QCRD: Quality-guided Contrastive Rationale Distillation for Large Language Models

Wei Wang[1,2], **Zhaowei Li**[2,3], **Qi Xu**[2], **Yiqing Cai**[2], **Hang Song**[2], **Qi Qi**[2], **Ran Zhou**[2], **Zhida Huang**[2], **Tao Wang**[2], **Li Xiao**[1]
[1]MoE Key Laboratory of Brain-Inspired Intelligence Perception and Cognition
University of Science and Technology of China, [2]ByteDance Inc, [3]Fudan University

Abstract

The deployment of large language models (LLMs) faces considerable challenges concerning resource constraints and inference efficiency. Recent research has increasingly focused on smaller, task-specific models enhanced by distilling knowledge from LLMs. However, prior studies have often overlooked the diversity and quality of knowledge, especially the untapped potential of negative knowledge. Constructing effective negative knowledge remains severely understudied. In this paper, we introduce a novel framework called quality-guided contrastive rationale distillation aimed at enhancing reasoning capabilities through contrastive knowledge learning. For positive knowledge, we enrich its diversity through temperature sampling and employ self-consistency for further denoising and refinement. For negative knowledge, we propose an innovative self-adversarial approach that generates low-quality rationales by sampling previous iterations of smaller language models, embracing the idea that one can learn from one's own weaknesses. A contrastive loss is developed to distill both positive and negative knowledge into smaller language models, where an online-updating discriminator is integrated to assess qualities of rationales and assign them appropriate weights, optimizing the training process. Through extensive experiments across multiple reasoning tasks, we demonstrate that our method consistently outperforms existing distillation techniques, yielding higher-quality rationales. The code will be released in https://github.com/wwangweii/QCRD_example.git.

1 Introduction

The reasoning capabilities of large language models (LLMs) have been observed to scale their model sizes, while necessitating substantial memory and computing resources (Chowdhery et al., 2023; Wei et al., 2022a). As such, efficient model compression is crucial in the deployment of LLMs, especially on resource-limited devices or platforms. Knowledge distillation from an LLM (teacher) to a smaller, more manageable language model (student) has recently emerged as a powerful and promising technique for model compression (Hinton et al., 2015; Phuong and Lampert, 2019). However, it is still open how to best reduce the performance gap between the teacher and the student on complex reasoning tasks (Zelikman et al., 2022).

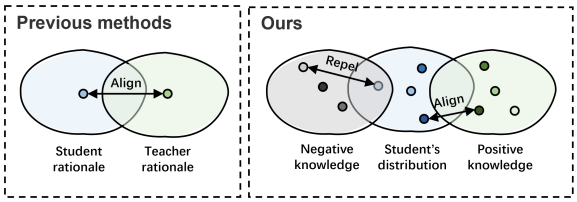

Figure 1: Comparison between previous methods and our proposed method, where circle points denote rationales, and colors of the circle points correspond to rationale types, and shades of darker indicate higher qualities. The "align" means minimizing the distance between rationales, while the "repel" means maximizing the distance.

In this regard, it has more recently been shown that adding explanation-augmented prompts, especially, Chain-of-Thought (CoT) (Wei et al., 2022b), can enable LLMs to generate reasonable explanations (also referred to as rationales) to justify the reasoning outcomes (Li et al., 2022). Distilling these rationales into smaller language models has been demonstrated to effectively improve the overall performance (Hsieh et al., 2023; Li et al., 2022). For example, distilling Step-by-Step (DSS) (Hsieh et al., 2023) was proposed as an innovative CoT distillation approach, which employed rationales from an LLM to guide a smaller language model under a multi-task learning setting. It involved training the smaller language model simultaneously on both label prediction and rationale generation tasks, effectively leveraging their mutual benefits.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 14334–14345
November 4-9, 2025

The essence of such distilling rationales is to guide the model in learning additional knowledge related to the labels. Knowledge can be generally concluded into two classes: positive and negative. Previous works on rationale distillation, although effective, still suffer from certain drawbacks. On the one hand, positive knowledge for distillation may be limited and noisy. Methods (Fu et al., 2023; Hsieh et al., 2023; Magister et al., 2022; Chen et al., 2024b) treated rationales generated by LLMs as golden answers and aimed to minimize the gap between these rationales and those generated by smaller language models. However, despite LLMs' powerful zero-shot/few-shot abilities, they may occasionally produce incorrect reasoning steps, leading to erroneous rationales/answers. Such erroneous rationales may degrade the reasoning performance of the distilled smaller language models. On the other hand, generating negative rationales and incorporating them into CoT distillation remain understudied, while negative knowledge has early proved constructive and effective for models.

To this end, we propose a general method, named Quality-guided Contrastive Rationale Distillation (QCRD), to guide the knowledge distillation to smaller language models from a contrastive learning perspective. The comparison between the previous methods and our proposed QCRD is illustrated in Fig. 1. Specifically, the previous methods focus on the alignment between the rationale of the student model and the corresponding one of the teacher model, while our proposed QCRD aligns the student's distribution and contrastive knowledge distribution with various sampled rationales. The core design of QCRD is to generate a diverse set of contrastive rationales and efficiently distill them into student models. For the positive part, to ensure the quality and variety of positive rationales, we prompt the LLM and sample the output to generate multi-round rationales for each input question. We then apply the self-consistency to denoise the rationale set and split it into positive rationales and negative rationales. For the negative part, we employ a self-adversarial strategy inspired by (Silver et al., 2018) during training to generate low-quality rationales from previous iterations of smaller language models with a high sampling temperature and treat them as negative rationales. Finally, for better knowledge learning, we present a contrastive loss to distill both positive and negative rationales into smaller language models. A discriminator is adopted to assess the qualities of the rationales and assign them appropriate weights to optimize the training process across the datasets.

