enable complex reasoning in small models that are more feasible for large-scale deployment. Therefore, Chain-of-thought Distillation (CoTD) has been proposed to distill such reasoning capabilities to smaller models (Ho et al., 2023; Magister et al., 2023; Li et al., 2023; Shridhar et al., 2023). Specifically, this line of works applies existing zero-shot or few-shot CoT prompting to generate rationales from very large language models such as ChatGPT and then to fine-tune smaller models such as T5 with those rationales-augmented datasets. Following this procedure, small student models can learn similar reasoning abilities from large teacher models by knowledge distillation (Hinton et al., 2015).

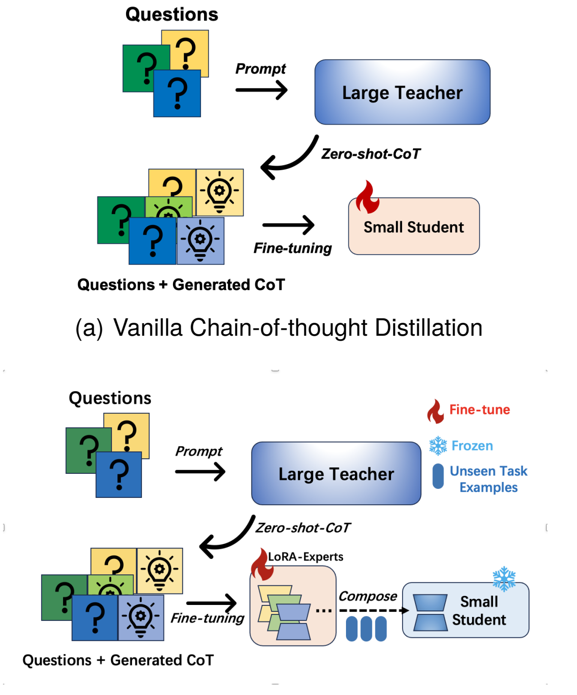
However, in the existing series of CoT distillation works, student models suffer from the following limitations: 1) ***Lack of Cross-Task Collaboration.*** Large teacher models contain cross-task

*Corresponding author

collaboration capabilities due to universal task-solving capabilities emerging from their enormous parameters. While the existing CoT Distillation works focus on distilling each student model for different tasks separately, without allowing for collaboration among the students across different reasoning tasks. In this paper, we believe that enabling cross-task collaboration among student models of diverse reasoning tasks can yield substantial benefits, because a certain reasoning task may require the integration of multiple types of reasoning abilities. For example, answering the arithmetic reasoning question `Claire makes a 3 egg omelet every morning, how many dozens of egg will she eat in 4 weeks?`, not only necessitates arithmetic reasoning but also relies on commonsense reasoning to recognize that there are `7 days in a week` and `12 eggs in a dozen`. 2) ***Catastrophic Degradation on Unseen Tasks***[1]. Due to the update of the overall parameters of the student model, CoT Distillation enhances the CoT reasoning performance of the student model on seen tasks. But this process also changes its internal parameters and disrupts its original ability to solve some general tasks. Therefore, CoTD also severely harms its performance on unseen tasks that are not included in the distillation process. For instance, as illustrated in the Table 1, the student model, distilled using the arithmetic reasoning dataset (GSM8K), loses its original CoT reasoning ability severely when confronted with commonsense reasoning and logical reasoning tasks (CommonSenseQA and Reclor).

To solve the aforementioned limitations, we propose a novel CoT Distillation method named MoDE-CoTD inspired by Low Rank Adaptation (LoRA) (Hu et al., 2021) and Mixture-of-Experts (MoE) (Huang et al., 2023; Shazeer et al., 2017). MoDE-CoTD decouples the CoT reasoning ability of diverse reasoning tasks from the student model to external LoRA modules. These decoupled LoRA modules are then combined and adapted to handle a wide range of reasoning tasks.

Specifically, we consider LoRA modules as experts for different reasoning tasks, which we refer to as *LoRA-Experts*. Rather than fully fine-tuning the entire student model, our approach focuses on LoRA-based fine-tuning those experts and distilling diverse reasoning capabilities from LLMs to them. We select a set of representative reasoning tasks (10 tasks in total) and distill corresponding LoRA-Experts. Then, the proposed CoTD method integrates the parameters of each decoupled LoRA-Expert into a newly constructed LoRA module for any reasoning task, regardless of whether the task is previously encountered or entirely new. This parameter integration process only requires the use of a limited number of examples (e.g., 5 examples) (Huang et al., 2023). (The process is illustrated in Figure 1(b).)



(a) Vanilla Chain-of-thought Distillation

(b) MoDE-CoTD

Figure 1: **Comparison between previous works and our proposed method.** Different from previous works, we only finetune a few external LoRA-Experts for various reasoning tasks. With just a handful of examples from any task, our approach can autonomously compose all LoRA-Experts and adapt to new tasks immediately.

