Enhancing Generalization in Chain of Thought Reasoning for Smaller Models

Maxwell J. Yin^{*1}, Dingyi Jiang^{*1}, Yongbing Chen^{2}, Boyu Wang$^{\dagger 1}$, Charles Ling$^{\dagger 1}$
[1]Western University, London, ON, Canada
[2]Wenzhou Academy of Agricultural Sciences, Wenzhou, China
jyin97@uwo.ca, djiang94@uwo.ca, chenyongbing@wzvcst.edu.cn, bwang@csd.uwo.ca,
charles.ling@uwo.ca

Abstract

Chain-of-Thought (CoT) reasoning in smaller language models is a challenging natural language process problem yet highly desirable in many real-life applications. Existing CoT knowledge distillation methods often suffer from overly conservative memorization in smaller LLMs, leading to low generalization confidence. As fully preserving the CoT ability of teacher model is impossible, we hypothesize that adversarial CoT fine-tuning is crucial for developing smaller LLM with robust CoT generalization. To this end, we propose *PRompt-Assisted Domain-Adversarial fine-tuning* (PRADA), a principled fine-tuning framework that integrates diverse CoT domains. Specifically, PRADA pioneers two CoT improvements in smaller LLM: (1) Recovering the domain-invariant feature insight which typically lost during distillation with domain adversarial fine-tuning; (2) Enhancing the domain adaptability of CoT prompt engineering by employing domain-adversarial approaches. We theoretically demonstrate the effectiveness of our approach and empirically show that it significantly outperforms the state of the arts in a wide range of tasks. Moreover, our empirical findings reveal that the smaller LLM, when leveraging PRADA, aligns closely with domain knowledge, thereby improving the explainability of our approach.

1 Introduction

In recent years, Large Language Models (LLMs) have witnessed significant advancements. To solve complex tasks, prompt-based CoT reasoning [Wei *et al.*, 2023] has emerged as a powerful tool to elicit advanced reasoning ability for LLM. Based on this method, LLMs smoothly achieved more cognitive and logical reasoning paths and higher answer correctness by requiring LLMs to perform CoT reasoning. For example, Kojima *et al.* [2023] found that LLMs can be prompted to generate a series of intermediate CoT reasoning steps by simply asking the model to think step by step.

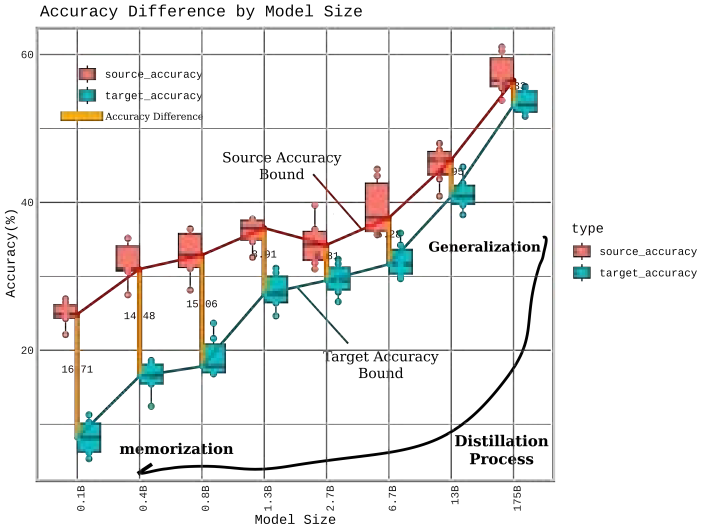
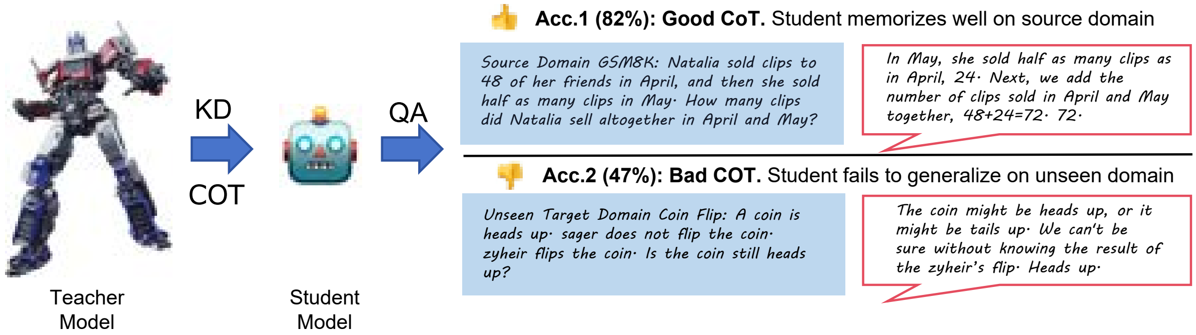


Figure 1: Transition from memorization to generalization, for GPT-3 families fine-tuned on four different domains (datasets detailed in 4. The source accuracy is tested on the original domain, and the target accuracy is tested on different domains.

Whereas the emergence of CoT reasoning is based on extremely large models that usually require more than 100B parameters [Wei *et al.*, 2023], the knowledge distillation of CoT reasoning models [Li *et al.*, 2022] is being used to reduce the number of parameters in LLM. Consequently, the CoT knowledge distillation enables smaller models to imitate the larger model's logical reasoning steps effectively, rather than generating simple and unreliable question-answer pairs [Shridhar *et al.*, 2023; Ho *et al.*, 2023; Fu *et al.*, 2023].

