Neural-Symbolic Collaborative Distillation: Advancing Small Language Models for Complex Reasoning Tasks

Huanxuan Liao[1,2], **Shizhu He**[1,2*], **Yao Xu**[1,2], **Yuanzhe Zhang**[4], **Kang Liu**[1,2,3], **Jun Zhao**[1,2]

[1] The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3] Shanghai Artificial Intelligence Laboratory, Shanghai, China
[4] National Science Library, Chinese Academy of Sciences, Beijing, China
liaohuanxuan2023@ia.ac.cn {yao.xu, shizhu.he, kliu, jzhao}@nlpr.ia.ac.cn

Abstract

In this paper, we propose **Ne**ural-**Sy**mbolic **C**ollaborative **D**istillation (**NesyCD**), a novel knowledge distillation method for learning the complex reasoning abilities of Large Language Models (LLMs, e.g., >13B). We argue that complex reasoning tasks are difficult for Small Language Models (SLMs, e.g., $\leq$ 7B), as these tasks demand not only general cognitive abilities but also specialized knowledge, which is often sparse and difficult for these neural-based SLMs to effectively capture. Therefore, NesyCD distills the general capabilities and specialized knowledge in LLMs in different ways. On the one hand, we distill only general abilities from teacher LLMs into the student SLMs of parameterized neural networks. On the other hand, for the specialized abilities and uncommon knowledge of a complex reasoning task, we employ a symbolic knowledge distillation approach to obtain and store the specialized knowledge within a symbolic knowledge base (KB). By decoupling general and specialized capabilities, the proposed NesyCD can achieve superior performance cost-effectively, utilizing smaller models and blending parameterized neural networks with symbolic KB. Moreover, the specialized KB generalizes well and is comprehended and manipulated by humans. Our experiments show that NesyCD significantly boosts SLMs' complex reasoning performance on in-domain (BBH, GSM8K) and out-of-domain (AGIEval, ARC) datasets. Notably, our approach enabled the LLaMA3-8B and Qwen2-7B to surpass GPT-3.5-turbo in performance and come close to matching LLaMA3-70B, despite the latter having 9$\times$ more parameters.

Code — https://github.com/Xnhyacinth/NesyCD
Extended version — https://arxiv.org/abs/2409.13203

1 Introduction

Large Language Models (LLMs) (Yang et al. 2024) excel in various complex reasoning tasks such as mathematical (Yu et al. 2023a), commonsense (Zhao, Lee, and Hsu 2023) and symbolic reasoning (Suzgun et al. 2022) with In-Context Learning (ICL) (Ye et al. 2023) and Chain-of-Thought (CoT) Prompting (Kaplan et al. 2020). Due to the high computational costs and expensive API calls required for LLMs (e.g., ChatGPT and LLaMA3-70B), enhancing Small Language Models (SLMs, e.g., $\leq$ 7B) to handle complex reasoning efficiently is more practical and crucial for large-scale deployment.

To meet the practical needs mentioned above, many research efforts (Magister et al. 2023; Hsieh et al. 2023; Li et al. 2024b) in recent years have proposed the transfer of reasoning capabilities from teacher LLMs to student SLMs through CoT distillation (shown in the middle of Figure 1). Specifically, LLMs generate high-quality rationales, which are then utilized to fine-tune SLMs. This CoT distillation enhances the performance of SLMs in many tasks that require complex reasoning abilities such as arithmetic (Cobbe et al. 2021) and symbolic reasoning (Wei et al. 2022).

Despite some progress in CoT distillation, significant challenges persist that limit the performance of SLMs in complex reasoning tasks: 1) **Inconsistency in Capabilities between Teacher and Student Models**: Existing methods often ignore the gap between the knowledge modeling and complex reasoning capabilities of LLMs and SLMs. Due to fewer parameters, student models struggle to acquire the comprehensive knowledge necessary for complex reasoning tasks (Kang et al. 2023). As illustrated in Figure 1, an SLM trained with traditional CoT distillation fails to solve hard questions that the teacher model can handle effectively. 2) **Difficulty in Modeling Sparse Specialized Knowledge**: Neural knowledge distillation faces challenges in representing sparse and specialized knowledge due to the limited parameter space of distilled SLMs. In complex reasoning tasks, particularly with hard questions, specialized knowledge may contradict general knowledge (e.g., velocity superposition in relative motion). This sparse and specialized knowledge is hard to model in small-scale models, significantly impacting the SLM's performance on challenging questions and deteriorating its ability to handle unseen tasks.

In this paper, we argue that complex reasoning tasks require both general knowledge (e.g., numerical addition) and specialized knowledge (e.g., relative displacement). It can be simply understood as follows: the former refers to what SLMs can effectively model and is primarily used for answering high-frequency, easy questions, while the latter in-