To demonstrate the superiority of QCRD, we conduct comprehensive experiments with two smaller types of T5 models (Raffel et al., 2020), i.e., T5-base (220M parameters) and T5-small (60M parameters), on four popular datasets, followed by detailed analysis and discussion. Our main contributions of this paper can be summarized below.

- We first develop a general CoT distillation approach (i.e., QCRD) from a contrastive learning perspective, aiming to guide the student model to learn both positive and negative knowledge from rationales.
- We explore a contrastive distillation loss to facilitate effective distillation of the generated positive and negative rationales, where the qualities of the rationales judged by a discriminator are considered to optimize the training process across the whole datasets.
- Experimental results across multiple datasets show that QCRD outperforms existing methods and can be widely applied, demonstrating its efficiency in utilizing contrastive reasoning knowledge for smaller language models.

2 Related Work

Knowledge distillation from LLMs. Knowledge distillation (KD) is a highly effective technique for transferring knowledge from larger teacher models to smaller student models that are more suitable for practical applications (Fu et al., 2023; Hsieh et al., 2023; Magister et al., 2022; Chen et al., 2024b; Wang et al., 2023). The KD technique can be generally classified into two different categories: (1) Black-box KD: only the teacher's predictions are accessible; (2) White-box KD: it provides access to the teacher's parameters. Both of them have shown promising potential in fine-tuning smaller models on the prompt response pairs generated by LLMs (Zhu et al., 2023). In this paper, we hypothesize that only the predictions (predict labels and rationales) generated by LLMs are accessible.

Multi-task learning with LLM generated rationales. Current LLMs have already exhibited their capabilities to generate high-quality reasoning steps, resulting in rationales of their predictions (Kojima et al., 2022), and these rationales have been found to be valuable additional knowledge for fine-tuning smaller models (Hsieh et al., 2023). A multi-task learning framework is com-

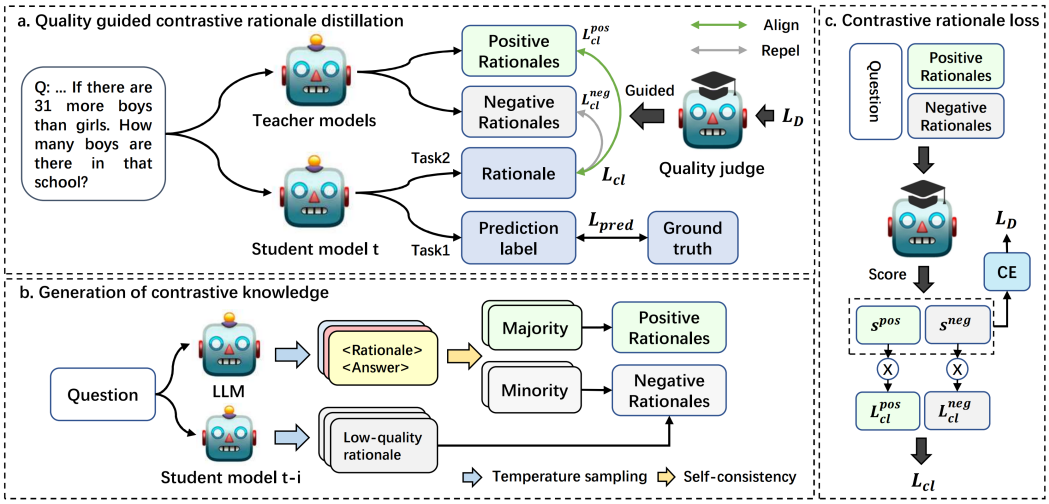

Figure 2: Illustration of the proposed quality-guided contrastive rationale distillation for distilling contrastive knowledge from teacher models into the student model. Fig.a represents our multi-task framework, i.e., the main prediction label task and additional rationale task. Fig.b represents generation of contrastive rationales for distillation. Fig.c represents details about the quality-guided contrastive rationale loss, and CE denotes the cross-entropy.

monly employed that enforces smaller models to output corresponding rationales, while maintaining their original functionality. However, previous studies only focused on aligning the output of the smaller model with that of the LLM with a single loss form (Hsieh et al., 2023; Magister et al., 2022).

Self-consistency of LLMs. The self-consistency of LLMs refers to the capacity to maintain coherent and rational reasoning during input processing. Based on the intuition that complex reasoning tasks typically admit multiple reasoning paths that reach a correct answer, the self-consistency can improve the LLMs' reasoning performance by integratedly sampling CoT outputs several times and choosing the most consistent predict answer (Stanovich and West, 1991; Wang et al., 2022).

Contrastive learning for LLMs. Contrastive learning has demonstrated its efficiency across diverse domains, e.g., computer vision, natural language processing (Jaiswal et al., 2020; Le-Khac et al., 2020). Notably, the application of contrastive learning to LLMs has recently emerged, highlighting the effectiveness of incorporating negative knowledge implicitly in model's inputs and showing promising outcomes (Li et al., 2024; Chen et al., 2024a). However, to the best of our knowledge, the application of contrastive learning in CoT rationale distillation has not been explored thus far.