However, a major drawback of CoT knowledge distillation is the shift of model ability from generalization to memorization (as shown in Figure 1), as the number of parameters of the student model is drastically reduced compared to the teacher model [Rabin *et al.*, 2023]. In the knowledge distillation setting, the smaller student model often memorizes the pretrained CoT paths [Fu *et al.*, 2023; Antoniades *et al.*, 2024] instead of maintaining the larger teacher model's CoT generalization ability. As shown in Figure 2, the parameter-reduced student model performs CoT reasoning well on the well-informed source domain but fails to generalize to the

[*] Equal contribution.
[†] Co-corresponding Author.


KD
COT
QA
Teacher Model
Student Model
👍 **Acc.1 (82%): Good CoT.** Student memorizes well on source domain
👎 **Acc.2 (47%): Bad COT.** Student fails to generalize on unseen domain

Source Domain GSM8K: Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?

In May, she sold half as many clips as in April, 24. Next, we add the number of clips sold in April and May together, 48+24=72. 72.

Unseen Target Domain Coin Flip: A coin is heads up. sager does not flip the coin. zyheir flips the coin. Is the coin still heads up?

The coin might be heads up, or it might be tails up. We can't be sure without knowing the result of the zyheir's flip. Heads up.

Figure 2: Case study of model CoT reasoning ability decay in the knowledge distillation process from GPT-3 175B to GPT-3 6.7B.The reduction in model parameters causes the student model's CoT generalization ability to diminish, shifting towards memorization. Consequently, the student model exhibits bad CoT generalization on unseen data.

new, unseen domain.

To address this, we propose a CoT distillation framework called *PRompt-Assisted Domain-Adversarial fine-tuning* (PRADA): (1) A large teacher model is prompted to generate various CoT reasoning responses. (2) prompt learning layers are incorporated into the smaller model to facilitate the acquisition of domain-agnostic knowledge. (3) Domain-adversarial fine-tuning is applied to both source CoT domain and target domain, enabling the model to learn invariant domain features and mitigate the effects of domain shifts.

We conduct extensive empirical evaluations using 12 datasets in various domains. Our results demonstrate that PRADA substantially outperforms former CoT knowledge distillation methods in cross-domain scenarios. We find that PRADA effectively enhances the cross-domain generalization versatility of smaller models while retaining the CoT reasoning capabilities of the teacher model, which previously relied on *hundreds of billions of* parameters [Kojima *et al.*, 2023; Wei *et al.*, 2023].

Our contributions can be summarized as:

- Enhancing the domain discrimination capability of LLMs through an adversarial fine-tuning framework.
- Improving the domain adaptability of CoT prompt tuning via adversarial learning.
- Addressing generalization degradation in the knowledge distillation process for Chain-of-Thought reasoning

2 Related Work

Knowledge Distillation Based on Chain-of-Thought Reasoning In the evolving realm of knowledge distillation (KD) and chain-of-thought (CoT) reasoning, the MT-COT framework [Li *et al.*, 2022] stands out by utilizing COT explanations generated by LLMs to bolster the training of compact reasoning engines. Building on this foundation, the Specialized model[Fu *et al.*, 2023] extracts CoT reasoning paths from large teacher models to enhance out-of-distribution generalization. This specialized extraction is crucial for improving the adaptability and accuracy of models when encountering unfamiliar data distributions. Further advancing the field, Fine-tune CoT[Ho *et al.*, 2023] introduces a technique where multiple COT reasoning solutions are generated from LLMs through random sampling. This method addresses the challenge of limited training data diversity, which often hampers the generalization performance of student models. Innovatively, the SOCRATIC CoT[Shridhar *et al.*, 2023] splits the training into two distilled models: a problem decomposer and a subproblem solver. This approach mitigates the cognitive load on student models, allowing for more effective learning and CoT reasoning.

Domain Adversarial Neural Network Our research is closely aligned with Adversarial Training (AT) [Goodfellow *et al.*, 2015], a prevalent machine-learning technique for enhancing model robustness. Building on this foundation, the Domain-Adversarial Neural Network (DANN) [Ganin *et al.*, 2016] is designed to improve the model's generalization ability. In the field of Natural Language Processing (NLP), the standard approach involves applying adversarial perturbations to word embeddings [Miyato *et al.*, 2021]. This technique has been demonstrated to significantly enhance model generalization ability when used for fine-tuning across various downstream tasks, as evidenced by methods such as FreeLB[Zhu *et al.*, 2020], SMART[Jiang *et al.*, 2020], InfoBERT[Wang *et al.*, 2021], and CreAT[Wu *et al.*, 2023]. Additionally, ALUM[Liu *et al.*, 2020] provides empirical evidence that adversarial training can yield substantial pretraining benefits. Our work advances fine-tuning processes for pre-trained language models (PrLMs) [Devlin *et al.*, 2019; Liu *et al.*, 2020; Raffel *et al.*, 2023; He *et al.*, 2021]. It is independent of the prevailing pre-training paradigms, such as Masked Language Modeling (MLM) [Devlin *et al.*, 2019], Replacement Token Detection (RTD)[Clark *et al.*, 2020], Permuted Language Modeling (PLM)[Cui *et al.*, 2022], and multiple-objective approaches[Wu *et al.*, 2022].