*Corresponding author

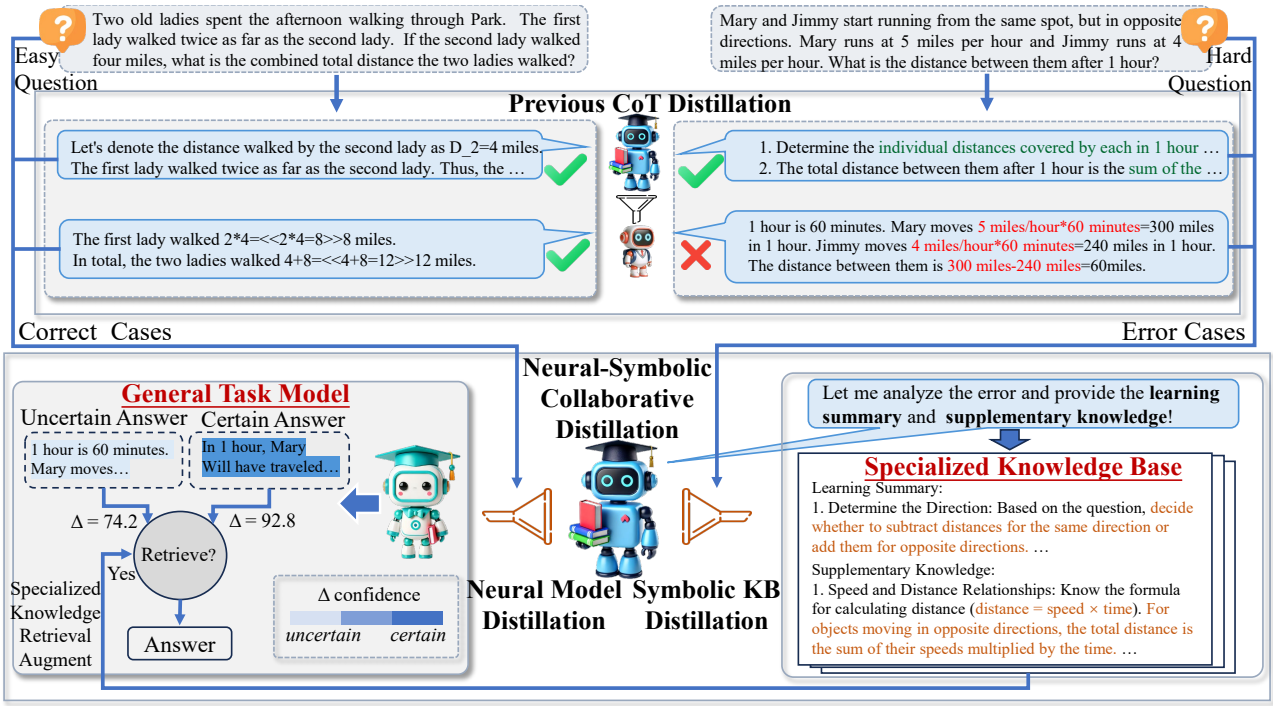

Figure 1: CoT distillation aims to train SLMs with the generated rationales obtained from LLMs, which is often limited by the SLMs' capabilities and frequently struggles to handle hard questions. The proposed NesyCD addresses this by decoupling the general and specialized knowledge of LLMs through SLMs' error analysis. It employs neural-based SLMs to model general knowledge while utilizing a symbolic specialized knowledge base (KB) to store specific knowledge. By adaptively utilizing the KB, NesyCD enhances the SLM's ability to handle complex reasoning tasks.

volves aspects that SLMs find challenging to model and are essential for addressing low-frequency, hard questions. For instance, as shown in Figure 1, answering the right hard question demands domain-specific knowledge and advanced reasoning skills, like applying physics formulas to calculate displacement and understanding relative displacement.

To tackle the aforementioned challenges in existing CoT distillation methods, we propose a novel **Ne**ural-**Sy**mbolic **C**ollaborative **D**istillation (NesyCD) which transfers the general reasoning capabilities and common knowledge of LLMs to SLMs. Unlike using only neural-based models to build students before, we use a symbolic knowledge base (KB) to model and store relatively sparse specialized knowledge. Firstly, we gather correct and error cases made by SLMs fine-tuned with CoT distillation. Next, LLMs analyze error cases, extracting specialized knowledge through elaborate prompts and storing them in a symbolic KB. Finally, we fine-tune the SLMs with specialized knowledge augmented distillation, enhancing SLMs' abilities for hard questions. Moreover, to enhance the student SLMs' robustness against potentially noisy retrieved knowledge, we incorporate novel auxiliary tasks like augmented distillation (AD), answer prediction (AP) and direct CoT (DC) for multi-task learning to effectively utilize specialized knowledge.

To validate the effectiveness of NesyCD, we empirically demonstrate that it significantly improves the baseline performance of several open-source SLMs, such as TinyLLaMA (Zhang et al. 2024a) and LLaMA2-7B (Touvron et al. 2023), across various benchmarks including GSM8k (Cobbe et al. 2021) for mathematical reasoning, BBH (Suzgun et al. 2022) and AGIEval (Zhong et al. 2023) for general reasoning, and ARC (Clark et al. 2018) for factual knowledge. Additionally, our extensive analysis shows that NesyCD is efficient regarding training data and model size. Specifically, the NesyCD-enhanced 1.1B TinyLLaMA outperforms the fine-tuned LLaMA2-7B and achieves superior results using only a quarter of the full training data compared to other strong baselines. Our findings and contributions are as follows:

- We propose a neural-symbolic collaborative distillation method, which, to our knowledge, is the first approach to leverage a co-distillation framework that integrates neural-based models with symbolic knowledge bases for learning the complex reasoning capabilities of LLMs.
- We distinguish complex reasoning into general and specialized abilities through SLMs' error analysis. General abilities are modeled by a neural network, while specialized abilities are captured by a symbolic KB. Integrating these components enhances SLMs' complex reasoning, leading to more efficient models and reduced costs.
- The experimental results demonstrate that the proposed NesyCD significantly enhances the performance of SLMs across wide benchmarks for knowledge, mathematical, symbolic and other complex reasoning tasks both in-domain and out-of-domain.

2 Related Work

2.1 CoT Distillation from LLMs

The Chain-of-Thought (CoT) reasoning ability of LLMs, characterized by step-by-step question solving, is known as an emergent ability to improve performance in various reasoning tasks. Recent works (Ho, Schmid, and Yun 2022; Fu et al. 2023) endeavor to transfer the CoT reasoning capabilities of LLMs to SLMs. Std-CoT (Magister et al. 2023) involves fine-tuning SLMs directly using CoTs extracted from teacher LLMs. Subsequent studies (Hsieh et al. 2023; Li et al. 2024b) have proposed treating the learning of rationales and answers as separate optimization objectives. CasCoD (Dai et al. 2024) takes a different approach by decomposing the traditional single-step learning process into two cascaded steps. However, the performance of these methods is hindered by the limited knowledge and capabilities of SLMs with fewer parameters (Ho, Schmid, and Yun 2022; Kang et al. 2023). This deficiency is particularly detrimental in complex reasoning tasks that require specialized knowledge and sophisticated reasoning skills. To address this issue, we propose to enhance SLMs by integrating knowledge retrieved from the specialized knowledge base (KB) generated by teacher LLMs.

2.2 Knowledge-Augmented LMs

Knowledge-augmented LMs (KALMs) enhance their reasoning by utilizing external KBs. A common approach involves retrieving relevant passages from sources like Wikipedia based on questions (Chen et al. 2017). KARD (Kang et al. 2023) applies KALMs to knowledge-intensive tasks and finds it crucial for accurate answers and factual rationales. However, challenges like chunk indexing and independent encoding of documents can hinder KALMs' effectiveness in using external KB. To address this, we propose harnessing the world knowledge and reasoning capabilities of LLMs to generate specialized knowledge for KALMs, including learning summaries and supplementary knowledge to boost complex reasoning abilities.

2.3 Learning from Errors

Humans learn from their errors to avoid repeating them, and this capability has inspired efforts to enhance LLMs (Li et al. 2023; Wang et al. 2024). LLM2LLM (Lee et al. 2024) employs an instructor model to help target models learn from their errors. TRAN (Tong et al. 2024) uses a rule-based system to prevent past errors, while LEAP (Zhang et al. 2024b) extracts and integrates principles from LLMs' errors into prompts. However, these approaches have not been adapted to improve SLMs, and the principles used in reasoning remain static rather than dynamically tailored based on the model's capabilities.