By applying a mixture of decoupled LoRA-Experts for CoT Distillation, our approach offers two notable advantages : 1) ***Cross-task Collaboration.*** For each seen task, our approach allows for collaboration across different tasks by integrating parameters from other LoRA-Experts. This collaboration enhances the student models' performance on each seen task by leveraging the collective knowledge acquired from related tasks, alleviating *Lack of Cross-Task Collaboration* problem. 2) ***Cross-task Generalization.*** For unseen tasks, our approach decouples the CoT reasoning abilities out of the student model. It requires no modification on parameters of student models and avoids the *Catastrophic Degradation on Unseen Tasks*. Moreover, through a mixture of LoRA-Experts, our approach enables the student models seamlessly applied to unseen tasks.

Overall, the main contributions are summarized

[1] The similar phenomenon is also observed by (Fu et al., 2023), where they find the student model distilled by arithmetic reasoning datasets loses all the CoT reasoning ability on BigBench Hard.

as follows:

- We propose to decouple the CoT reasoning abilities from student model to LoRA-Experts for each complex reasoning tasks. LoRA-Experts are then combined and adapted to handle a wide range of reasoning tasks.
- We apply LoRA-Tuning in CoT Distillation and a Mixture-of-Expert strategy to combine LoRA-Experts. To our best knowledge, we are the first to introduce LoRA Tuning and MoE to CoT Distillation.
- Experimental results on 14 public datasets demonstrate that MoDE-CoTD significantly enhanced the reasoning capabilities of the student model, not only on tasks that have been previously encountered but also on unseen tasks.

2. Related Work

Chain-of-Thought Reasoning This work is highly relevant to the seminal work of CoT prompting (Wei et al., 2022). They demonstrate that LLMs can learn to generate intermediate reasoning steps that lead to a problem solution with step-by-step reasoning. This enables state-of-art performance on complex reasoning datasets such as GSM8K (Cobbe et al., 2021). Additionally, (Kojima et al., 2022) find that this can also be done by LLMs in an unsupervised setting, using Zero-shot-CoT. This requires no fine-tuning or examples and substantially outperforms standard zero-shot learning even dew-shot learning on a wide range of tasks.

Chain-of-Thought Distillation Although Chain-of-thought prompting achieves remarkable success on a wide range of natural language processing tasks. However, previous work has shown that CoT extremely relies on large models with enormous parameters (e.g., more than tens of billions of parameters) (Hoffmann et al., 2022; Chowdhery et al., 2022). This leads to overwhelming computational requirements and inference costs, hindering the deployment in practice. As a result, Chain-of-thought distillation (Ho et al., 2023) is proposed to distill CoT reasoning capabilities of LLMs to much smaller models by fine-tuning them on rationales generated by LLMs. Furthermore, (Li et al., 2023; Fu et al., 2023) extends various distillation paradigms. (Magister et al., 2023) extensively explores the improvement of the reasoning ability of small models across multiple model architecture and observes the effects of student model size and data size on accuracy. Apart from that, (Wang et al., 2023) focus on generating more faithful and consistent rationales for CoT Distillation. In contrast to the previous work, where they focus on distilling specialized student models for each reasoning task, in this work, we propose to develop a LoRA-based distillation method, where our student model can tackle diverse reasoning tasks with decoupled modules.

Mixture-of-Experts and LoRA Tuning The Mixture-of-Experts (MoE) has been investigated thoroughly in Natural Language Processing (Shazeer et al., 2017; Komatsuzaki et al., 2022) as an effective way of increasing the model's capacity in parameter size where certain parts of the model are activated for various tasks. In this work, we utilize the idea of MoE for CoT Distillation and adapt distilled student models to diverse reasoning tasks. LoRA (Hu et al., 2021), a parameter-efficient fine-tuning method, facilitates the adaptation of LLMs using a small-scale external module, eliminating the need for fine-tuning the entire model. Recently, (Huang et al., 2023; Zadouri et al., 2023) propose different frameworks to compose multiple LoRA modules. Unlike them, in this work, we utilize LoRA Tuning to distill LoRA-Experts for different reasoning tasks and assemble LoRA-Experts for resolving seen and unseen complex reasoning tasks.

3. Method

In this section, we provide a detailed description of our method, as illustrated in Figure 2, MoDE-CoTD can be divided into three stages:

1. **CoT Generation**, we prompt a very large teacher language model to generate reasoning steps for a wide array of diverse reasoning tasks, preparing datasets for training LoRA-Experts
2. **LoRA-Experts Distillation**, we distill the reasoning capabilities from a large teacher model to a group of LoRA-Experts. For each reasoning task, we distill the corresponding reasoning ability to a specific LoRA-Expert.
3. **LoRA-Experts Composition**, when encountering a specific complex reasoning task, we integrate parameters of all LoRA-Experts into a single model. Each LoRA-Expert is allocated a scalar coefficient, the coefficients are iteratively optimized by leveraging a small set of examples from the target task.