Prompt Learning The field of prompt learning has witnessed significant advancements, providing innovative approaches to enhance Natural Language Understanding (NLU) tasks. "Prefix Tuning," introduced by [Li and Liang, 2021], improves model comprehension by appending prefixes to input sequences. The "WARP" method by [Hambardzumyan *et al.*, 2021] and "P-tuning" by [Liu *et al.*, 2023] modify language model outputs and input sequences, respec-

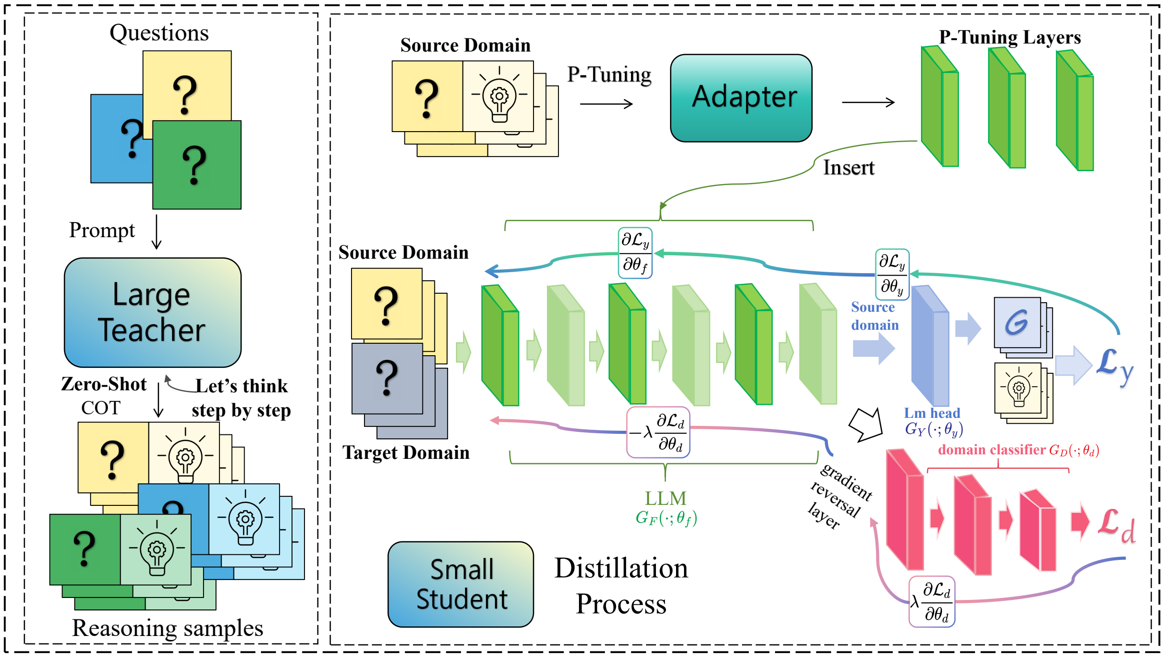

Questions
Prompt
Large Teacher
Zero-Shot COT
Let's think step by step
Reasoning samples
Source Domain
P-Tuning
Adapter
P-Tuning Layers
Insert
Source Domain
Target Domain
$\frac{\partial \mathcal{L}_y}{\partial \theta_f}$
$\frac{\partial \mathcal{L}_y}{\partial \theta_y}$
Source domain
Lm head $G_Y(\cdot;\theta_y)$
$\mathcal{L}_y$
$-\lambda \frac{\partial \mathcal{L}_d}{\partial \theta_d}$
LLM $G_F(\cdot;\theta_f)$
gradient reversal layer
domain classifier $G_D(\cdot;\theta_d)$
$\mathcal{L}_d$
$\lambda \frac{\partial \mathcal{L}_d}{\partial \theta_d}$
Small Student
Distillation Process

Figure 3: Illustration of our method Prompt-Assisted Domain-Adversarial fine-tuning (PRADA). **Firstly**, the teacher model is prompted to generate diverse CoT reasoning responses using a Zero-Shot-CoT approach. **Secondly**, an adapter is inserted into the student LLM for P-Tuning, which refines the domain-agnostic knowledge of the source domain. **Thirdly**, the knowledge distillation process incorporates data from both the labeled source domain and the unlabeled target domain. The proposed architecture includes an LLM (green) and a language modeling head (blue), which together form a standard feed-forward architecture. Unsupervised domain adversarial fine-tuning is achieved by adding a domain classifier (red).

tively, achieving robustness comparable to traditional fine-tuning while employing fewer task-specific parameters. Domain Adaptation via Prompt Learning (DAPL)[Ge *et al.*, 2022] leverages pretrained vision-language models by optimizing a minimal number of parameters to embed domain-specific information into prompts. This method enables the classifier to dynamically adapt to each domain. Prompt-Assisted Self-Adaptive Learning (PASAL)[Yin *et al.*, 2024] utilizes prompts to facilitate the acquisition of domain-agnostic knowledge, effectively addressing domain shifts.

3 Methodology

3.1 Problem Definition

Given a set of source questions $\mathcal{D}_s = \{(q_i^s)\}_{i=1}^{N_s}$ and a set of target questions $\mathcal{D}_t = \{(q_i^t)\}_{i=1}^{N_t}$, we adopt a model trained from a source domain to a target domain simultaneously. Here, N_s and N_t denote the sizes of the source domain dataset $\mathcal{D}_s$ and the target domain dataset $\mathcal{D}_t$, respectively. Let $G_F(\cdot;\theta_f)$ be a pretrained language model with a hidden state size of H and a vocabulary size of $|\mathcal{V}|$. The input embeddings are given through the pretrained embedding layer, where $e \in \mathbb{R}^{|\mathcal{V}|\times H}$. In the following, we characterize the general PRADA architecture in three steps, which are also visualized in Figure 3.

3.2 Teacher: Diverse Reasoning Generation

First, we leverage a large teacher model to generate a variety of CoT reasoning responses for a given task. To enhance usage flexibility and minimize teacher inference costs, we employ the task-agnostic Zero-Shot-CoT prompting method [Kojima *et al.*, 2023] on teacher models. Consider a standard sample S_i comprising a question q_i and its correct answer a_i. By using Zero-Shot-CoT, we prompt the teacher model to generate a reasoning rationale, $\hat{r}_i$, to address question q_i and provide a final answer prediction $\hat{a}_i$. The result text sequence includes the question and generated content:“ Q: $\langle q_i \rangle$. A: Let's think step by step. $\langle \hat{r}_i \rangle$ Therefore, the answer is $\langle \hat{a}_i \rangle$.” To enhance reasoning ability, we leverage the effort of diverse CoT reasoning. For a given sample S_i, we employ a stochastic sampling strategy, specifically temperature sampling with a large t, to generate D distinct outputs $\{(\hat{r}_{ij}, \hat{a}_{ij})\}_{j=1}^{D}$. We refer to D as the degree of reasoning diversity.