3 Methods

We propose Neural-Symbolic Collaborative Distillation (NesyCD), which consists of four learning processes (illustrated in Figure 2): 1) **General Distillation** (§3.1), where a large language model (LLM) serves as a teacher model (General Teaching) $\mathcal{T}_G$ to generate rationales. Subsequently, a small language model (SLM) is fine-tuned to generate these rationales and provide answers to given questions, resulting in the Student Model (Primary Learning) $\mathcal{S}_P$; 2) **Demonstration Collection** (§3.2), where we evaluate the performance of the $\mathcal{S}_P$ on the specific task and dataset, and collect the correct and error cases for the following steps; 3) **Symbolic Knowledge Distillation** (§3.3), where the previous teacher model, acting as the Targeted Teaching model $\mathcal{T}_T$, analyzes and generates specialized knowledge aimed at the errors made by $\mathcal{S}_P$. This knowledge assists $\mathcal{S}_P$ in addressing similar tasks in the future and is then stored in a specialized knowledge base (KB). 4) **Symbolic KB Augmented Neural Distillation** (§3.4), where the student model (Enhanced Learning) $\mathcal{S}_E$ is initialized with the fine-tuned $\mathcal{S}_P$. Using multi-task learning, fine-tune $\mathcal{S}_E$ to generate rationales and answers based on both questions and retrieved specialized knowledge, as well as on the questions alone.

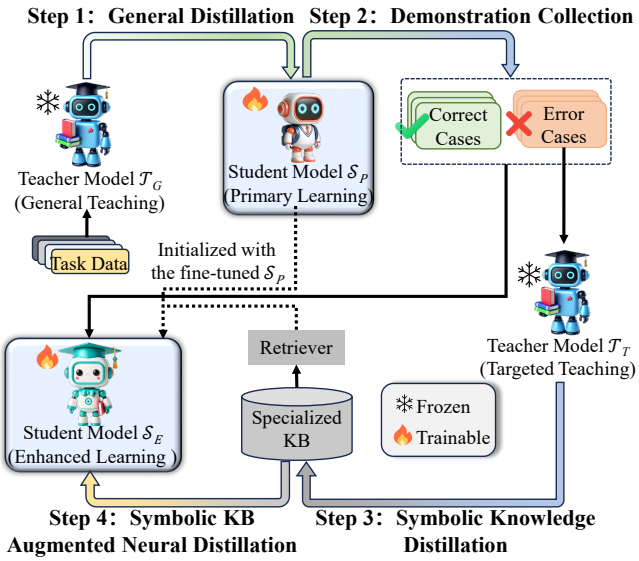

Figure 2: Overview of NesyCD. 1) General Distillation (§3.1): Fine-tune the student $\mathcal{S}_P$ to generate rationales obtained from the teacher $\mathcal{T}_G$ and answers. 2) Demonstration Collection (§3.2): Evaluate $\mathcal{S}_P$ and collect correct and error cases addressed by $\mathcal{S}_P$. 3) Symbolic Knowledge Distillation (§3.3): The teacher $\mathcal{T}_T$ analyzes errors and generate specialized KB. 4) Symbolic KB Augmented Neural Distillation (§3.4): Use multi-task learning to fine-tune $\mathcal{S}_E$, enabling it to effectively utilize retrieved specialized knowledge.

3.1 General Distillation

Rationale Generation with LLMs: The ability to generate high-quality rationales is known as the emergent ability of LLMs (Ho, Schmid, and Yun 2022). Our objective is to transfer this capability to SLMs through CoT distillation. First, we employ CoT prompts (Wei et al. 2022) to guide the $\mathcal{T}_G$ in generating CoT. We generate rationales for each training data point $\mathcal{D}_{\text{train}} = \{(\boldsymbol{q}_i, \boldsymbol{a}_i)\}_{i=1}^{n}$, where $\boldsymbol{q}_i$ is a question and $\boldsymbol{a}_i$ is an answer, retaining only those that align with the

correct answers in the dataset (Dai et al. 2024).

$$\boldsymbol{r}_{ij} = \mathcal{T}_G(\boldsymbol{p}, \boldsymbol{q}_i, \boldsymbol{a}_i) \quad (1)$$

where $\boldsymbol{r}$ are generated rationales, $j \in \{1, ..., l\}$ and $\boldsymbol{p}$ is the CoT prompt which is shown in Appendix F.1.

Fine-tuning SLMs with Rationales: We initially conduct Std-CoT (Magister et al. 2023) to develop the $\mathcal{S}_P$. Given a question $\boldsymbol{q}_i$, we fine-tune the $\mathcal{S}_P$ with trainable parameters θ, to generate the rationale $\boldsymbol{r}_{ij}$ derived from the $\mathcal{T}_G$ and answer $\boldsymbol{a}_i$. We aim to minimize the negative log-likelihood of the sequence comprising the rationale $\boldsymbol{r}_{ij}$ and the answer $\boldsymbol{a}_i$, ensuring that the rationale precedes the answer.

$$\mathcal{L}_{\text{Std-CoT}}(\theta) = -\frac{1}{n \cdot l} \sum_{i=1}^{n} \sum_{j=1}^{l} \log p_\theta(\boldsymbol{r}_{ij}, \boldsymbol{a}_i \mid \boldsymbol{q}_i) \quad (2)$$

Rationales offer a deeper understanding of the reasoning behind answers, helping SLMs to respond more accurately (Hsieh et al. 2023). However, SLMs with limited parameters may struggle to retain all training data and complex reasoning capabilities, which can affect the quality of rationale generation (Kang et al. 2023). Furthermore, this implicit learning may cause SLMs to focus on answering questions directly after reading, potentially impairing generalization in reasoning (Dai et al. 2024). Therefore, it is essential to assess the knowledge and capabilities that SLMs fail to acquire and have teacher LLMs generate specialized knowledge. This approach enables SLMs to retrieve and utilize knowledge effectively, enhancing their ability to produce high-quality rationales and perform complex reasoning when needed.

3.2 Demonstration Collection

The $\mathcal{S}_P$ analyzes $\mathcal{D}_{\text{train}}$ to generate predicted rationales $\hat{\boldsymbol{r}}$ and answers $\hat{\boldsymbol{a}}$. Incorrect solutions are identified by comparing each $\hat{\boldsymbol{a}}_i$ with the actual answer $\boldsymbol{a}_i$. The collected errors, $\mathcal{D}_{\text{neg}}$, reveal the weaknesses of the student model.

$$\mathcal{D}_{\text{neg}} = \{(\boldsymbol{q}_i, \hat{\boldsymbol{r}}_i, \boldsymbol{a}_i) \mid \hat{\boldsymbol{a}}_i \neq \boldsymbol{a}_i, (\boldsymbol{q}_i, \boldsymbol{a}_i) \in \mathcal{D}_{\text{train}}\} \quad (3)$$

In our view, correct cases are those manageable by conventional distillation models using general knowledge. In contrast, error cases are difficult for small-scale neural models to handle effectively, requiring additional specialized knowledge and advanced reasoning capabilities. This specialized knowledge is sparse, making it more cost-effective to represent through a symbolic KB, as neural networks would need a larger parameter scale to capture it.