3.1. CoT Generation

For a fair comparison, we adopt the same CoT prompting process as previous works (Ho et al., 2023). First, we utilize a large teacher model to generate CoT reasoning explanations for a given

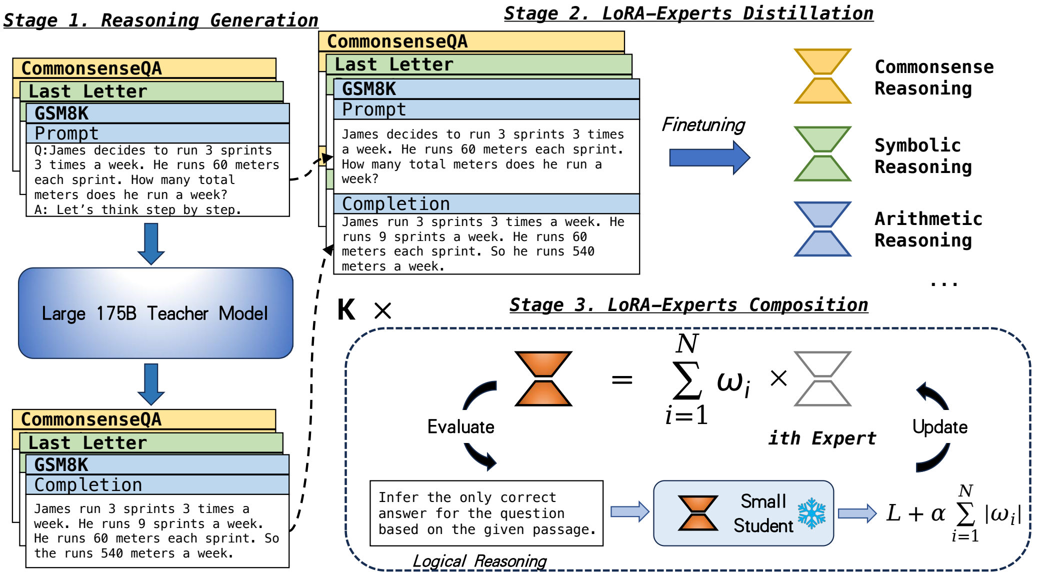

Stage 1. Reasoning Generation
CommonsenseQA
Last Letter
GSM8K
Prompt
Q:James decides to run 3 sprints 3 times a week. He runs 60 meters each sprint. How many total meters does he run a week?
A: Let's think step by step.
Large 175B Teacher Model
CommonsenseQA
Last Letter
GSM8K
Completion
James run 3 sprints 3 times a week. He runs 9 sprints a week. He runs 60 meters each sprint. So the runs 540 meters a week.
Stage 2. LoRA-Experts Distillation
CommonsenseQA
Last Letter
GSM8K
Prompt
James decides to run 3 sprints 3 times a week. He runs 60 meters each sprint. How many total meters does he run a week?
Completion
James run 3 sprints 3 times a week. He runs 9 sprints a week. He runs 60 meters each sprint. So he runs 540 meters a week.
Finetuning
Commonsense Reasoning
Symbolic Reasoning
Arithmetic Reasoning
...
K ×
Stage 3. LoRA-Experts Composition
Evaluate
= $\sum_{i=1}^{N} \omega_i \times$
ith Expert
Update
Infer the only correct answer for the question based on the given passage.
Logical Reasoning
Small Student
$L + \alpha \sum_{i=1}^{N} |\omega_i|$

Figure 2: Overview of our proposed MoDE-CoTD method. **Stage 1**: a very large teacher model is prompted to solve complex questions by generating multi-step reasoning explanations (CoT). The question, CoT/rationale, and answer are used to compose a *CoT Distillation sample*. **Stage 2**: CoTD samples are used to fine-tune a group of LoRA-Experts for different reasoning tasks, a group of lightweight modules distilled by diverse reasoning capabilities. **Stage 3**: to handle any seen and unseen complex reasoning task, the diverse distilled LoRA-Experts are combined and adapted by utilizing a few examples from the target task. The combination coefficients $\{\omega_i\}_{i=1}^{N}$ are optimized using a gradient-free algorithm iteratively.

task. Consider a standard sample S_i consisting of a question q_i and its true answer a_i. Using Zero-shot-CoT (Kojima et al., 2022)[2]. we prompt the teacher model to generate a reasoning explanation, or rationale, $\hat{r}_i$ to solve question q_i and make a final answer prediction $\hat{a}_i$. The resulting text sequence, including the prompt and generations, takes the following form: "Q: <q_i>. A: Let's think step by step. <$\hat{r}_i$> Therefore, the answer is <$\hat{a}_i$>".

Next, we filter the generated samples and reformat them into prompt-completion pairs. Following (Ho et al., 2023; Zelikman et al., 2022; Huang et al., 2022), we filter the samples by comparing the final prediction of the teacher model $\hat{a}_i$ with the ground-truth a_i. Note that implementing this filtering process will result in the loss of some training samples. For each instance i where $\hat{a}_i = a_i$, we repackage $(S_i, \hat{r}_i, \hat{a}_i)$ into a prompt-completion pair $S_i' = (p_i, c_i)$.