3 Methodology

In this paper, we first propose a general contrastive CoT distillation approach, called quality-guided contrastive rationale distillation (QCRD), for training smaller models by distilling contrastive knowledge from teacher models. As illustrated in Fig. 2, our approach consists of the following three parts. (1) Following the method developed in (Hsieh et al., 2023), we apply a multi-task learning framework for the supervised training of the student model, i.e., the main prediction label task and additional rationale generation task; see Fig. 2a. (2) As displayed in Fig. 2b, we design a general approach to generate contrastive knowledge from LLMs and student model itself for rationale distillation. (3) As shown in Fig. 2c, for better knowledge learning from rationales, we design a quality-guided contrastive learning strategy, where a contrastive loss is applied with the guidance of an online-updated discriminator to distinguish between positive and negative rationales and assign them quality scores.

3.1 Multi-task learning framework for the student model

Previous works have already demonstrated the advantages of the multi-task learning framework (Fu et al., 2023; Hsieh et al., 2023; Magister et al., 2022). Accordingly, as shown in Fig. 2a, we apply the label prediction task and the rationale generation task to the training of smaller language models. Specifically, we use different prefixes to enforce smaller language models to generate different types of output. Given an input question, for the label prediction task, the smaller language model outputs the prediction label with input prefix $< Predict >$,

while for the rationale generation task, it outputs the corresponding explanation with input prefix $< Explain >$. These outputs are then aligned to the corresponding ground truth and rationales using autoregressive loss, respectively.

3.2 Generation of contrastive knowledge

We use CoT prompting (Wei et al., 2022b) to elicit and extract rationales from LLMs. As illustrated in Fig. 3, the LLM is provided with few examples to follow the output format. Instead of only generating one output for each input, we replace the "greedy decode" in CoT prompting with sampling from the language model's decoder to generate a diverse set of reasoning paths (Wang et al., 2022). We apply temperature sampling (Renze and Guven, 2024) to the LLM K times, where a temperature value τ can control the diversity of the generated output. Therefore, for each input, there are K pairs of rationales and corresponding labels.

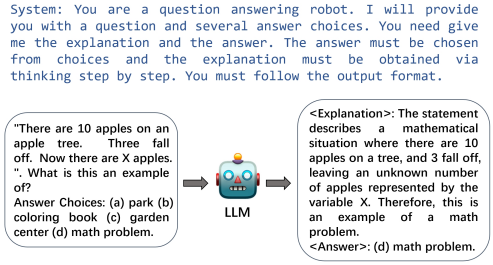

Figure 3: A case of the prompt and rationale output.

3.2.1 Build positive and negative rationale sets

Language models are not infallible reasoners; they can produce incorrect reasoning paths or mistakes in individual steps. Research indicates that correct reasoning processes, despite their diversity, generally yield more consistent final answers than incorrect ones (Wang et al., 2022). Thus, we select the rationales with the most consistent labels across all outputs of the LLM as positive rationales, while the remaining ones are classified as negative rationales. The number of sampling times primarily affects the ratio of positive to negative samples (with negative samples typically being fewer, as detailed in Appendix A.1), as well as the associated time and storage costs. Moreover, negative rationales from LLMs are likely to be positive for smaller language models, which may limit their effectiveness.

To deal with this issue, we conduct a self-adversarial mechanism that the student model generates its own negative rationales by sampling from its previous iterations with a high temperature value during training (we illustrate its rationality in Section 5.3 and demonstrate its superiority in Appendix A.2), and we regard these low-quality rationales as negative ones based on the hypothesis that the rationale quality of LLMs is higher than that of smaller models. As a result, for each input question $\mathbf{x} = [x_1, x_2, ..., x_n]$, we collect a positive rationale set $S_{pos} = \{\mathbf{r}_1^{pos}, \mathbf{r}_2^{pos}, ..., \mathbf{r}_m^{pos}\}$ and a negative rationale set $S_{neg} = \{\mathbf{r}_1^{neg}, \mathbf{r}_2^{neg}, ..., \mathbf{r}_k^{neg}\}$.

3.3 Contrastive knowledge distillation

In this subsection, we present our designed quality-guided contrastive rationale distillation for better knowledge learning.

3.3.1 Train a discriminator to judge rationales

The quality of rationales for the same question still differs. Moreover, as the training epoch increases, the rationales generated by the above self-play may become gradually closer to the positive rationales, and then viewing them as negative ones is no longer reasonable. Therefore, there is a need to train a discriminator $\mathcal{D}$ that can effectively judge the positive and negative rationales and output a score that represents the quality of each rationale. The input of the discriminator $\mathcal{D}$ is the question and the rationale, and we take an encoder architecture to measure the score, i.e.,

$$s_j^{pos} = \mathcal{D}\left(\mathbf{x}, \mathbf{r}_j^{pos}\right) \quad \text{or} \quad s_j^{neg} = \mathcal{D}\left(\mathbf{x}, \mathbf{r}_j^{neg}\right). \tag{1}$$

We pretrain the $\mathcal{D}$ with the positive and negative rationales from the LLM, and during training, the discriminator $\mathcal{D}$ is updated at regular epoch intervals (details can be seen in Appendix A.3). The loss function can be formulated as

$$\mathcal{L}_{\mathcal{D}} = \mathbb{E}_{\mathbf{x}}\left[-\log \frac{\sum_{j=1}^{m} \exp(s_j^{pos})}{\sum_{j=1}^{k} \exp(s_j^{neg})}\right]. \tag{2}$$