3.3 Symbolic Knowledge Distillation

The $\mathcal{T}_T$ examines each error $\hat{\boldsymbol{r}}_i$ in $\mathcal{D}_{\text{neg}}$ and generates specialized knowledge $\boldsymbol{k}_i$, including generalized learning summaries $\boldsymbol{k}_i^m$ and supplemental knowledge $\boldsymbol{k}_i^p$. This addresses the issues of insufficient knowledge and lack of reasoning ability in the SLM. For each question $\boldsymbol{q}_i$ in $\mathcal{D}_{\text{neg}}$, we construct the teacher model's prompt[1] $\boldsymbol{p'}$ that incorporates the student's incorrect rationale $\hat{\boldsymbol{r}}_i$, and the correct answer $\boldsymbol{a}_i$ which constructs a specialized KB, represented as $\mathcal{D}_{\text{k}} = \{(\boldsymbol{q}_i, \boldsymbol{k}_i) \mid (\boldsymbol{q}_i, \hat{\boldsymbol{r}}_i, \boldsymbol{a}_i) \in \mathcal{D}_{\text{neg}}$. The process is primarily driven by the identification and correction of errors.

$$\boldsymbol{k}_i = \mathcal{T}_T(\boldsymbol{p'}, \boldsymbol{q}_i, \hat{\boldsymbol{r}}_i, \boldsymbol{a}_i) \quad (4)$$

[1]The prompt for generating $\boldsymbol{k}_i$ is in the Appendix F.2.

3.4 Symbolic KB Augmented Neural Distillation

Inspired by knowledge augmentation (Kang et al. 2023), we propose retrieving relevant specialized knowledge from the specialized KB which is generated through error analysis of the SLM to support its memory and reasoning capabilities. Acquiring specialized knowledge is crucial for training the SLM to produce high-quality rationales, subsequently leading to the correct answers to given questions. In alignment with prior knowledge-intensive tasks, we employ a dense retriever Contriever (Izacard et al. 2021) to retrieve a set of relevant questions for each question: $\mathcal{Q}_i = \text{topk}(\rho(\boldsymbol{q} \mid \boldsymbol{q}_i; \mathcal{D}_{\text{k}}), m)$, where ρ scores the questions $\boldsymbol{q} \in \mathcal{D}_{\text{k}}$ based on their relevance to the question $\boldsymbol{q}_i$, and topk selects the top m questions with the highest relevance scores. Then we can get the specialized knowledge:

$$\mathcal{K}_i = \{\boldsymbol{k}_j \mid (\boldsymbol{q}_j, \boldsymbol{k}_j) \in \mathcal{D}_{\text{k}}, \boldsymbol{q}_j \in \mathcal{Q}_i\} \quad (5)$$

Finally, we fine-tune the student model $\mathcal{S}_E$, initialized with the fine-tuned student $\mathcal{S}_P$, using the retrieved specialized knowledge $\mathcal{K}_i$ to generate the rationale $\boldsymbol{r}_{ij}$ and the answer $\boldsymbol{a}_i$ for the question $\boldsymbol{q}_i$.

$$\mathcal{L}_{\text{NesyCD}}(\theta) = -\frac{1}{n \cdot l} \sum_{i=1}^{n} \sum_{j=1}^{l} \log p_\theta(\boldsymbol{r}_{ij}, \boldsymbol{a}_i \mid \boldsymbol{q}_i, \mathcal{K}_i) \quad (6)$$

where the rationale and answer are sequentially generated as we did in §3.1. Beyond fine-tuning the SLM with **augmented distillation** (AD), we propose two auxiliary tasks in **multi-task learning** to enhance reasoning capabilities. These tasks aim to improve the SLM's ability to integrate and apply specialized knowledge effectively: 1) **Answer Prediction** (AP), which generates answers directly, aiming to help SLMs internalize the reasoning required for direct questions that do not necessitate a CoT; 2) **Direct CoT** (DC), which relies solely on the SLM's intrinsic knowledge (i.e., $\mathcal{K}_i$ is empty) to address relatively easy questions.

During the inference stage, as illustrated at the bottom of Figure 1, we determine the necessity of retrieval based on the model confidence (Wang and Zhou 2024). For instances requiring retrieval, we extract the most specialized knowledge from the specialized knowledge base relevant to the question to assist in generating the rationale and answer.

$$\Delta_{\text{answer}} = \frac{1}{|\text{answer}|} \sum_{x_t \in \text{answer}} p(x_t^1 \mid x_{<t}) - p(x_t^2 \mid x_{<t}) \quad (7)$$

Here, x_t^1 and x_t^2 represent the top two tokens at the t-th decoding step. These tokens are selected based on their highest post-softmax probabilities from the vocabulary, given that x_t is part of the answer tokens. More analysis about Δ_{answer} can be seen in Appendix B.1.

4 Experiments

In this section, we conduct extensive experiments and comprehensive analysis to evaluate the effectiveness of NesyCD on both in-domain (ID) and out-of-domain (OOD) datasets.