3.2. LoRA-Experts Distillation

After building CoT reasoning dataset in 3.1, we utilize LoRA Tuning (Hu et al., 2021) to train LoRA-Experts on diverse reasoning tasks. Specifically, for N distinct reasoning tasks, we separately train N LoRA-Experts, each represented as m_i for task $T_i \in T$. In this work, we choose ten representative reasoning tasks from a wide range of reasoning tasks, including commonsense reasoning, arithmetic reasoning, and temporal reasoning.

LoRA Tuning decomposes the attention weight $W_0 \in R^{d \times k}$ update of the LLM by low-rank matrices, denoted as $W_0 + \delta W = W + AB$ $(A \in R^{d \times r}, B \in R^{r \times k})$, where A and B are low-rank matrics with rank r, a dimension significantly smaller than those of d and k. Compared to full-finetune student models, we distill the reasoning capabilities of LLM to LoRA-Experts with LoRA Tuning, paving the way for extending CoT Distillation to diverse reasoning tasks, regardless of whether the tasks are seen or unseen.

3.3. LoRA-Experts Composition

This section describes the behavior of MoDE-CoTD in the inference stage. After the aforementioned two stages, we obtain a set of LoRA-Experts, denoted as $\{m_i\}_{i=1}^{N}$. Each LoRA-Expert $m_i = A_i B_i$ is distilled from a specific reasoning task. During inference stage, we integrate parameters of all LoRA-Experts into a single module $\hat{m}$, using $\{\omega_1, \omega_2, ..., \omega_N\}$ coefficients, represented as $\hat{m} = \sum_{i=1}^{N} \omega_i \times m_i$, where ω_i is a scalar weight that can be either positive or negative values.

[2] Note that Zero-shot-CoT is a two-step prompting method, where the intermediate CoT is generated in the first step and the final answer is generated in the second step.

The coefficients $\{\omega_1, \omega_2, ..., \omega_N\}$ are calculated through an iterative algorithm with few-shot examples $Q = \{E_i\}_{i=1}^{K}$ from any task, including two phases: 1) EVALUATE phase, 2) UPDATE phase.

3.3.1. EVALUATE Phase

Within the EVALUATE phase, we first compose all LoRA-Experts with current coefficients:

$$\begin{aligned} \hat{m} =& (\omega_1 A_1 + \omega_2 A_2 + ... + \omega_N A_N) \\ & \times (\omega_1 B_1 + \omega_2 B_2 + ... + \omega_N B_N) \end{aligned} \tag{1}$$

Then, we calculate the cross-entropy loss L on Q, furthermore, we incorporate L1 regularization to penalize the sum of the absolute values of all the ωs, preventing extreme values. Consequently, the final optimization objective is:

$$\hat{L} = L + \alpha \sum_{i=1}^{N} |\omega_i| \tag{2}$$

where α serves as a hyperparameter.

3.3.2. UPDATE Phase

Within the UPDATE phase, we optimize the objective calculated. Our goal is to find the best weight $\{\omega_1, \omega_2, ..., \omega_N\}$ and minimize the loss $L + \alpha \sum_{i=1}^{N} |\omega_i|$. Following (Huang et al., 2023), we adopt a gradient-free method for optimization given that ω consists of a relatively small number of parameters.

In terms of the gradient-free method, we leverage Shiwa (Liu et al., 2020), a combinatorial optimization approach. Shiwa offers a variety of algorithms and selects the most appropriate optimization algorithm for different circumstances. In our case, we employ this algorithm to shape the search space of parameter ω, aiming to select the optimal weights based on their performance on the few-shot examples from any seen or unseen task.

The complete process of the calculation of coefficients is summarized by Algorithm 1

Algorithm 1 Coefficients Calculation Algorithm

```
Initialization:$\{w_i \leftarrow 0\}_{i=1}^{N}$, MAXSTEP $\leftarrow$ $T$, $n \leftarrow 0$
repeat
    $n \leftarrow n + 1$
    Compose $\hat{m}$ base on Equation (1)
    Calculate Objective $\hat{L}$ based on Equation (2)

    Update $\{\omega_i\}_{i=1}^{N}$ with Shiwa algorithm
until $n$ =MAXSTEP
Output: optimized weights $\{\omega_i\}_{i=1}^{N}$
```

4. Experiments

4.1. Tasks and Datasets

We evaluate our method on 14 datasets pertaining to five categories of complex reasoning, following (Ho et al., 2023; Kojima et al., 2022). Among them, 10 datasets are used for training LoRA-Experts These include arithmetic (SingleEq, AddSub, MultiArith, GSM8K, Aqua, SVAMP), temporal/spatial reasoning (Date Understanding, Tracking Shuffled Objects), symbolic (Last Letter Concatenation), and common sense (CommonSenseQA) reasoning.

Moreover, we choose 4 datasets for cross-task generalization evaluation. All the 4 datasets are quite different from 10 seen tasks:

1) Coin Flip (Wei et al., 2022) asks the model to answer whether a coin still heads up after people either flip or don't flip the coin.

2) StrategyQA (Geva et al., 2021) is a commonsense reasoning task that poses additional challenges compared to CommonSenseQA due to reasoning steps that are implicit in the question.