3.3.2 Quality-guided contrastive distillation

As mentioned in sec. 3.2.1, there is a diverse set of positive rationales. In addition, the negative rationales are of significance, which can enforce the smaller model away from their distribution. Since some of the negative samples are generated by the previous-iteration smaller model, the smaller model can further refine its reasoning capability through playing against instances of itself and promote the generated rationales closer to golden rationales of the LLM. Therefore, we propose a many-to-one contrastive distillation loss, while previous

studies typically utilize a single rationale for each question and distill it into the smaller model, i.e.,

$$\mathcal{L}_{cl} = \frac{1}{N}\sum_{i=1}^{N}[l(f(\mathbf{x}_i), S^i_{pos}) - \beta \cdot l(f(\mathbf{x}_i), S^i_{neg})], \quad (3)$$

where $\mathbf{x}_i$ denotes the i-th question, S^i_{pos} and S^i_{neg} denote the corresponding positive and negative rationale sets, respectively, N is the number of questions, and $\beta > 0$ is a tunable hyper-parameter. The function $l(\cdot)$ denotes the cross-entropy loss and $f(\cdot)$ denotes the rationale generation for its given input. In (3),

$$l(f(\mathbf{x}_i), S^i_{pos}) = \min_{\mathbf{r}^{pos,i}_j \in S^i_{pos}} \left\{ l\left(f(\mathbf{x}_i), \mathbf{r}^{pos,i}_j\right)\right\}, \quad (4)$$

$$l(f(\mathbf{x}_i), S^i_{neg}) = \max_{\mathbf{r}^{neg,i}_j \in S^i_{neg}} \left\{ l\left(f(\mathbf{x}_i), \mathbf{r}^{neg,i}_j\right)\right\}, \quad (5)$$

which are designed to learn both the most relevant positive knowledge and the least-disturbed negative knowledge from teacher models. Moreover, we set a margin δ for the negative rationales to filter out cases that are too simplistic, i.e., $l(f(\mathbf{x}), \mathbf{r}^{neg}_j) = \min(l(f(\mathbf{x}), \mathbf{r}^{neg}_j) - \delta, 0)$ with respect to the j-th negative rationale for an input question $\mathbf{x}$. Let us rethink the effectiveness of negative rationales generated by the previous-iteration smaller model, which enforces the smaller model to break out of local optima and yield a golden rationale that is closer to the output of the LLM. However, when the smaller model comes to converging, the previous-iteration smaller model is likely to output the rationales that are similar to those of the LLM, and then regarding them as negative samples is inaccurate. To address this issue, we introduce the quality-guided distillation to optimize the training process and redefine the loss formulas in (4) and (5) as, respectively,

$$l(f(\mathbf{x}_i), S^i_{pos}) = s^{pos,i} \cdot \min_{\mathbf{r}^{pos,i}_j \in S^i_{pos}} \left\{ l\left(f(\mathbf{x}_i), \mathbf{r}^{pos,i}_j\right)\right\}, \quad (6)$$

$$l(f(\mathbf{x}_i), S^i_{neg}) = (1-s^{neg,i}) \cdot \max_{\mathbf{r}^{neg,i}_j \in S^i_{neg}} \left\{ l\left(f(\mathbf{x}_i), \mathbf{r}^{neg,i}_j\right)\right\}, \quad (7)$$

where $s^{pos,i}$ and $s^{neg,i}$ are the corresponding quality scores obtained by the discriminator $\mathcal{D}$. By (6) and (7), the positive rationales of higher quality should have larger weights across the datasets, while for the negative rationales of higher quality, it is on the contradiction. In the latter sec. 5.1, we will further discuss different schemes for the many-to-one distillation.

3.3.3 Training loss

The final training loss is given by

$$\mathcal{L}_{total} = \alpha_1 \mathcal{L}_{pred} + \alpha_2 \mathcal{L}_{cl} + \alpha_3 \mathcal{L}_{\mathcal{D}}, \quad (8)$$

where $\{\alpha_i\}_{i=1}^3 > 0$ are tunable hyper-parameters, $\mathcal{L}_{pred}$ represents the cross entropy loss of the label prediction task, $\mathcal{L}_{cl}$ is the many-to-one contrastive distillation loss in (3), and $\mathcal{L}_{\mathcal{D}}$ is the discriminator loss in (2).

4 Experiments

4.1 Experimental setting

Datasets. We conducted extensive experiments on four widely-used benchmark datasets (see details in Appendix Table 1) across three different natural language processing tasks, including SVAMP (Patel et al., 2021) for arithmetic word problem solving, CQA (Talmor et al., 2018) for commonsense question answering, as well as e-SNLI (Camburu et al., 2018) and ANLI (Nie et al., 2019) for natural language inference. The rationales we used were generated by GPT-3.5-turbo[1] and an opened code source by (Hsieh et al., 2023) was referred.

Implementation details. Following the properties of CoT and the comparative experimental studies in (Hsieh et al., 2023; Chen et al., 2024b), our QCRD utilized T5-base (220M parameters) and T5-small (60M parameters) as the student models to ensure a fair comparison. α_1, α_2, α_3 were set to 0.5 empirically. α_3 was multiplied by 0.9 per iteration. We set $\beta = 0.2$ and $\delta = 3$. We sampled the LLM's output 5 times with the temperature being 0.7, and sampled 5-iteration-before models with the temperature being 1.5. The batchsize was 8 and learning rate was 5e-5. We trained our models with 10000 max steps on one A100-80G about 13 hours for T5-base and 8.5 hours for T5-small. The reported metric was accuracy.