Methods	In-Domain		Out-Of-Domain					Average
	BBH-test	GSM8K	BB-sub	AGIEval	GSM8K-PLUS	ARC-E	ARC-C	
# Closed-source model and Open-source models (Zero-shot-CoT)								
GPT-3.5-turbo (*Teacher*)	43.2	72.6	44.0	50.5	55.9	91.8	84.1	63.2
LLaMA-3-70B-Instruct	62.6	89.2	51.0	66.3	72.9	97.6	93.2	76.1
# TinyLLaMA-1.1B based								
Zero-shot (Radford et al. 2019)	14.0	2.0	17.7	17.8	1.5	19.4	15.0	12.5
Zero-shot-CoT (Kojima et al. 2022)	13.5	1.4	17.7	10.4	1.3	16.0	13.4	10.5
Fine-tuning	48.8	3.5	26.0	21.2	3.7	28.0	24.6	22.3
Knowledge-Augmented Fine-tuning	49.3	3.7	27.4	21.9	3.3	29.4	25.3	22.9
Std-CoT (Magister et al. 2023)	$47.8_{\pm.43}$	$7.9_{\pm.27}$	$27.6_{\pm.31}$	$21.5_{\pm.56}$	$4.3_{\pm.62}$	$28.2_{\pm.69}$	$25.0_{\pm.48}$	23.2
MT-CoT (Li et al. 2024b)	$44.1_{\pm.78}$	$4.1_{\pm.35}$	$25.0_{\pm.45}$	$21.4_{\pm.64}$	$2.8_{\pm.83}$	$33.5_{\pm.52}$	$25.1_{\pm.59}$	22.3
Step-by-step (Hsieh et al. 2023)	$42.4_{\pm.56}$	$4.3_{\pm.47}$	$26.2_{\pm.38}$	$21.1_{\pm.72}$	$3.1_{\pm.54}$	$29.6_{\pm.61}$	$25.9_{\pm.66}$	21.8
KARD (BM25) (Kang et al. 2023)	$49.5_{\pm.61}$	$7.6_{\pm.40}$	$26.9_{\pm.43}$	$20.2_{\pm.48}$	$4.0_{\pm.77}$	$28.2_{\pm.85}$	$26.5_{\pm.91}$	23.3
CasCoD (Dai et al. 2024)	$48.1_{\pm.49}$	$6.8_{\pm.39}$	$23.1_{\pm.64}$	$19.4_{\pm.73}$	$4.8_{\pm.48}$	$29.0_{\pm.63}$	$27.1_{\pm.42}$	22.6
NesyCD (*ours*)	$\mathbf{66.3}_{\pm.42}$	$\mathbf{11.8}_{\pm.83}$	$\mathbf{30.6}_{\pm.27}$	$\mathbf{23.1}_{\pm.41}$	$\mathbf{7.2}_{\pm.93}$	$\mathbf{36.2}_{\pm.76}$	$\mathbf{29.0}_{\pm.58}$	**29.3**
# LLaMA2-7B based								
Zero-shot (Radford et al. 2019)	17.3	2.7	18.6	19.2	2.4	25.2	20.6	17.0
Zero-shot-CoT (Kojima et al. 2022)	13.5	3.1	12.2	10.3	2.1	29.1	20.2	12.9
Fine-tuning	57.8	5.8	33.3	31.0	5.8	73.3	56.3	37.6
Knowledge-Augmented Fine-tuning	58.7	6.3	34.2	31.8	6.1	75.1	57.0	38.5
Std-CoT (Magister et al. 2023)	$58.1_{\pm.74}$	$20.5_{\pm.71}$	$30.7_{\pm.48}$	$23.6_{\pm.65}$	$12.0_{\pm.26}$	$73.4_{\pm.81}$	$55.9_{\pm.78}$	39.2
MT-CoT (Li et al. 2024b)	$46.4_{\pm.52}$	$7.5_{\pm.48}$	$28.1_{\pm.55}$	$32.1_{\pm.53}$	$5.8_{\pm.39}$	$70.3_{\pm.67}$	$55.7_{\pm.45}$	35.1
Step-by-step (Hsieh et al. 2023)	$53.9_{\pm.69}$	$8.3_{\pm.57}$	$32.3_{\pm.33}$	$32.4_{\pm.40}$	$5.6_{\pm.41}$	$74.9_{\pm.52}$	$60.0_{\pm.56}$	38.2
KARD (BM25) (Kang et al. 2023)	$59.2_{\pm.93}$	$23.5_{\pm.62}$	$30.8_{\pm.66}$	$29.2_{\pm.79}$	$15.2_{\pm.54}$	$70.2_{\pm.71}$	$55.4_{\pm.48}$	40.5
CasCoD (Dai et al. 2024)	$59.6_{\pm.78}$	$23.6_{\pm.87}$	$32.2_{\pm.71}$	$28.8_{\pm.63}$	$14.5_{\pm.68}$	$72.6_{\pm.49}$	$56.7_{\pm.83}$	41.1
NesyCD (*ours*)	$\mathbf{75.5}_{\pm.69}$	$\mathbf{32.4}_{\pm.53}$	$\mathbf{36.9}_{\pm.38}$	$\mathbf{33.6}_{\pm.71}$	$\mathbf{24.1}_{\pm.47}$	$\mathbf{77.5}_{\pm.89}$	$\mathbf{60.8}_{\pm.56}$	**48.7**

Table 1: Performance (%) of LLaMA2-7B (Touvron et al. 2023) and TinyLLaMA-1.1B (Zhang et al. 2024a) with different methods across seven selected datasets. **Bold** indicates the best in each setting. We report the mean and standard deviation of accuracy with 3 different runs for CoT distillation methods. We provide a systematic case study in Appendix E.

4.1 Datasets

Following (Wang et al. 2023a; Ying et al. 2024), we focus on three practical abilities: factual, mathematical, and general reasoning. For each ability, we select a relevant public dataset, integrate its training data into the target dataset $\mathcal{D}_{\text{train}}$ for mixed training, and combine its test data into the evaluation dataset $\mathcal{D}_{\text{eval}}$. Additionally, each ability includes an OOD dataset in $\mathcal{D}_{\text{eval}}$. This setup allows us to evaluate the model's ability to generalize and enhance performance beyond the ID training environment.

Factual Reasoning: We select the Multitask Language Understanding (MMLU) (Hendrycks et al. 2021a) as the ID dataset, which includes multiple-choice questions across 57 subjects. For OOD evaluation, we use the ARC (Clark et al. 2018), comprising both Easy and Challenge segments.

Mathematical Reasoning: We select MetaMathQA (Yu et al. 2023a) as the ID dataset, which includes a high-quality collection of forward and reverse mathematical reasoning question-answer pairs, derived from GSM8K (Cobbe et al. 2021) and MATH (Hendrycks et al. 2021b). For OOD evaluation, we use GSM8K and GSM8K+ (Li et al. 2024a).

General Complex Reasoning: We chose BIG-Bench Hard (BBH) (Suzgun et al. 2022) as the ID dataset, which includes 27 challenging tasks spanning arithmetic, symbolic reasoning, and more, derived from BIG-Bench (BB) (Srivastava et al. 2022). Most of the data consists of multiple-choice questions. For OOD evaluation, we use BB-Sub filtered by CasCoD, and AGIEval (Zhong et al. 2023) subtasks about English multiple-choice questions.

4.2 Baselines

We compare our method with the following baselines: *1)* **Teacher & Vanilla Student** in Zero-shot (Radford et al. 2019), Zero-shot-CoT (Kojima et al. 2022). *2)* **Fine-tuning** involves fine-tuning a model to generate answers given only questions. The performance of the baselines above illustrates the capability of SLMs to solve tasks using only training data, without external guidance or additional knowledge. *3)* **CoT distillation** includes **Std-CoT** (Magister et al. 2023) which is the standard CoT distillation method, enabling direct fine-tuning of the student model with CoT data; **Step-by-step** (Hsieh et al. 2023) is a multi-task method that extracts rationales and answers separately; **MT-CoT** (Li et al. 2024b) is another multi-task method that optimizes both answer prediction and CoT generation simultaneously; **CasCoD** (Dai et al. 2024) decomposes the traditional single-step learning process into two cascaded learning steps. *4)* **Knowledge-Augmentation** involves attaching retrieved passages to the question during both training and inference. This includes **Knowledge-Augmented Fine-tuning** focuses on generating answers only, and **KARD** (Kang et al. 2023) emphasizes learning the generation of rationales.