Next, we filter the generated samples and reformat them into prompt-completion pairs. For filtering, we compare the teacher model's final prediction $\hat{a}_i$ with the ground truth answer a_i [Huang *et al.*, 2022; Zelikman *et al.*, 2022]. For all instances where $\hat{a}_i = a_i$, we repackage $(S_i, \hat{r}_i, \hat{a}_i)$ into a reasoning sample $S_i' = (q_i, c_i)$, a question-completion pair. Specifically, q_i and c_i take the forms of “$\langle q_i \rangle$ ###” and “$\langle \hat{r}_i \rangle - -\rangle \langle a_i \rangle$ END”.

3.3 Student: Prompt Learning

We start with a source domain $\mathcal{D}_s$ comprising CoT reasoning samples $\{(q_i, c_i)\}_{i=1}^{N_s}$, where q_i represents the question sequence and c_i denotes the corresponding completion. Next, we deploy the P-Tuning adapter on the student model, which employs continuous prompt embeddings to enhance stability and improve prompting performance. Let $[P]$ represent the continuous prompt embedding. The prompt template for P-Tuning is defined as follows:

$$T_i = \{P_{[0:j]}, q_i, P_{[(j+1):k]}, c_i, P_{[(k+1):l]}\} \tag{1}$$

P-Tuning leverages an additional embedding function $f : [P_i] \rightarrow h_i$ to map the template to:

$$\{h_0, \ldots, h_j, e(q_i), h_{j+1}, \ldots, h_k, e(c_i), h_{k+1}, \ldots, h_l\}$$

During the prompt learning, the embeddings $\{P_i\}_{i=1}^{k}$ are updated to optimize a task loss function $\mathcal{L}_p$ using the template. These multiple P-tuning layers are trained as an adapter on the source domain, which employs deep prompt learning that involves a lot of P-tuning layers to improve and stabilize prompting [Liu *et al.*, 2022]. Then, we insert the adapter into the student LLM. When the student model is fine-tuning or inferencing, the prompts in different P-tuning layers are added as prefix tokens, providing the student LLM with more tunable domain-agnostic parameters of the source domain data [Yin *et al.*, 2024].

3.4 Student: Domain Adversarial Fine-Tuning

Let $G_F(\cdot;\theta_f)$ represent the student LLM, with parameters θ_f. Then, let $G_Y(\cdot;\theta_y)$ denote the natural language modeling head, with parameters θ_y, while $G_D(\cdot;\theta_d)$ corresponds to the domain classifier of the PRADA architecture, with parameters θ_d. The input source domain data $\{(q_i, c_i)\}_{i=0}^{N_s}$ and target domain data $\{(q_i)\}_{i=0}^{N_t}$ are transformed into $\{(q_i, c_i, d_s)\}_{i=0}^{N_s}$ and $\{(q_i, d_t)\}_{i=0}^{N_t}$, respectively, where d_s and d_t are unique symbols distinguishing the domains. We denote the prediction loss and the domain loss, respectively, :

$$L_y^i(\theta_f, \theta_y) = L_y(G_y(G_f(q_i;\theta_f);\theta_y), c_i) \tag{2}$$

$$L_d^i(\theta_f, \theta_d) = L_d(G_d(G_f(q_i;\theta_f);\theta_d), d_i) \tag{3}$$

Then, training PRADA parallels the single-layer case and consists of optimizing:

$$\begin{aligned} E(\theta_f, \theta_y, \theta_d) &= \frac{1}{N_s}\sum_{i=1}^{n} L_y^i(\theta_f, \theta_y) \\ &- \lambda\Big(\frac{1}{N_s}\sum_{i=1}^{N_s} L_d^i(\theta_f, \theta_d) + \frac{1}{N_t}\sum_{i=1}^{N_t} L_d^i(\theta_f, \theta_d)\Big), \end{aligned} \tag{4}$$

A saddle point can be found as a stationary point of the following gradient updates[Ganin *et al.*, 2016]:

$$\theta_f \leftarrow \theta_f - \mu\left(\frac{\partial L_y^i}{\partial \theta_f} - \lambda\frac{\partial L_d^i}{\partial \theta_f}\right), \tag{5}$$

$$\theta_y \leftarrow \theta_y - \mu\frac{\partial L_y^i}{\partial \theta_y}, \tag{6}$$

Algorithm 1: PRADA Algorithm

```
Input: Source questions $\mathcal{D}_s$, Target questions $\mathcal{D}_t$,
       Teacher LLM $\boldsymbol{F}$, Domain Classifier $\boldsymbol{D}$
Output: Learned student LLM $\boldsymbol{f}$
Step 1: Teacher - Diverse CoT Generation
for each $q_i \in \mathcal{D}_s$ do
    Obtain the CoT reasoning $c_i \leftarrow \boldsymbol{F}(q_i)$;
end
Obtain the Query-CoT Pairs
  $\mathcal{C}_s \leftarrow \{(q_i, c_i) \mid q_i \in \mathcal{D}_s\}$;
Step 2: Student - Domain Adversarial Fine-Tuning
Initialize Adapter $\boldsymbol{f}$ using $\boldsymbol{P}$-Tuning on $\mathcal{C}_s$;
while not converged do
    Sample $(q_i^s, c_i^s) \in \mathcal{C}_s$, $q_i^t \in \mathcal{D}_t$;
    Compute task-specific loss $L_y^i$ using $(q_i^s, c_i^s)$ via
      Eq. (2);
    Compute domain adversarial loss $L_d^i$ using $q_i^s$ and
      $q_i^t$ via Eq. (3);
    Update $\boldsymbol{f}$ and $\boldsymbol{D}$ via Eq. (10);
end
```