Baselines. Four methods in learning task-specific models were compared, i.e., (1) Finetuning, which is the standard finetuning with the prevailing pretrain-then-finetune paradigm that finetunes a model with ground-truth labels via standard label supervision (Howard and Ruder, 2018); (2) Single-Task, where student models are distilled to predict labels with the teacher model's predicted labels; (3) DSS (Hsieh et al., 2023), where student models are distilled with both the predict labels and rationales of the LLM; (4) Mutual information (MI) (Chen

[1] https://platform.openai.com/docs/models

et al., 2024b), which is based on DSS and applies an additional task to maximizing the mutual information between prediction labels and rationales.

4.2 Experimental results

Experiments across four benchmarks. We conducted experiments across four benchmarks with two types of T5-model to evaluate the effectiveness of our proposed method. In the top of Table 1, we summarized the experimental results of the T5-base model distilled by our method and the baselines individually on all the four datasets. Of note, in Single-Task, the rationale and label were combined into a single sequence, which was then treated as the target during the training (Hsieh et al., 2023). It is clear that our method outperformed the baselines on most datasets, particularly when compared to the baseline DSS.

Table 1: CoT distillation results on the T5-base model.

Model	SVAMP	CQA	ANLI1	ESNLI
Finetuning	63.00	62.19	43.58	88.38
Single-Task	59.00	63.11	47.90	88.77
DSS	65.50	63.23	52.80	90.09
MI	67.50	63.50	**54.20**	90.15
Ours	**69.00**	**63.64**	54.00	**90.26**

In like manner, we performed our method and the baselines individually on the T5-small model, and their performance on all the four datasets was presented in Table 2. Our method consistently surpassed the baselines on all the four datasets.

Table 2: CoT distillation results on the T5-small model.

Model	SVAMP	CQA	ANLI1	ESNLI
Finetuning	45.00	43.16	42.00	82.90
Single-Task	46.50	44.98	42.50	83.67
DSS	48.00	45.21	42.80	84.23
MI	47.00	45.49	42.10	83.55
Ours	**50.50**	**46.11**	**44.10**	**85.30**

Distillation with LLM labels. To evaluate the impact of label qualities on CoT distillation, without loss of generality, we conducted additional experiments on the three datasets (namely, CQA, ANLI1, and ESNLI) using the T5-base model distilled by our method and DSS. Instead of using ground truth labels, we employed the labels generated by GPT-3.5-turbo to distill student models. The results were presented in Table 3. On one hand, from the top of Table 3, it demonstrates the effectiveness of temperature sampling and self-consistency (SC), which help denoise rationales and their corresponding labels. On the other hand, the results at the bottom of Table 3 indicate that our method outperformed DSS on CQA and ANLI1, even when utilizing labels generated by the LLM. Furthermore, comparing the results of the T5-base models in Table 1 with those of GPT-3.5 in Table 3, we observe that even with tiny parameters, these expert models achieve comparable performance on CQA and improved results on ESNLI.

Table 3: CoT distillation results on the T5-base model using predicted labels (noisy labels) from the LLM.

Model	CQA	ANLI1	ESNLI
GPT-3.5	66.30	78.21	66.27
GPT-3.5 with SC	69.05	80.15	67.08
DSS	59.15	44.10	**74.88**
MI	59.22	45.90	74.67
Ours	**59.80**	**46.70**	**74.88**

Distillation with smaller datasets. In addition, to demonstrate the superiority of our method on smaller datasets, we compared the performance of Finetuning and our method using T5-base models across varying sizes of each of the four datasets. Figure 4 illustrates that our method consistently achieved better performance, indicating the robustness and generality of QCRD. Notably, a more pronounced performance gain was observed on CQA when the number of training samples was limited.

Ablation study on QCRD. Compared to previous related methods, the contrastive distillation in our QCRD introduces several key enhancements as follow. (1) The extension and denoising for positive knowledge (ED): we sample the outputs of the LLM and leverage the self-consistency to denoise rationales. (2) The distillation for negative knowledge (NK): we incorporate a self-supervised mechanism to generate low-quality rationales as negative rationales. (3) The guidance of the Quality Judge (QJ): the use of discriminator helps assess rationales and optimize the training process. Additional experiments were so conducted on SVAMP to evaluate the effectiveness of each module, with the results being summarized in Table 4. The findings demonstrated that integrating more high-quality rationales significantly improved performance, while the inclusion of negative rationales proved effective. The discriminator mechanism played a positive role by considering the quality of each rationale, and we further found that the results when using the Quality Judge were more stable. We further conducted experiments to demonstrate the generalization capability of QCRD by applying it to other baseline

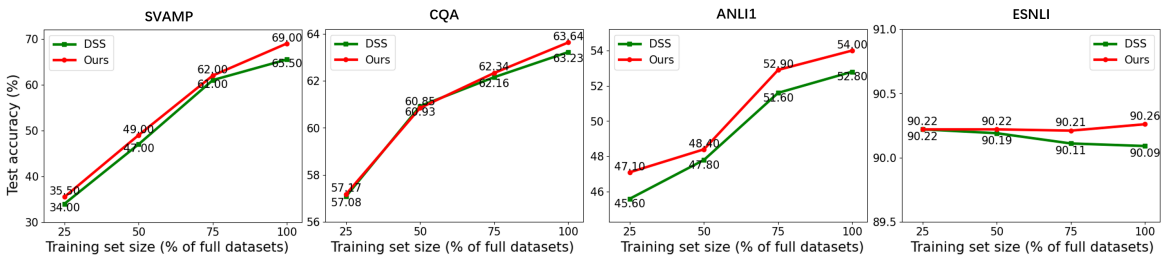

Figure 4: Comparisons with varying sizes of training datasets on the T5-base model for four benchmarks.

and larger models, as detailed in Appendix A.4.