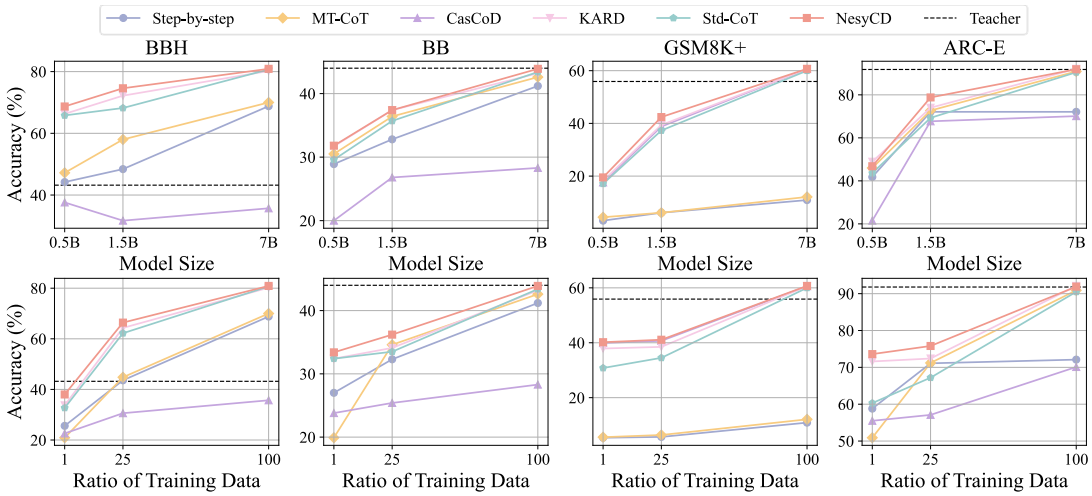

Figure 3: Efficiency on training data and model size. The backbone model for the data size variation is Qwen2-1.5B.

Retriever	BBH	BB	AGIEval	GSM8K+	ARC-E
Contriever	75.49	36.92	**33.63**	**24.11**	**77.53**
DPR	74.47	36.84	32.87	23.43	77.48
BM25	**75.55**	**37.11**	31.44	23.57	77.10

Table 2: Results for different retrievers.

# Knowledge	BBH	BB	AGIEval	GSM8K+	ARC-E
$m = 1$	**75.49**	**36.92**	**33.63**	**24.11**	**77.53**
$m = 2$	72.41	35.36	32.57	21.28	77.48
$m = 3$	73.28	34.71	31.83	20.92	76.98

Table 3: Results for different m.

4.3 Implementations

For all experiments, we use the LLaMA2-7B (Touvron et al. 2023) and TinyLLaMA-1.1B (Zhang et al. 2024a) as the student SLM. We query the teacher model GPT-3.5-turbo to annotate the CoTs data with the manual prompt (Suzgun et al. 2022). Unless otherwise specified, m is set to 1 (§4.7) and $\Delta_{\text{threshold}}$ is set to 0.68 (§4.8). We follow the standard metrics and datasets statics elaborated in Appendix A.1.

We employ LoRA (Hu et al. 2022) for parameter-efficient fine-tuning of the student SLMs. All experiments are conducted on 2 A100 GPUs with 80GB. During the inference stage, we utilize vLLM (Kwon et al. 2023) to accelerate inference. Detailed information about training and hyperparameters is provided in Appendix A.2.

4.4 Main Results

Table 1 shows that the NesyCD has **achieved significant improvements on both ID and OOD datasets** using two weaker SLMs[2]. Specifically, LLaMA2-7B and TinyLLaMA-1.1B demonstrated an average improvement of 8.4% and 5.9% respectively, consistently outperforming all existing baselines. For an analysis of model size, please refer to §4.5. The impact of NesyCD decreases as the model size (and hence capability) increases because larger models can retain knowledge better during pre-training and fine-tuning.

Compared to Zeroshot and Zeroshot-CoT, **CoT distillation has significantly improved the performance of SLM**. While fine-tuning methods can significantly enhance factual and general reasoning abilities, **fine-tuning's impact on mathematical reasoning remains minimal**. CoT distillation aids SLM in generating rationales that clarify intermediate steps, thereby enhancing mathematical reasoning capabilities. Among various CoT distillation methods, NesyCD not only boosts symbolic knowledge learning to rectify SLM errors but also adaptively retrieves specialized knowledge based on confidence during tests to assist in producing high-quality rationales, thus improving overall performance. In comparison to KARD (Kang et al. 2023), **our specialized KB is more relevant to the capabilities of the SLM and the question at hand**. Additionally, it generates a distribution more aligned with the SLM through the LLM, significantly enhancing effectiveness (Yu et al. 2023b). Our NesyCD significantly enhances the SLM's performance, demonstrating the effectiveness of neural-symbolic knowledge integration in complex reasoning tasks.

4.5 Efficiency on Dataset and Model Sizes

To evaluate the efficiency of NesyCD in terms of training data and model size, we measured test accuracy using Qwen2's (Yang et al. 2024) 0.5B, 1.5B, and 7B models across various methods while varying the amount of training data and model size. As shown at the bottom of Figure 3, NesyCD successfully transfers the reasoning capabilities of the teacher LLM by generating symbolic specialized knowledge, even with minimal training data. As the training data decreases, the performance gap between NesyCD and other baselines widens, demonstrating NesyCD's superior robustness and sample efficiency. **This suggests that NesyCD performs better with fewer samples and that its effectiveness can be further enhanced by increasing the**

[2]More results about LLaMA3 and Qwen2 are in Appendix D.

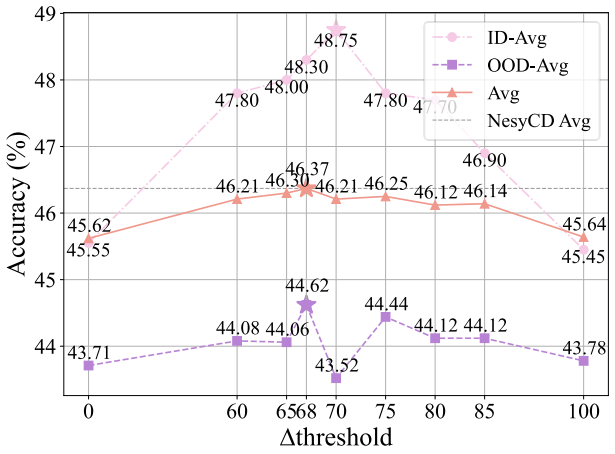

Figure 4: Performance variation trend on $\Delta_{\text{threshold}}$. The results are reported by ID-Avg and OOD-Avg which respectively denote average accuracy on ID and OOD datasets.

training data, allowing for more effective distillation.

Regarding model size efficiency, as shown at the top of Figure 3, NesyCD outperforms other baselines across various model scales. Notably, NesyCD enables Qwen2-7B to surpass the teacher GPT-3.5 Turbo in both ID and OOD performance, despite having over $10\times$ fewer parameters. These results highlight NesyCD's significant practical benefits in resource-constrained environments, as it reduces the computational cost required for SLMs while achieving performance levels that exceed those of larger LLMs. **This further illustrates that SLMs cannot fully utilize the CoT reasoning generated by LLMs, thereby necessitating the implementation of our proposed NesyCD**.

4.6 Performances with Different Retrievers

We examined the impact of various retrievers including DPR (Karpukhin et al. 2020), Contriever (Izacard et al. 2021), and BM25 (Robertson and Zaragoza 2009) on the performance of NesyCD using LLaMA2-7B. As shown in Table 2, the performance differences among these retrievers are minimal, with Contriever performing slightly better. This finding suggests that NesyCD can attain improved benefits as the quality of the retriever advances, effectively retrieving more relevant specialized knowledge and enhancing the capability to solve complex reasoning tasks.