3) OpenBookQA (Mihaylov et al., 2018) consists of elementary-level science questions, which require broad common knowledge to solve.

4) Reclor (Yu et al., 2020) is a challenging logical reasoning dataset extracted from logical reasoning questions of standardized graduate admission examinations.

4.2. Teacher and Student Models

For teacher models, we use GPT-3 175B (Brown et al., 2020), provided by the OpenAI API. Unless otherwise stated, we use `text-davinci-002` (Ouyang et al., 2022) as the teacher model. For student models, we consider the instruction-tuned version of T5, Flan-T5-{Base, Large, XL} (Chung et al., 2022). We further train our student model with LoRA-Tuning (Hu et al., 2021) and merely keep the parameter weights of LoRA modules for the subsequent inference stage.

Baseline methods We provide a comparison of MoDE-CoTD (ours) with three baseline methods:

- **Vanilla CoT Distillation** (Ho et al., 2023), a student model is trained for each reasoning task with full-parameter fine-tuning.

- **Zero-shot-CoT** (Kojima et al., 2022): We apply standard Zero-shot-CoT prompt for Flan-t5.

- **Multi-task CoT Distillation**: We extend the vanilla CoT Distillation (Ho et al., 2023) to a multi-task setting by merging all training datasets and training one single multi-task student model for all reasoning tasks.

		Seen tasks										Uneen tasks			
Method	Params	Single Eq	Add Sub	Multi Arith	GSM8K	Aqua	SVAMP	Date Understanding	Shuffled Objects	Last Letter	Common SenseQA	Coin Flip	Strategy QA	Reclor	Open BookQA
Random		0.00	0.00	0.00	0.00	20.00	0.00	17.12	33.33	0.00	20.00	50.00	50.00	25.00	25.00
Teacher: InstructGPT (`text-davinci-002`)															
Zero-shot-CoT	175B	82.24	78.99	78.89	40.26	34.25	64.67	73.87	50.22	56.00	61.75	92.67	53.57	53.00	63.40
Student: Flan-T5 (`base`, `large`, `xl`)															
Zero-shot-CoT	250M	3.29	2.52	9.44	5.60	25.20	**7.33**	18.20	33.78	0.00	67.90	50.00	54.29	36.20	44.60
	780M	5.26	5.88	10.00	6.06	24.01	6.67	18.02	28.88	0.00	82.80	54.67	54.29	39.21	53.47
	3B	9.87	17.64	16.67	6.65	24.80	12.00	30.36	24.44	0.00	84.77	55.33	55.98	50.20	62.38
Vanilla CoT Distillation	250M	4.61	**9.42**	12.22	4.40	29.13	6.00	83.78	**48.89**	50.00	59.05	-	-	-	-
	780M	**11.84**	10.92	**14.44**	7.12	28.35	10.67	84.68	55.11	64.00	66.83	-	-	-	-
	3B	20.39	11.76	**26.67**	7.60	45.67	12.33	88.29	43.11	53.33	74.12	-	-	-	-
Multi-task CoT Distillation	250M	5.22	8.40	8.33	6.00	**47.24**	2.33	80.18	31.55	43.33	73.33	52.67	52.83	**42.40**	43.80
	780M	11.89	**16.81**	8.33	6.36	**46.45**	9.00	79.23	35.56	44.43	73.21	53.33	50.09	40.46	45.45
	3B	22.36	**36.9**	17.22	7.73	48.07	11.33	81.93	52.46	50.00	75.85	56.00	52.11	39.28	50.00
MoDE-CoTD	250M	**5.26**	7.56	**13.89**	**6.11**	39.76	5.33	**85.55**	35.55	**60.67**	**73.79**	**60.00**	**56.18**	38.89	**44.60**
	780M	10.52	10.92	13.89	**7.28**	43.77	**11.33**	**89.19**	**62.22**	**79.33**	**86.73**	**61.33**	**56.47**	**42.44**	**61.61**
	3B	**23.68**	24.37	23.33	**9.78**	**49.21**	**17.33**	**93.69**	**70.67**	**86.00**	**89.56**	**62.33**	**60.99**	**56.83**	**74.81**

Table 2: **MoDE-CoTD Performance.** Accuracy (%) of MoDE-CoTD and baseline methods on 14 tasks (10 seen tasks and 4 unseen tasks) under various settings. 'Random' refers to random-guess performance derived based on the number of choices in multi-choice tasks. We highlight the best method for each setting. For 'Zero-shot-CoT', we use the same prompt setting as (Ho et al., 2023)

For text generation, we use greedy decoding following Wei et al. (2022); Kojima et al. (2022) throughout our experiments.

4.3. Implementation Details

We implement LoRA tuning with Huggingface PEFT library[3], and keep the LoRA tuning hyperparameter at $r = 16$. Regarding the training hyperparameters, we maintain consistency across all LoRA modules, setting the learning rate at $5e-4$, and batch size at 16.

The gradient-free algorithm is implemented by the open-source Nevergrad optimization library. Initially, we set all LoRA-Experts to zero weights and constrain the absolute value of weights under 1.5. And the hyperparameter α is set as 0.05.