Table 4: Ablation study on T5-base model, where ED denotes positive knowledge extension and denoising, NK denotes negative knowledge, and QJ denotes using of Quality Judge.

w/o ED	w/o NK	w/o QJ	SVAMP
✗	✗	✗	65.5
✓	✗	✗	67.0
✓	✓	✗	68.5
✓	✓	✓	69.0

5 Discussion

5.1 Different contrastive distillation schemes

In sec. 3.3.2, We defined the many-to-one distillation by taking the min loss for positive rationales and max loss for negative rationales (i.e., MinMax), which imposes a relatively weak constraint on rationale alignment. We further discuss different schemes for the many-to-one distillation. (1) MaxMin: we compute the max loss for positive rationales and min loss for negative rationales. This scheme enforces the smaller model to learn hard rationale examples. (2) Sampling: we randomly choose a positive rationale and a negative rationale for each input. (3) Mean: we average the loss for all rationales. (4) Weighted mean (W-mean): we weight the loss with quality scores and then average the loss. The results of the T5-base model distilled by our method on SVAMP were presented in Table 5 with respect to the above different schemes. One can clearly see that the MinMax achieved the best performance. Besides, the Mean scheme had a negative impact on the results. The reason may be that enforcing small models align with multi-target rationales of differences is not suitable, especially for positive knowledge.

5.2 Influence of the sampling count

In the above experiments, we sampled the output of the LLM five times and the output of iteration-before model once. We further explore the influence of the sampling count. When fixing the sampling counts for iteration-before models, results of setting different sampling counts for the LLM on SVAMP were displayed in the top of Table 6. Moreover, when fixing the sampling counts for the LLM, results on SVAMP were displayed in the bottom of Table 6 in terms of different numbers of generated negative samples. We found that sampling many negative rationales had an adverse impact on the performance, and the best performance was achieved when k was 1. Note that when $m = 1$, the performance of our method was still better than that of other related methods, again indicating the effectiveness of negative rationales.

Table 5: Results of our method with different many-to-one distillation schemes on SVAMP.

Model	MinMax	MaxMin	Sampling	Mean	W-mean
T5-base	69.0	67.0	66.5	65.0	66.0

Table 6: Results of our method on SVAMP with different sampling counts, i.e., the sampling count m for positive rationales and k for negative rationales.

Positive sample m	1	5	10	20
T5-base	67.5	69.0	68.0	68.5
negative sample k	0	1	2	3
T5-base	67.0	69.0	68.5	66.0

5.3 Rationality for negative knowledge

Temperature sampling is a commonly used decoding strategy for LLMs' generation process. By adjusting the temperature τ, we can modify the probability distribution of each word before sampling. The higher the temperature is, the smaller the difference in the probability distribution of LLM's outputs becomes, increasing the chance of sampling words with lower probabilities. In Fig 5, we provided a case of output rationales from the trained T5-base model with different temperature

settings for illustrative visualization. It validated the rationality that we generated negative rationales by sampling the iteration-before smaller models with a high temperature value. We further explored the influence of the negative sampling temperature on model performance in Appendix A.5.

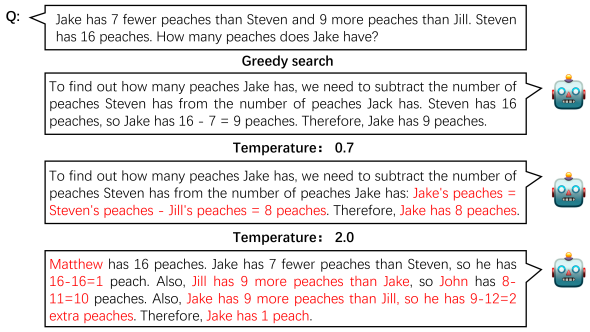

Figure 5: A case study of output rationales from the T5-base model on SVAMP at different temperatures, with incorrect details highlighted in red.

5.4 Assessment for generated rationales

We assessed the qualities of CoT examples using GPT-3.5-turbo. Inspired by the ranking model, we prompted GPT-3.5-turbo to rank the rationales generated by both DSS and our QCRD, rather than providing scores based on the qualities of the rationales. This is easier for the LLM and allows for a more straightforward comparison. The prompt fed to GPT-3.5-turbo was presented in Appendix Table 4. To evaluate the rationales, we randomly selected 50 examples from each of the four datasets and asked GPT-3.5-turbo to determine rationales of which our method was better. We then aggregated the counts of "DSS is better," "Both are good," and "QCRD is better," as shown in Table 7. From the results, we observed that on SVAMP, CQA, and ESNLI, the model trained using our method generated better rationales than using DSS. However, on ANLI1, the model trained using DSS exhibited slightly better performance.

Table 7: The quality assessment results on the T5-base model for different sampling temperature settings, where three numbers represent counts of "DSS is better", "Both are good", and "QCRD is better", respectively.

	SVAMP	CQA	ANLI1	ESNLI
$\tau = 0$	21/0/29	19/6/25	26/1/23	17/11/22
$\tau = 0.7$	22/0/28	20/6/24	25/1/24	14/3/33

5.5 Distribution of rationale quality scores

The probability density estimation for the sampled rationale scores from the trained discriminator on the SVAMP test dataset is shown in Fig 6. Specifically, we considered the quality scores of: (1) positive and negative rationales from LLM's sampled outputs (sampled 5 times); (2) negative rationales from sampling a trained T5-base model with temperature τ set to 1.5 and 2.0, respectively. It showed that the trained discriminator can effectively score different types of rationales. Scores of LLM's positive rationales were around 0.95. For the trained student model, scores of the sampled negative rationales sometimes exceeded 0.7 (see the orange distribution), and it was necessary for the discriminator to assign low weights to these rationales. Furthermore, by comparing the orange distribution and the red distribution, we can see that the sampling temperature has a significant influence on the qualities of the rationales.