$$\theta_d \leftarrow \theta_d - \mu\lambda\frac{\partial L_d^i}{\partial \theta_d}, \tag{7}$$

where μ is the learning rate and λ is the domain classifier loss weight. We use stochastic estimates of these gradients by sampling examples from the dataset. The only difference is that in Equation (3), the gradients from the LLM and domain predictors are subtracted, instead of being summed. This distinction is crucial, as otherwise, SGD would attempt to make features dissimilar across domains in order to minimize the domain classification loss. Fortunately, this can be achieved by introducing a special gradient reversal layer (GRL)[Ganin *et al.*, 2016], defined as follows:

$$R(q) = q, \tag{8}$$

$$\frac{dR}{dq} = -I, \tag{9}$$

where I is an identity matrix. We can then define the objective "pseudo-function" of $(\theta_f, \theta_y, \theta_d)$ that is being optimized by the stochastic gradient descent within our method:

$$\begin{aligned} \tilde{E}(\theta_f, \theta_y, \theta_d) &= \frac{1}{N_s}\sum_{i=1}^{N_s} L_y(G_y(G_f(q_i;\theta_f);\theta_y), c_i) \\ &- \lambda\Big(\frac{1}{N_s}\sum_{i=1}^{N_s} L_d(G_d(R(G_f(q_i;\theta_f));\theta_d), d_i) \\ &+ \frac{1}{N_t}\sum_{i=1}^{N_t} L_d(G_d(R(G_f(q_i;\theta_f));\theta_d), d_i)\Big). \end{aligned} \tag{10}$$

Models	#Source Domain	Comparison Experiments for PRADA on Arithmetic Reasoning Problems					
		SVAMP	MultiArith	AQUA	Last Letter	Strategy QA	Tracking Shuffled
		Acc.	Acc.	Acc.	Acc.	Acc.	Acc.
Specialized, T5	GSM8K	24.21	27.5	14.56	19.49	58.7	27.6
MT-CoT, T5		22.8	15.45	19.82	16.94	79.35	32.87
PRADA, T5		26.93	28.15	19.85	67.85	81.56	38.36
Fine-Tune-CoT, Llama3		52.85	52.05	52.13	**41.67**	88.53	79.8
PRADA, Llama3		**53.07**	**53.11**	**57.47**	38.2	**91.42**	**89.25**
Specialized, T5	Coin Flip	32.12	25.50	14.62	61.4	63.72	28.62
MT-CoT, T5		21.81	24.47	18.85	66.91	78.13	21.38
PRADA, T5		35.68	32.84	16.48	81.24	84.68	30.97
Fine-Tune-CoT, Llama3		**42.88**	32.01	31.13	90.67	87.53	88.8
PRADA, Llama3		42.63	**34.09**	**38.00**	**92.10**	**90.44**	**96.99**
Specialized, T5	CommonSense QA	16.23	22.57	21.65	30.48	33.71	29.62
MT-CoT, T5		12.12	23.74	27.81	25.99	27.32	30.85
PRADA, T5		**33.86**	35.59	27.56	59.48	38.64	53.78
Fine-Tune-CoT, Llama3		31.81	34.01	30.13	90.67	47.53	78.18
PRADA, Llama3		27.05	**42.39**	**39.14**	**94.92**	**50.86**	**79.52**
Specialized, T5	Date Understanding	23.23	32.25	21.06	61.42	70.17	57.36
MT-CoT, T5		20.81	33.41	17.83	55.97	77.33	60.88
PRADA, T5		25.19	34.17	20.03	61.18	**94.72**	72.49
Fine-Tune-CoT, Llama3		31.89	34.03	40.13	90.67	77.53	78.84
PRADA, Llama3		**33.96**	**37.2**	**41.08**	**92.1**	76.11	**84.58**

Table 1: Comparison of PRADA(ours) with the related methods for reasoning tasks. Each row represents the accuracy (%) of a compared method adapting between the source and target domain.

Notably, the overall student parameter update includes the update of P-tuning layers. To maintain the effectiveness of P-tuning layers, specifically the domain-agnostic learning ability of P-tuning layers learned on the source domain, we use a learning rate difference strategy between the training process of the adapter and the LLM. In the prompt learning process, we utilize the learning rate in the order of $e^{-3} - e^{-4}$. However, the overall student model fine-tuning requires the learning rate in the order of $e^{-5} - e^{-6}$. This learning rate difference ensures that the previous prompt learning effect will not be significantly affected while domain-adversarial fine-tuning.

4 Experiments

Tasks and Datasets We assessed the effectiveness of our approach using 12 datasets that span four categories of complex reasoning, in line with the framework established by Kojima *et al.* [2023]. These categories are:

- Arithmetic Reasoning: SingleEq, AddSub, MultiArith, GSM8K, and SVAMP, which test various aspects of mathematical problem-solving and numerical reasoning.
- Symbolic Reasoning: Last Letter Concatenation and Coin Flip are used to evaluate reasoning that involves symbolic manipulation and pattern recognition.
- Commonsense Reasoning: CommonSenseQA and StrategyQA, which test the ability to apply everyday knowledge and reasoning strategies to answer questions.
- Other Reasoning:Date Understanding and Tracking Shuffled Objects, focusing on logical reasoning skills and the ability to track and understand sequences or temporal data.

For each dataset, we follow [Ho *et al.*, 2023], using the teacher model's various CoT reasoning for downstream student model fine-tuning.