4.7 The Number of Knowledge Used for Inference

Even LLMs can be easily distracted by irrelevant background information (Shi et al. 2023) or extended context (Liu et al. 2023). Therefore, simply adding more knowledge during the inference process does not necessarily enhance performance unless the relevant knowledge is selected. Table 3 illustrates the impact of the number of knowledge used during inference in the NesyCD (m in §4.7) on LLaMA2-7B. We observe that performance decreases as m increases which implies that including additional knowledge does not always enhance reasoning and can interfere with the model's judgment, corroborating previous research findings.

Methods	BBH	BB	AGIEval	GSM8K+	ARC-E
NesyCD	**75.49**	**36.92**	**33.63**	**24.11**	**77.53**
w/o $\boldsymbol{k}^m$	68.48	35.32	31.75	20.93	76.86
w/o $\boldsymbol{k}^p$	68.86	35.64	31.59	22.35	77.06
w/o AD	64.34	32.23	28.85	20.43	74.79
w/o AP & DC	66.51	35.12	29.83	21.88	77.41
Std-CoT	58.13	30.68	23.61	12.02	73.36
w $\boldsymbol{k}$	61.24	31.33	26.41	13.75	74.32
Zero-shot-CoT	13.51	12.19	10.32	2.08	29.13
w $\boldsymbol{k}$	14.57	13.23	11.68	2.92	30.68

Table 4: Ablation studies on different components.

4.8 Impact of Confidence Threshold

We investigated the effect of varying the confidence threshold $\Delta_{\text{threshold}}$ on performance across both ID and OOD datasets using LLaMA2-7B, as shown in Figure 4. A higher confidence threshold means the model requires greater certainty to trust its output. In extreme cases, $\Delta_{\text{threshold}} = 0$ means no retrieval is performed, while $\Delta_{\text{threshold}} = 100$ means retrieval is performed for all cases. Both extremes lead to significant performance degradation, underscoring the need for adaptive retrieval. For easy tasks, the model might produce hallucinations and incorrect answers with additional knowledge, while for complex tasks lacking external guidance, the model cannot rely solely on internal parameters. Despite these challenges, our method consistently outperforms other baselines, even in extreme scenarios.

4.9 Ablation Studies

To demonstrate the effectiveness of NesyCD, we created four variants by individually removing the learning summary $\boldsymbol{k}^m$, the supplementary knowledge $\boldsymbol{k}^p$, the augmented distillation (AD), and the multi-task learning (AP & DC, §3.4) respectively. Specialized knowledge $\boldsymbol{k}$ is composed of $\boldsymbol{k}^m$ and $\boldsymbol{k}^p$ (§3.2). We employed LLaMA2-7B as the SLM for ablation studies, and the results are presented in Table 4. We can observe that performance diminishes with the exclusion of any single component, underscoring the significance of each element. Additionally, specialized knowledge exhibits orthogonality and universality, enhancing Zero-shot-CoT and other CoT distillation methods (w $\boldsymbol{k}$) which confirms the importance of refining symbolic knowledge.

5 Conclusion

In this work, we introduce Neural-Symbolic Collaborative Distillation (NesyCD), a method aimed at enhancing the capabilities of Small Language Models (SLMs) for complex reasoning tasks that require additional knowledge and advanced reasoning skills. NesyCD uses Large Language Models (LLMs) to analyze SLM errors and generate specialized knowledge, including learning summaries and supplementary knowledge, organized into an external knowledge base. By integrating parameter updates with retrieving specialized knowledge, NesyCD improves both rationale generation and answer accuracy for SLMs. Empirical experiments show that NesyCD surpasses fine-tuning and CoT distillation baselines in in- and out-of-domain scenarios.

Acknowledgements

This work was supported by the National Key R&D Program of China (No. 2022ZD0160503) and the National Natural Science Foundation of China (No.62376270, No.62276264).

References

Chen, D.; Fisch, A.; Weston, J.; and Bordes, A. 2017. Reading Wikipedia to Answer Open-Domain Questions. In Barzilay, R.; and Kan, M.-Y., eds., *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1870–1879. Vancouver, Canada: Association for Computational Linguistics.

Clark, P.; Cowhey, I.; Etzioni, O.; Khot, T.; Sabharwal, A.; Schoenick, C.; and Tafjord, O. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *ArXiv*, abs/1803.05457.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*.

Dai, C.; Li, K.; Zhou, W.; and Hu, S. 2024. Improve Student's Reasoning Generalizability through Cascading Decomposed CoTs Distillation. *arXiv preprint arXiv:2405.19842*.

Fu, Y.; Peng, H.-C.; Ou, L.; Sabharwal, A.; and Khot, T. 2023. Specializing Smaller Language Models towards Multi-Step Reasoning. *ArXiv*, abs/2301.12726.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2021a. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021b. Measuring Mathematical Problem Solving With the MATH Dataset. *NeurIPS*.

Ho, N.; Schmid, L.; and Yun, S.-Y. 2022. Large Language Models Are Reasoning Teachers. In *Annual Meeting of the Association for Computational Linguistics*.

Hsieh, C.-Y.; Li, C.-L.; Yeh, C.-K.; Nakhost, H.; Fujii, Y.; Ratner, A.; Krishna, R.; Lee, C.-Y.; and Pfister, T. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Hu, E. J.; yelong shen; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; and Grave, E. 2021. Unsupervised Dense Information Retrieval with Contrastive Learning.

Kang, M.; Lee, S.; Baek, J.; Kawaguchi, K.; and Hwang, S. J. 2023. Knowledge-Augmented Reasoning Distillation for Small Language Models in Knowledge-Intensive Tasks. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, December 10-16, 2023, New Orleans*.

Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T. B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; and Amodei, D. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Karpukhin, V.; Oğuz, B.; Min, S.; Lewis, P.; Wu, L. Y.; Edunov, S.; Chen, D.; and tau Yih, W. 2020. Dense Passage Retrieval for Open-Domain Question Answering. *ArXiv*, abs/2004.04906.

Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; and Iwasawa, Y. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213.

Kwon, W.; Li, Z.; Zhuang, S.; Sheng, Y.; Zheng, L.; Yu, C. H.; Gonzalez, J. E.; Zhang, H.; and Stoica, I. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Lee, N.; Wattanawong, T.; Kim, S.; Mangalam, K.; Shen, S.; Anumanchipali, G.; Mahoney, M. W.; Keutzer, K.; and Gholami, A. 2024. LLM2LLM: Boosting LLMs with Novel Iterative Data Enhancement. *arXiv*.

Li, Q.; Cui, L.; Zhao, X.; Kong, L.; and Bi, W. 2024a. GSM-Plus: A Comprehensive Benchmark for Evaluating the Robustness of LLMs as Mathematical Problem Solvers. *ArXiv*, abs/2402.19255.