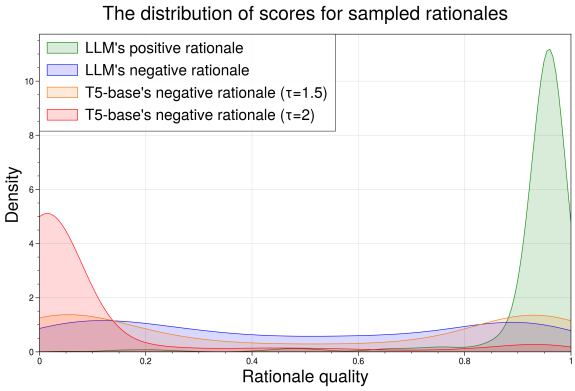

Figure 6: The probability density estimation for sampled rationale scores on the SVAMP test dataset, where rationales were from LLM's and trained T5-base model's sampled outputs, and τ denotes sampling temperature.

6 Conclusion

The knowledge distillation of CoT rationales from LLMs into smaller language models using a multi-task learning framework has been empirically shown to enhance performances of smaller language models. Building upon the framework, we introduces a general CoT distillation method, incorporating a contrastive learning perspective that considers both positive and negative knowledge. To generate positive and negative rationales, we propose an innovative approach that combines temperature sampling, the self-consistency of LLMs, and the self-adversarial of small language models them-

selves. Additionally, we develop a many-to-one contrastive distillation loss for better knowledge learning, where an online-update discriminator is used to judge qualities of rationales and assign them weights for optimizing the training process across the whole datasets. Extensive experiments conducted on multiple reasoning tasks demonstrate the superiority of our method over previous ones.

Limitations

Our work serves as a distillation method for deployed smaller language models, paving the way for further improvements. On one hand, as illustrated in Appendix A.6, it requires additional training time due to the distillation of sampled positive and online-inferenced negative rationales. However, our proposed method enhances model performance without incurring additional deployment costs and can be applied generally. On the other hand, the quality of knowledge for distillation is crucial. In this paper, we prompt the LLM to generate chain-of-thought (CoT) rationales and further classify them into positive and negative categories through self-consistency. Different types of prompts and decoding strategies can lead the LLM to produce various forms of positive CoT knowledge and more intuitive negative CoT knowledge, which may further improve the distillation effect.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant 62202442.

References

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. *Advances in Neural Information Processing Systems*, 31.

Justin Chih-Yao Chen, Swarnadeep Saha, Elias Stengel-Eskin, and Mohit Bansal. 2024a. Magdi: Structured distillation of multi-agent interaction graphs improves reasoning in smaller language models. *arXiv preprint arXiv:2402.01620*.

Xin Chen, Hanxian Huang, Yanjun Gao, Yi Wang, Jishen Zhao, and Ke Ding. 2024b. Learning to maximize mutual information for chain-of-thought distillation. *arXiv preprint arXiv:2403.03348*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023. Specializing smaller language models towards multi-step reasoning. In *International Conference on Machine Learning*, pages 10421–10430. PMLR.

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

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. 2020. A survey on contrastive self-supervised learning. *Technologies*, 9(1):2.

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.

Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. 2020. Contrastive representation learning: A framework and review. *Ieee Access*, 8:193907–193934.

Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, et al. 2022. Explanations from large language models make small reasoners better. *arXiv preprint arXiv:2210.06726*.

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

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

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*.

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

Mary Phuong and Christoph Lampert. 2019. Towards understanding knowledge distillation. In *International conference on machine learning*, pages 5142–5151. PMLR.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Matthew Renze and Erhan Guven. 2024. The effect of sampling temperature on problem solving in large language models. *arXiv preprint arXiv:2402.05201*.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. 2018. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144.

Keith E Stanovich and Richard F West. 1991. 24. individual differences in reasoning: Implications for the rationality debate?

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Peifeng Wang, Zhengyang Wang, Zheng Li, Yifan Gao, Bing Yin, and Xiang Ren. 2023. Scott: Self-consistent chain-of-thought distillation. *arXiv preprint arXiv:2305.01879*.

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

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

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

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

Daniel Zeman, Jan Hajic, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. Conll 2018 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual parsing from raw text to universal dependencies*, pages 1–21.

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

A Appendix

A.1 Details about Datasets

Following the setting in (Hsieh et al., 2023), we provide detailed descriptions of the four benchmark datasets in Table 1. To illustrate the unbalanced proportion of positive and negative rationales from LLMs given the ground truth, we displayed the statistical description of the generated rationale annotations on training datasets for four benchmarks in Table 2. On the one hand, the number of positive rationales was larger than that of negative rationales (3.87:1.13). On the other hand, for many samples in the training dataset (more than 50%), there were only positive rationales. Therefore, there is a need to generate effective negative rationales in other ways.

Table 1: Descriptions of the four benchmark datasets.

Dataset	Training	Validation	Test
SVAMP	720	80	200
CQA	8766	975	1221
ANLI1	16946	1000	1000
ESNLI	549367	9842	9824

Table 2: Statistical descriptions of the generated rationale annotations, where r denotes rationale, and positive r achieves correct answers.