4.4. Results

In this section, we present the CoT reasoning performance of our MoDE-CoTD method. We compare our method with baselines within different model sizes. Our method's effectiveness is demonstrated through experimental results on both seen and unseen tasks, showcasing its cross-task collaboration and cross-task generalization capabilities[4].

MoDE-CoTD enables cross-task collaboration for seen tasks Tabel 2 summarize the accuracy of student models using the proposed MoDE-CoTD method, compared to Zero-shot-CoT, vanilla CoT Distillation, and Multi-task CoT Distillation. Although Zero-shot-CoT exhibits remarkable performance on very large language models (Kojima et al., 2022), the same cannot be said for smaller models. Notably, the Zero-shot-CoT approach fails to enable certain complex reasoning tasks, such as the Last Letter task, in all three smaller models. In contrast, CoT Distillation elicits notable reasoning performance, demonstrating significant gains over Zero-shot-CoT almost across all tasks.

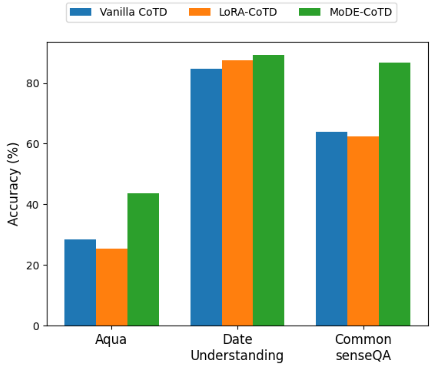


Figure 3: **Benefits of cross-task collaboration** Accuracy (%) of MoDE-CoTD and LoRA-CoTD on three seen tasks. As shown in the results, LoRA-CoTD, without cross-task collaboration, does not perform competitively compared to Vanilla-CoTD. In contrast, MoDE-CoTD, which introduces cross-task collaboration, achieves a significant improvement in accuracy.

In contrast, MoDE-CoTD significantly improves CoT reasoning performance by incorporating cross-task collaboration on seen tasks. MoDE-CoTD achieves the highest accuracy on more than half of

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

[4] Our code is available at https://github.com/Xiang-Li-oss/MoDE-CoTD

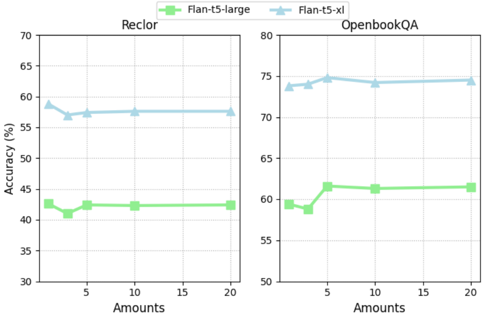


Figure 4: **Effect of amounts of examples.** Accuracy (%) for Reclor and OpenbookQA with various amounts of examples. As shown in the results, reducing the amounts does not cause an apparent decline on accuracy, demonstrating the robustness of our method.

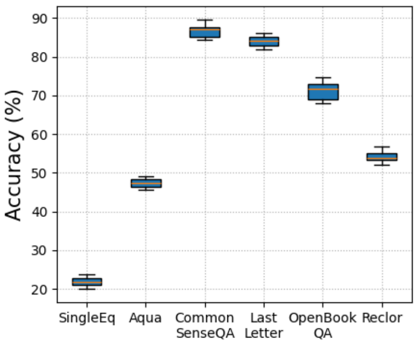


Figure 5: **Robustness Evaluation of MoDE-CoTD.** MoDE-CoTD demonstrates robustness, as sampling different examples does not result in significant fluctuations in accuracy.

all the reasoning tasks. For instance, on tasks like GSM8K and SVAMP, our method surpasses vanilla CoT Distillation by 3% and 5% in terms of accuracy, respectively. This demonstrates that MoDE-CoTD significantly improves CoT reasoning performance by incorporating cross-task collaboration on seen tasks. Moreover, to further confirm the benefits of cross-task collaboration, we compare the performance of MoDE-CoTD with a setting where we solely employ LoRA-Tuning on a student model for a specific reasoning task without incorporating a mixture of LoRA-Experts. This latter approach is referred to as LoRA-CoTD. Figure 3 demonstrates that LoRA-CoTD is usually weaker than Vanilla-CoTD, while MoDE-CoTD significantly outperforms Vanilla-CoTD. This highlights the effectiveness of cross-task collaboration in enhancing the performance of CoTD.

We also observe that tasks that are not overly complex, which include other reasoning tasks (Date Understanding, Shuffled Obejects) and symbolic reasoning (Last Letter), significantly outperform other baselines. This suggests that tasks that are relatively simpler derive greater benefits from tasks that are more complex and challenging, such as arithmetic reasoning.