Li, S.; Chen, J.; yelong shen; Chen, Z.; Zhang, X.; Li, Z.; Wang, H.; Qian, J.; Peng, B.; Mao, Y.; Chen, W.; and Yan, X. 2024b. Explanations from Large Language Models Make Small Reasoners Better. In *2nd Workshop on Sustainable AI*.

Li, Y.; Yuan, P.; Feng, S.; Pan, B.; Sun, B.; Wang, X.; Wang, H.; and Li, K. 2023. Turning Dust into Gold: Distilling Complex Reasoning Capabilities from LLMs by Leveraging Negative Data. In *AAAI Conference on Artificial Intelligence*.

Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; and Liang, P. 2023. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12: 157–173.

Magister, L. C.; Mallinson, J.; Adamek, J.; Malmi, E.; and Severyn, A. 2023. Teaching Small Language Models to Reason. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 1773–1781. Toronto, Canada: Association for Computational Linguistics.

Meta. 2024. Introducing Meta Llama 3: The most capable openly available LLM to date. https://ai.meta.com/blog/meta-llama-3/. Accessed: 2024-04.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Köpf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.;

and Chintala, S. 2019. *PyTorch: an imperative style, high-performance deep learning library*. Red Hook, NY, USA: Curran Associates Inc.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In Su, J.; Duh, K.; and Carreras, X., eds., *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2383–2392. Austin, Texas: Association for Computational Linguistics.

Robertson, S.; and Zaragoza, H. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.*, 3(4): 333–389.

Shi, F.; Chen, X.; Misra, K.; Scales, N.; Dohan, D.; hsin Chi, E. H.; Scharli, N.; and Zhou, D. 2023. Large Language Models Can Be Easily Distracted by Irrelevant Context. In *International Conference on Machine Learning*.

Srivastava, A.; Rastogi, A.; Rao, A.; and so on. 2022. Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models. *ArXiv*, abs/2206.04615.

Suzgun, M.; Scales, N.; Scharli, N.; Gehrmann, S.; Tay, Y.; Chung, H. W.; Chowdhery, A.; Le, Q. V.; hsin Chi, E. H.; Zhou, D.; and Wei, J. 2022. Challenging BIG-Bench Tasks and Whether Chain-of-Thought Can Solve Them. In *Annual Meeting of the Association for Computational Linguistics*.

Tong, Y.; Li, D.; Wang, S.; Wang, Y.; Teng, F.; and Shang, J. 2024. Can LLMs Learn from Previous Mistakes? Investigating LLMs' Errors to Boost for Reasoning. *ArXiv*, abs/2403.20046.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, R.; Li, H.; Han, X.; Zhang, Y.; and Baldwin, T. 2024. Learning From Failure: Integrating Negative Examples when Fine-tuning Large Language Models as Agents. *ArXiv*, abs/2402.11651.

Wang, X.; and Zhou, D. 2024. Chain-of-Thought Reasoning Without Prompting. *ArXiv*, abs/2402.10200.

Wang, Y.; Ivison, H.; Dasigi, P.; Hessel, J.; Khot, T.; Chandu, K.; Wadden, D.; MacMillan, K.; Smith, N. A.; Beltagy, I.; and Hajishirzi, H. 2023a. How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Wang, Y.; Li, P.; Sun, M.; and Liu, Y. 2023b. Self-Knowledge Guided Retrieval Augmentation for Large Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 10303–10315. Singapore: Association for Computational Linguistics.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; Davison, J.; Shleifer, S.; von Platen, P.; Ma, C.; Jernite, Y.; Plu, J.; Xu, C.; Le Scao, T.; Gugger, S.; Drame, M.; Lhoest, Q.; and Rush, A. 2020. Transformers: State-of-the-Art Natural Language Processing. In Liu, Q.; and Schlangen, D., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. Online: Association for Computational Linguistics.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; Huang, F.; Dong, G.; Wei, H.; Lin, H.; Tang, J.; Wang, J.; Yang, J.; Tu, J.; Zhang, J.; Ma, J.; Xu, J.; Zhou, J.; Bai, J.; He, J.; Lin, J.; Dang, K.; Lu, K.; Chen, K.-Y.; Yang, K.; Li, M.; Xue, M.; Ni, N.; Zhang, P.; Wang, P.; Peng, R.; Men, R.; Gao, R.; Lin, R.; Wang, S.; Bai, S.; Tan, S.; Zhu, T.; Li, T.; Liu, T.; Ge, W.; Deng, X.; Zhou, X.; Ren, X.; Zhang, X.; Wei, X.; Ren, X.; Fan, Y.; Yao, Y.; Zhang, Y.; Wan, Y.; Chu, Y.; Cui, Z.; Zhang, Z.; and Fan, Z.-W. 2024. Qwen2 Technical Report. *ArXiv*.

Ye, J.; Wu, Z.; Feng, J.; Yu, T.; and Kong, L. 2023. Compositional exemplars for in-context learning. In *International Conference on Machine Learning*, 39818–39833. PMLR.

Ying, J.; Lin, M.; Cao, Y.; Tang, W.; Wang, B.; Sun, Q.; Huang, X.; and Yan, S. 2024. LLMs-as-Instructors: Learning from Errors Toward Automating Model Improvement. *arXiv preprint arXiv:2407.00497*.

Yu, L.; Jiang, W.; Shi, H.; Yu, J.; Liu, Z.; Zhang, Y.; Kwok, J. T.; Li, Z.; Weller, A.; and Liu, W. 2023a. MetaMath: Bootstrap Your Own Mathematical Questions for Large Language Models. *arXiv preprint arXiv:2309.12284*.

Yu, W.; Iter, D.; Wang, S.; Xu, Y.; Ju, M.; Sanyal, S.; Zhu, C.; Zeng, M.; and Jiang, M. 2023b. Generate rather than retrieve: Large language models are strong context generators. In *International Conference for Learning Representation (ICLR)*.

Zhang, P.; Zeng, G.; Wang, T.; and Lu, W. 2024a. TinyLlama: An Open-Source Small Language Model. *ArXiv*, abs/2401.02385.

Zhang, T.; Madaan, A.; Gao, L.; Zhang, S.; Mishra, S.; Yang, Y.; Tandon, N.; and Alon, U. 2024b. In-Context Principle Learning from Mistakes. In *ICML 2024 Workshop on In-Context Learning*.

Zhao, Z.; Lee, W. S.; and Hsu, D. 2023. Large Language Models as Commonsense Knowledge for Large-Scale Task Planning. In *RSS 2023 Workshop on Learning for Task and Motion Planning*.

Zheng, Y.; Zhang, R.; Zhang, J.; Ye, Y.; Luo, Z.; Feng, Z.; and Ma, Y. 2024. LlamaFactory: Unified Efficient Fine-Tuning of 100+ Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*. Bangkok, Thailand: Association for Computational Linguistics.

Zhong, W.; Cui, R.; Guo, Y.; Liang, Y.; Lu, S.; Wang, Y.; Saied, A.; Chen, W.; and Duan, N. 2023. Agieval: A human-centric benchmark for evaluating foundation models. *arXiv preprint arXiv:2304.06364*.