Teacher/Student Models For the teacher models, we utilize the GPT-3 175B [Brown *et al.*, 2020], accessible via the OpenAI API. Specifically, the teacher model used for PRADA is text-davinci-002, which is based on Instruct-GPT 175B [Ouyang *et al.*, 2022]. The experiments involving student models are categorized into two types: Encoder-Decoder Architecture and Decoder-Only Architecture. For the decoder-only architecture, we choose the newly updated Llama3 model[Dubey *et al.*, 2024] for Lora fine-tuning [Hu *et al.*, 2021]. By the way, we originally wanted to choose GPT-3 and GPT-2 $\{$GPT 2-small-125M, GPT 2-median-335M, GPT 2-large-774M$\}$ as models, but because of the black-box nature of GPT 3 and the lack of mathematical ability of GPT 2, we discarded GPT families for the decoder-only model. For the encoder-decoder architecture, we incorporate Flan-T5-Large [Chung *et al.*, 2022], an instruction-tuned version of T5, to evaluate the impact of the PRADA on the encoder-decoder model. These student models are 25 to 250 times smaller than the teacher model, making them more practical for real-world deployment.

Baselines We provide a comparison of PRADA (ours) with three CoT knowledge distillation methods:

- **Specialized LLM** [Fu *et al.*, 2023]: This baseline extracts specialized reasoning paths, or Chains of Thought, from a large teacher model to enhance out-of-distribution generalization ability. By fine-tuning on these specialized CoT paths, the student model is expected to achieve superior performance on tasks requiring complex reasoning and extrapolation beyond the training distribution.
- **Fine-tune CoT** [Ho *et al.*, 2023]: This method involves

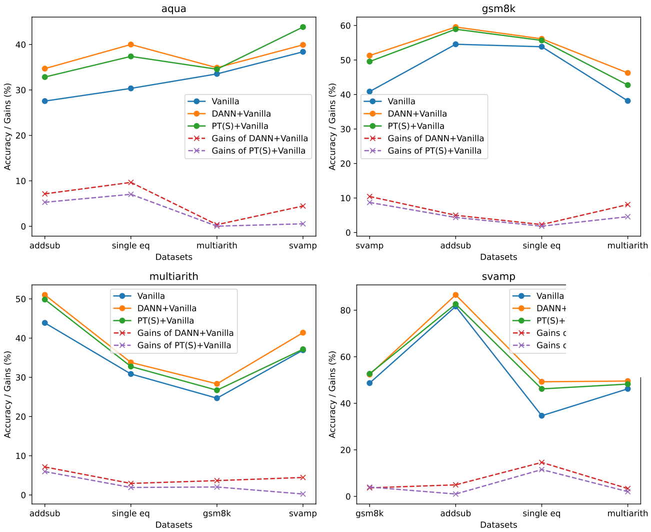


Figure 4: Ablation study (Acc. %) of Components of PRADA: Vanilla Llama3 versus Enhanced with Additional Components.

randomly sampling multiple reasoning solutions generated by LLMs. The augmentation of training data with diverse reasoning paths facilitates the learning process of the student model, enabling it to develop robust reasoning abilities through exposure to varied inferential strategies.

- **MT-CoT (Multi-task Chain of Thought)** [Li *et al.*, 2022]: This approach leverages explanations generated by large language models to enhance the training of smaller reasoning models. By integrating the interpretative ability of LLMs, MT-COT aims to improve the reasoning performance of compact models through enriched training data.

4.1 Main Results

We conducted extensive experiments on 12 datasets, comparing our Prompt-Assisted Domain-Adversarial fine-tuning (PRADA) models with several baseline and contemporary models. Specifically, we compared against the pretrained versions of Davinci-002, FlanT5-Large, and Llama3-8B, as well as the fine-tuned models: Fine-tune CoT, Specialized, and MT-COT. Our evaluation metrics were accuracy (Acc.) and the improvement in accuracy (Δ). For a fair comparison, we ensured that all methods followed the same experimental settings. Table 1 shows the performance of all models across multiple datasets, recording the average accuracy of target domain samples across various arithmetic reasoning tasks. Our PRADA-enhanced models consistently achieved the highest accuracy across all datasets. For instance, in the source SVAMP dataset, Llama3-8B + PRADA outperformed its pretrained counterpart by over 10% in accuracy.

4.2 Ablation Study

Without Domain Classifier. First, we investigated the impact of the domain classifier by eliminating it from the model.

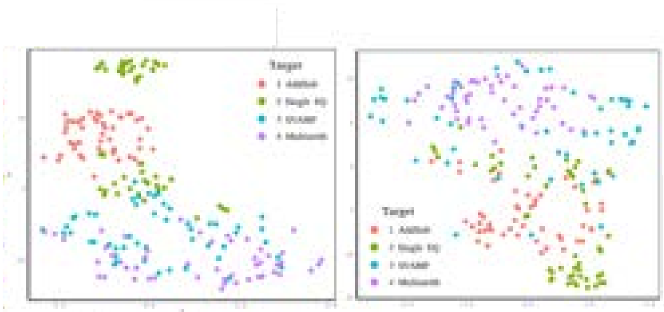


Figure 5: The t-SNE visualization of embeddings of Llama 3-8B using our method PRADA or not on source GSM8K for 4 arithmetic targets.

This experiment was designed to explore the domain classifier's role in facilitating effective model generalization ability. A decline in target domain performance following the removal of the domain classifier would suggest that it is vital for distinguishing domain-specific features and thereby maintaining robust performance across different domains. The results of these experiments are visualized in Figure 4. A specific case where the model was trained on GSM8K as the source domain and evaluated across various datasets is presented in Table 2.

Without P-Tuning Layers. Next, we assessed the role of P-Tuning layers by removing them from the model. The objective here was to determine how crucial these layers are to the system's overall performance. By comparing the results of this modified model against the full version that includes P-Tuning, we aimed to identify any significant drops in performance that would indicate the importance of these layers in enhancing the model generalization ability. The results of these experiments are visualized in Figure 4. A specific case is presented in Table 2.