MoDE-CoTD enables cross-task generalization for unseen reasoning tasks As shown in Table 2, vanilla CoT Distillation has limitations in handling multiple tasks. One approach to address this is through multi-task learning, which involves combining training data from all tasks. However, while Multi-task CoT Distillation shows improvements over vanilla CoT Distillation in certain reasoning tasks (such as GSM8K and Aqua), it falls short in other tasks (such as SVAMP and Last Letter). Apart from this, Multi-task CoT Distillation also struggles with unseen tasks, resulting in a significant decrease in accuracy. For example, it experiences a 22.38% (73.8% - 50%) accuracy drop on OpenbookQA and an 11% (50.20% - 39.20%) accuracy decrease on Reclor compared to Zero-shot-CoT. These findings indicate that applying Multi-task learning to CoT Distillation may cause overfitting on training datasets, severely harming its generalization capability.

In contrast, MoDE-CoTD demonstrates a strong cross-task generalization capability on unseen tasks. The integration of parameters from LoRA-Experts trained on seen tasks provides a foundation for quick adaptation and accurate predictions in the context of new tasks. For example, our method outperforms Zero-shot-CoT on Reclor and OpenBookQA by 6% and 12% respectively. On four unseen tasks, our method consistently outperforms Zero-shot-CoT, and the performance gap widens as the model size increases. This observation aligns with our expectations since the introduction of LoRA modules entails additional parameters. Theoretically, our method will degrade to Zero-shot-CoT if all weights are set to zero, which ensures that our method will be superior to Zero-Shot CoT in most cases due to the incorporation of valuable information through the LoRA-Experts.

Furthermore, MoDE-CoTD demonstrates remarkable performance by outperforming the teacher model on three out of four unseen tasks, while requiring approximately 50 times fewer parameters. Specifically, it surpasses the teacher model by achieving 3% and 11% higher accuracy on Reclor and OpenBookQA, respectively. These results highlight the effectiveness and efficiency of our approach.

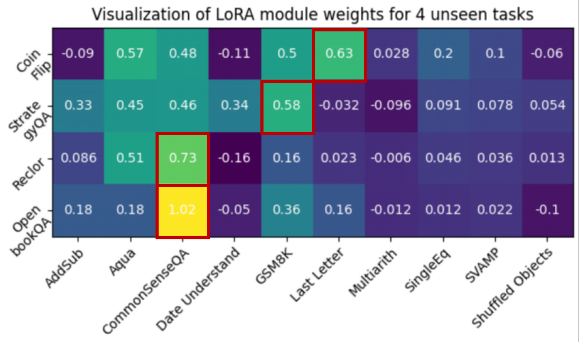


Figure 6: **Visualization of LoRA-Experts.** Weights of 10 LoRA-Experts computed by the gradient-free algorithm for 4 unseen tasks. As shown in the result, experts that are distilled from more complex tasks (for example GSM8K) or more general tasks (for example, CommonSenseQA) play the most important role.

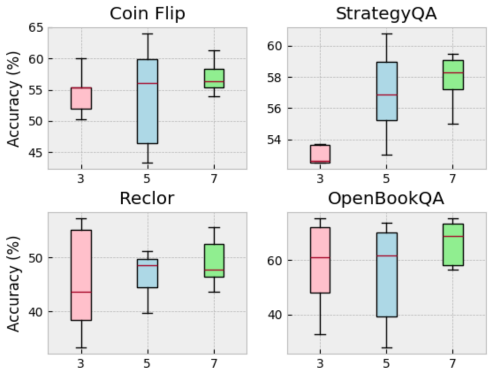


Figure 7: **Ablation study on LoRA-Experts.** As shown in the results, as the amount of experts reduces (from right to left), reasoning accuracy (%) on four unseen tasks noticeably becomes unstable.

4.5. Analysis

In this section, we further analyze the reliability and interpretability of our method by four aspects:

Can MoDE-CoTD still works with fewer examples from new tasks ? The answer is Yes. Figure 4 shows the reasoning accuracy on two unseen tasks: Reclor and OpenbookQA. It is clear that reducing the number of examples from N=5 to N=1 results in a negligible decline in accuracy, with a difference of less than 2%. This highlights the robustness of our method, MoDE-CoTD, as it remains effective even in a one-shot setting. Additionally, in the case of Reclor, we have noticed that N=1 outperforms N=3 and N=5 to a slight extent. This disparity could potentially be attributed to the presence of noise in the rationales generated by the teacher model. Reclor is a complex and challenging logical reasoning dataset constructed from standardized graduate admission examinations. To arrive at the correct answer, the teacher model must generate rationales that are more intricate and lengthier, thereby increasing the risk of introducing additional noise.

We have also observed that increasing the number of examples has minimal impact on reasoning performance. This could be attributed to the fact that the coefficients of LoRA-Experts are already finely optimized with just a few examples (e.g. 5).

Robustness Evaluation of MoDE-CoTD towards example selection. We randomly select 5 examples from the training dataset and repeat this process ten times. We measure the reasoning accuracy of each sample and create box plots that display the distribution of reasoning accuracy. Figure 5 clearly demonstrates that sampling different examples during the inference stage does not cause significant fluctuations in accuracy, proving the robustness of MoDE-CoTD when sampling different examples.