Dataset	SVAMP	CQA	ANLI1	ESNLI
Average pos r (total 5)	3.87	3.89	3.93	3.31
Proportion with only pos r	0.55	0.68	0.66	0.50
Proportion with only neg r	0.08	0.13	0.11	0.20

A.2 Iteration-before-models for negative rationale generators

In this paper, we dynamically generated negative rationales using iteration-before-models through online temperature sampling. We took these iteration-before-models as negative generators, and we sampled them with a relatively high temperature value

to generate negative rationales for every batch of datasets. As depicted in Fig. 1, to select a j-iteration-before-model for the negative rationale generator, we need to save a minimum of j checkpoints for the model. This allows us to load the negative generator online and train the student model end-to-end instead of using a multi-turn approach. Additionally, as shown in Table 3, we found that the performance of the student models was not sensitive to the choice of j from $\{3, 5, 10\}$, all of which outperformed the results obtained with a fixed negative generator (pretrained by DSS (Hsieh et al., 2023)) or by using negative rationales derived from the self-consistency of the LLM.

Table 3: Results on SVAMP with different negative knowledge strategies, where the "Fixed" denotes fixing negative generator with the pretrained model, "SC" denotes using negative rationales from self-consistency.

Negative source	j=3	j=5	j=10	Fixed	SC
T5-base	68.0	69.0	68.5	66.5	64.5

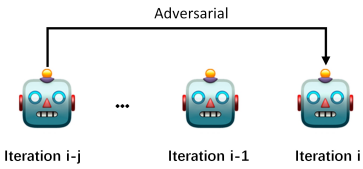

Figure 1: A case of the j-iteration-before-model for the negative rationale generator.

Table 4: The prompt for GPT-3.5-turbo to judge rationales.

The prompt for GPT-3.5-turbo
There is an input pair of a question and an answer of a taskname task, and we provide you two explanations. You need judge which explanation is better. The better explanation should be more accurate and explain the answer better.

A.3 Details about the Quality Judge

We incorporated a discriminator into our training process to assess the quality of rationales and assign corresponding weights to the losses. To construct the discriminator, we leveraged the encoder of the T5-base model along with one maxpooling layer and two linear layers to compute the quality score. Prior to training, the discriminator needs to be pretrained using the output rationales generated by LLMs with applying data augmentations to the negative rationales. Specifically, we employed word mask and replacement with the assistance of StanfordNLP (Zeman et al., 2018) to balance the proportions of positive and negative rationales. The training objective is L_D in (2). We pretrained the discriminator 500 max steps and we ensured scores for positive rationales close to 1 and scores for negative rationales close to 0. The discriminator was further online-updated during training.

A.4 Generalization Capability of QCRD

Our proposed QCRD is a general method that can be applied to other methods or models. We further conducted experiments to validate it. Specifically, we applied our QCRD to the baseline MI (Chen et al., 2024b) and the larger T5 model (T5-large with 770M parameters) on the SVAMP and ANLI1 benchmarks, which clearly demonstrate performance gains. As shown in Table 5, our QCRD effectively improves performance; for example, the accuracy increased by 2.5% for MI and by 2.0% for T5-large on SVAMP. Additionally, we fine-tuned the Qwen2.5-0.5B model (Team, 2024) on the SVAMP benchmark, and the results urther confirm the broader applicability for QCRD.

Table 5: Results of our QCRD applied to baselines and base models.

Benchmark	Model	Baseline	w/o QCRD	
			×	✓
SVAMP	T5-base	MI	67.5	70.0(+2.5)
ANLI1	T5-base	MI	54.2	56.0(+1.8)
SVAMP	T5-large	DSS	78.0	80.0(+2.0)
ANLI1	T5-large	DSS	53.2	55.1(+1.9)
SVAMP	Qwen2.5-0.5B	DSS	72.0	76.0(+4.0)

A.5 Influence of negative sampling temperature

The results of the T5-base model distilled by our method on SVAMP were displayed in Table 6 in terms of different negative sampling temperature settings. It was observed that when no sampling was performed (i.e., $\tau = 0$) or a lower temperature value was used (i.e., $\tau = 0.7$), the smaller model exhibited relatively poorer performance and showed larger fluctuations in accuracy. The best results were achieved when the temperature τ was set to 1.5. The reason for this can be attributed to the fact that when the model approaches convergence, the output rationales with lower temperature values tend to be similar to the golden ones. Considering these similar outputs as negative samples can lead

to detrimental effects.

Table 6: Results of our method on SVAMP with different sampling temperature τ.

Temperature τ	0	0.7	1.5	2
T5-base	65.0	64.5	69.0	67.5

Table 7: The comparison for training time on T5-base and T5-small models, where D denotes the Quality Judge.

Method	Base/small training time (h)
Finetune	2.0 / 1.25
DSS	4.0 / 2.5
MI	4.2 / 2.6
QCRD	13.0 / 8.5
QCRD (w/o D)	12.0 / 7.5

A.6 Computational cost

The training times for the T5-base and T5-small models using each method are presented in Table 7. Specifically, we trained the models on a single A100-80G GPU utilizing the SVAMP benchmark. Compared to DSS (Hsieh et al., 2023), our method requires an additional 9 hours for T5-base and 6 hours for T5-small. This increase is attributed to each input necessitating 5 positive rationales and 1 online-inferred negative rationale for contrastive rationale distillation. However, we emphasize the motivation behind our method: to enhance the performance of deployed small language models to the fullest extent, even surpassing general LLMs in specialized fields. Our proposed QCRD effectively improves model performance without incurring additional parameter storage during deployment and can be widely applied to other methods or models.