Visualization of embedding with and without PRADA. The t-SNE visualizations (shown in Figure 5) compare the embeddings produced by LLM with and without the PRADA method across four arithmetic reasoning tasks. On the left of Figure 5, embeddings are shown for the vanilla Llama3 model fine-tuned on GSM8K as the source domain. On the right, embeddings are displayed for Llama3 utilizing PRADA, also trained on GSM8K as the source domain. The visualizations demonstrate the impact of prompt learning in aligning domain-capturing features, with the PRADA approach showing a more generalized clustering of embeddings across different tasks.

Difficulty of Transfer Learning For arithmetic reasoning datasets, we observe a ranking in terms of the difficulty of transfer with respect to dataset characteristics, which influences the transfer learning performance of PRADA. First, the AQUA dataset involves complex arithmetic tasks, such as solving systems of equations and combinatorics. Secondly, datasets like GSM8K, MultiArith, SVAMP, and AddSub involve more basic arithmetic operations, such as addition, subtraction, multiplication, and division. Based on these preliminary observations, we analyze the results in Table 1 to assess the difficulty within these arithmetic tasks. In particular, we observe that the sets GSM8K and AQUA appear to be too

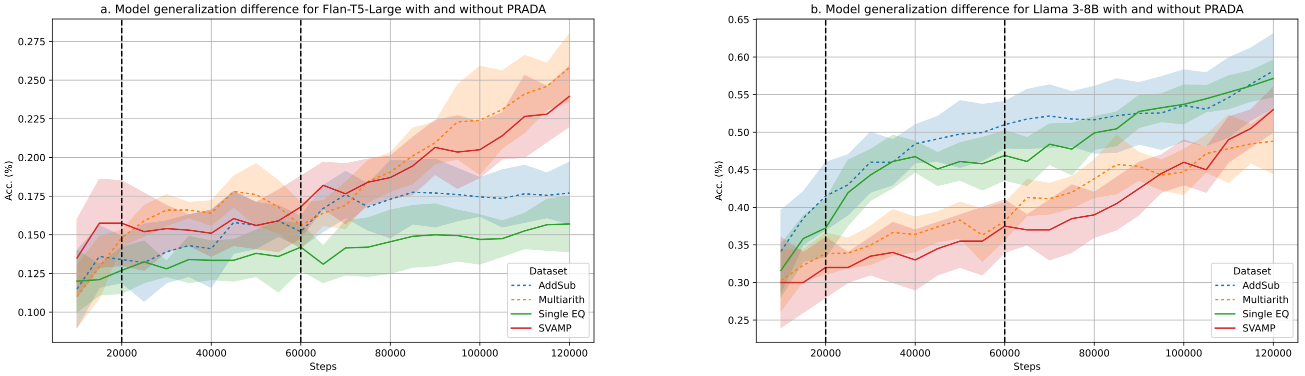


Figure 6: The generalization ability of models with and without PRADA (ours) finetuned on source domain GSM8K. The lower limit of the same color curve is the Vanilla model's accuracy, and the upper limit is the PRADA method's accuracy.

difficult for any small student model, given that the teacher model gets below 65% accuracy on both. The arithmetic task transfer difficulty decreasing from Left can be ranked as follows: AQUA, GSM8K, Multiarith, SVAMP, Single EQ, Addsub. By the way, this generalization difficulty based on domain-agnostic features is worth studying on other datasets.

Components			**Target Domains**			
$\mathcal{L}_y$	$\mathcal{L}_p$	$\mathcal{L}_d$	addsub	multiarith	single eq	svamp
✓			54.56	44.89	54.53	44.39
✓	✓		58.92	47.88	55.84	49.84
✓		✓	59.55	51.28	56.85	51.02
✓	✓	✓	**62.68**	**53.11**	**59.01**	**53.07**

Table 2: Ablation study (Acc. %) of PRADA based on llama3 without different components are fine-tuned and evaluated from a domain (GSM8K) to the other four arithmetic tasks.

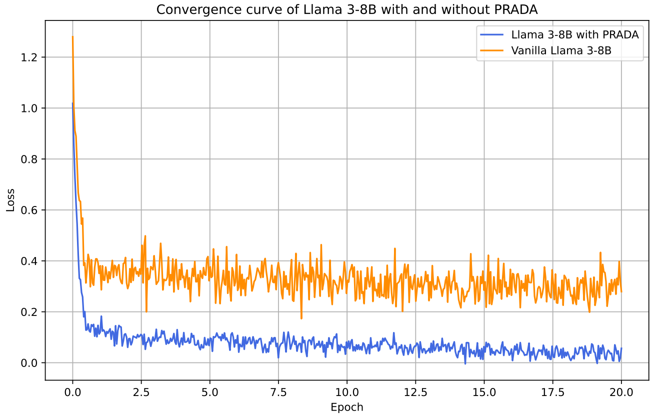


Figure 7: Convergence curve of Llama 3-8B with and without PRADA fine-tuned on GSM8K dataset.

Generalization ability Analysis In this part, we illustrate the enhancement in generalization ability for two models, Flan-T5-Large and Llama 3-8B, when utilizing the PRADA method compared to the vanilla model counterparts. Figure 6 displays the accuracy difference between models with and without PRADA as the training steps increase. For each dataset (AddSub, MultiArith, SingleEq, SVAMP), the upper limit of the shaded area represents the accuracy achieved by the model with PRADA, while the lower limit represents the accuracy without PRADA. The visualized result shows that PRADA can capture the essence of the generalization visualizations in the context of model generalization evaluation. The area between these curves reflects the improvement in model generalization due to the PRADA method's domain-agnostic feature capturing, highlighting its effectiveness across multiple domains.