How are LoRA-Experts composed by new tasks? Figure 6 visualizes the corresponding weights of LoRA-Experts calculated by the gradient-free algorithm in our method. By intuition, experts with larger weights play a more important role when generalizing to new tasks. The results displayed in the heatmap confirm our hypothesis. For instance, CommonsenseQA appears to provide the greatest assistance to OpenBookQA, which is reasonable as both tasks involve commonsense reasoning. StrategyQA and Reclor rely heavily on experts like GSM8K and CommonsenseQA, as these tasks require the combination of commonsense knowledge from CommonsenseQA and the ability to solve complex tasks provided by GSM8K and other arithmetic reasoning experts.

How does the amount of LoRA-Experts affect generalization on new tasks? Figure 7 illustrates the results of our ablation study on LoRA-Experts. We investigate the effect of LoRA-Experts by randomly sampling 3, 5, or 7 LoRA-Experts out of 10 Experts in the inference stage. This experiment was repeated 10 times, and the reasoning accuracy was measured for each sampling scenario. To analyze and visualize the results, we created box plots that display the distribution of reasoning accuracy for the different sampling scenarios. As shown in the results, after reducing the number of experts, the accuracy noticeably becomes unstable. While sampling fewer experts may occasionally result in higher maximum accuracy, this trade-off comes at the cost of decreased stability and average performance. The observed pattern is reasonable and logical. When fewer experts are sampled, there is a higher likelihood of including both important and unimportant experts in the selection. This mixture of experts can introduce variability and inconsistency in the model's reasoning process, leading to decreased stability in performance.

5. Conclusion

In this work, we identify two limitations of current Chain-of-thought Distillation works, namely *Lack of Cross-Task Collaboration* and *Catastrophic Degradation on Unseen Tasks*. To solve these limitations, we propose a novel method named MoDE-CoTD. MoDE-CoTD decouples the CoT reasoning capabilities out of the student model by distilling LoRA-Experts instead of fine-tuning the entire student model, avoiding the problem of *Catastrophic Degradation on Unseen Tasks*. Also, by distilling multiple LoRA-Experts from diverse reasoning tasks, MoDE-CoTD alleviates the problem of *Restriction to Specific Tasks*, enhancing the CoT reasoning ability of the student model on a wide range of reasoning tasks.

Acknowledge

This work was supported by the National Key R&D Program of China (No.2022ZD0118501) and the National Natural Science Foundation of China (No.62376270, No.62171183), Youth Innovation Promotion Association CAS, and OPPO Research Fund.

6. Bibliographical References

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

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

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*.

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023. Specializing smaller language models towards multi-step reasoning. *arXiv preprint arXiv:2301.12726*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

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, Toronto, Canada. Association for Computational Linguistics.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. 2022. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. 2023. Lorahub: Efficient cross-task generalization via dynamic lora composition. *arXiv preprint arXiv:2307.13269*.

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*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Aran Komatsuzaki, Joan Puigcerver, James Lee-Thorp, Carlos Riquelme Ruiz, Basil Mustafa, Joshua Ainslie, Yi Tay, Mostafa Dehghani, and Neil Houlsby. 2022. Sparse upcycling: Training mixture-of-experts from dense checkpoints. *arXiv preprint arXiv:2212.05055*.

Liunian Harold Li, Jack Hessel, Youngjae Yu, Xiang Ren, Kai-Wei Chang, and Yejin Choi. 2023. 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)*, pages 2665–2679, Toronto, Canada. Association for Computational Linguistics.

Jialin Liu, Antoine Moreau, Mike Preuss, Jeremy Rapin, Baptiste Roziere, Fabien Teytaud, and Olivier Teytaud. 2020. Versatile black-box optimization. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*, pages 620–628.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching small language models to reason. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1773–1781, Toronto, Canada. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. 2023. Distilling reasoning capabilities into smaller language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7059–7073, Toronto, Canada. Association for Computational Linguistics.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*.

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

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

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

Ted Zadouri, Ahmet Üstün, Arash Ahmadian, Beyza Ermiş, Acyr Locatelli, and Sara Hooker. 2023. Pushing mixture of experts to the limit: Extremely parameter efficient moe for instruction tuning. *arXiv preprint arXiv:2309.05444*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488.

7. Language Resource References

Geva, Mor and Khashabi, Daniel and Segal, Elad and Khot, Tushar and Roth, Dan and Berant, Jonathan. 2021. *Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies*. MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info. PID https://allenai.org/data/strategyqa.

Mihaylov, Todor and Clark, Peter and Khot, Tushar and Sabharwal, Ashish. 2018. *Can a suit of armor conduct electricity? a new dataset for open book question answering*. PID https://github.com/allenai/arc-solvers.

Wei, Jason and Wang, Xuezhi and Schuurmans, Dale and Bosma, Maarten and Xia, Fei and Chi, Ed and Le, Quoc V and Zhou, Denny and others. 2022. *Chain-of-thought prompting elicits reasoning in large language models*.

Weihao Yu and Zihang Jiang and Yanfei Dong and Jiashi Feng. 2020. *ReClor: A Reading Comprehension Dataset Requiring Logical Reasoning*. PID https://github.com/yuweihao/reclor.