Convergence Behavior Comparison Figure 7 presents the convergence curves of the Llama 3-8B model with and without the PRADA method during training on GSM8K dataset. The model trained with PRADA (blue curve) demonstrates faster convergence and lower overall loss compared to the vanilla model (orange curve), indicating more efficient learning of domain-agnostic features and better model performance with the PRADA technique.

5 Conclusion

In this work, we propose PRompt-Assisted Domain-Adversarial fine-tuning (PRADA), a novel adversarial fine-tuning framework to enhance the CoT ability of student models in knowledge distillation. Leveraging CoT knowledge distillation, prompt-assisted learning, and domain adversarial optimization, PRADA mitigates the degradation of generalization performance during Chain-of-Thought (CoT) reasoning distillation. This research explores factors causing CoT degradation, designs novel adversarial optimization techniques to improve the fidelity of distilled reasoning while preserving CoT power for portable LLMs. Extensive evaluations across twelve datasets highlight PRADA's superiority over prior CoT distillation methods, especially in scenarios requiring robust cross-domain generalization.

References

Antonis Antoniades, Xinyi Wang, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. Generalization v.s. Memorization: Tracing Language Models' Capabilities Back to Pretraining Data, July 2024. arXiv:2407.14985 [cs].

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners, July 2020. arXiv:2005.14165 [cs].

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling Instruction-Finetuned Language Models, December 2022. arXiv:2210.11416 [cs].

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators, March 2020. arXiv:2003.10555 [cs].

Yiming Cui, Ziqing Yang, and Ting Liu. PERT: Pre-training BERT with Permuted Language Model, March 2022. arXiv:2203.06906 [cs].

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, and Pandey. The Llama 3 Herd of Models, July 2024. arXiv:2407.21783 [cs].

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. Specializing Smaller Language Models towards Multi-Step Reasoning, January 2023. arXiv:2301.12726 [cs].

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-Adversarial Training of Neural Networks, May 2016. arXiv:1505.07818 [cs, stat].

Chunjiang Ge, Rui Huang, Mixue Xie, Zihang Lai, Shiji Song, Shuang Li, and Gao Huang. Domain Adaptation via Prompt Learning, February 2022. arXiv:2202.06687 [cs].

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples, March 2015. arXiv:1412.6572 [cs, stat].

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. WARP: Word-level Adversarial ReProgramming, June 2021. arXiv:2101.00121 [cs].

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced BERT with Disentangled Attention, October 2021. arXiv:2006.03654 [cs].

Namgyu Ho, Laura Schmid, and Se-Young Yun. Large Language Models Are Reasoning Teachers, June 2023. arXiv:2212.10071 [cs].

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models, October 2021. arXiv:2106.09685 [cs].

Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. Large Language Models Can Self-Improve, October 2022. arXiv:2210.11610 [cs].

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. SMART: Robust and Efficient Fine-Tuning for Pre-trained Natural Language Models through Principled Regularized Optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2177–2190, 2020. arXiv:1911.03437 [cs, math].

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large Language Models are Zero-Shot Reasoners, January 2023. arXiv:2205.11916 [cs].

Xiang Lisa Li and Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation, January 2021. arXiv:2101.00190 [cs].

Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, Wenhu Chen, and Xifeng Yan. Explanations from Large Language Models Make Small Reasoners Better, October 2022. arXiv:2210.06726 [cs].

Xiaodong Liu, Hao Cheng, Pengcheng He, Weizhu Chen, Yu Wang, Hoifung Poon, and Jianfeng Gao. Adversarial Training for Large Neural Language Models, April 2020. arXiv:2004.08994 [cs].

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-Tuning v2: Prompt Tuning Can Be Comparable to Fine-tuning Universally Across Scales and Tasks, March 2022. arXiv:2110.07602 [cs].

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. GPT Understands, Too, October 2023. arXiv:2103.10385 [cs].

Takeru Miyato, Andrew M. Dai, and Ian Goodfellow. Adversarial Training Methods for Semi-Supervised Text Classification, November 2021. arXiv:1605.07725 [cs, stat].

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, March 2022. arXiv:2203.02155 [cs].

Md Rafiqul Islam Rabin, Aftab Hussain, Mohammad Amin Alipour, and Vincent J. Hellendoorn. Memorization and Generalization in Neural Code Intelligence Models. *Information and Software Technology*, 153:107066, January 2023. arXiv:2106.08704 [cs].

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, September 2023. arXiv:1910.10683 [cs, stat].

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. Distilling Reasoning Capabilities into Smaller Language Models, May 2023. arXiv:2212.00193 [cs].

Boxin Wang, Shuohang Wang, Yu Cheng, Zhe Gan, Ruoxi Jia, Bo Li, and Jingjing Liu. InfoBERT: Improving Robustness of Language Models from An Information Theoretic Perspective, March 2021. arXiv:2010.02329 [cs].

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, January 2023. arXiv:2201.11903 [cs].

Hongqiu Wu, Ruixue Ding, Hai Zhao, Boli Chen, Pengjun Xie, Fei Huang, and Min Zhang. Forging Multiple Training Objectives for Pre-trained Language Models via Meta-Learning, October 2022. arXiv:2210.10293 [cs].

Hongqiu Wu, Yongxiang Liu, Hanwen Shi, Hai Zhao, and Min Zhang. Toward Adversarial Training on Contextualized Language Representation, May 2023. arXiv:2305.04557 [cs].

Maxwell Yin, Boyu Wang, and Charles Ling. Source-Free Unsupervised Domain Adaptation for Question Answering via Prompt-Assisted Self-learning. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 700–713, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping Reasoning With Reasoning, May 2022. arXiv:2203.14465 [cs].

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. FreeLB: Enhanced Adversarial Training for Natural Language Understanding, April 2020. arXiv:1909.11764 [cs].