Hard Sample Aware Prompt-Tuning

Yuanjian Xu[1,2*†], **Qi An**[1,2*†], **Jiahuan Zhang**[1*], **Peng Li**[1], **Zaiqing Nie**[1,3‡]
Institute for AI Industry Research (AIR), Tsinghua University[1]
School of Software and Microelectronics, Peking University[2]
Beijing Academy of Artificial Intelligence[3]
{xyj, angel}@stu.pku.edu.cn;
{zhangjiahuan, lipeng, zaiqing}@air.tsinghua.edu.cn

Abstract

Prompt-tuning based few-shot learning has garnered increasing attention in recent years due to its efficiency and promising capability. To achieve the best performance for natural language processing (NLP) tasks with just a few samples, it is vital to include as many informative samples as possible and to avoid misleading ones. However, there is no work in prompt-tuning literature addressing the problem of differentiating informative hard samples from misleading ones in model training, which is challenging due to the lack of supervision signals about the quality of the samples to train a well-performed model. We propose a framework named Hard Sample Aware Prompt-Tuning (HardPT) to solve the non-differentiable problem in hard sample identification with reinforcement learning, and to strengthen the discrimination of the feature space without changing the original data distribution via an adaptive contrastive learning method. An extensive empirical study on a series of NLP tasks demonstrates the capability of HardPT in few-shot scenarios. HardPT obtains new state-of-the-art results on all evaluated NLP tasks, including pushing the SST-5 accuracy to 49.5% (1.1% point absolute improvement), QNLI accuracy to 74.6% (1.9% absolute improvement), NMLI accuracy to 71.5 (0.7% absolute improvement), TACREV F_1-score to 28.2 (1.0 absolute improvement), and i2b2/VA F_1-score to 41.2 (1.3 absolute improvement).

1 Introduction

In recent years, self-supervised pre-trained language models (PLMs) like GPT (Radford et al., 2019), and BERT (Devlin et al., 2019) have gained significant popularity in various natural language processing (NLP) tasks. These PLMs follow a general paradigm of transferring contextual knowledge to specific NLP tasks by fine-tuning model parameters. With the release of GPT-3 (Brown et al., 2020), prompt-tuning has received much attention of late for its outstanding performance in downstream NLP tasks, without the cost of fine-tuning giant PLMs. Particularly, prompt-tuning has shown great superiority in few-shot learning scenarios over fine-tuning methods.

To achieve the best performance for NLP tasks with just a few samples, it is vital to include as many informative samples as possible and to avoid misleading ones. However, there is no work in the literature addressing the problem of differentiating informative hard samples from misleading ones in prompt-tuning.

The concept of hard samples has been discussed in the field of computer vision for object detection and image classification tasks (Shrivastava et al., 2016; Lin et al., 2017). During the training process, high-loss samples are typically selected as hard samples, and their weights are adjusted through resampling or modifying the loss. However, relying solely on the loss function results in poor performance in identifying misleading samples. Additionally, this approach can result in a drift of the original data distribution and overlook the optimization of the sample feature space.

Most recently, Zhu et al. (2022) proposes an approach named EHN which is specifically designed for distinguishing hard samples from noisy samples in the context of histopathology image classification tasks. However, this method relies on additional prior knowledge as supervision signals, making it impractical for real-world applications. Automatically differentiating misleading and hard samples in NLP tasks is challenging due to the lack of supervision signals about the quality of the samples to train a quality classification model. Another

[*] Equal contribution.
[†] Work done during Yuanjian Xu and Qi An's internship at AIR, Tsinghua University.
[‡] Corresponding author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 12356–12369
July 9-14, 2023

work, Dataset Cartography (DC) (Swayamdipta et al., 2020) aims to mine the statistical metrics, confidence, and variability, to evaluate the quality of samples in NLP datasets. However, DC fails to differentiate between misleading and hard samples, categorizing them both as "hard-to-learn". Accurately estimating the quality of NLP samples, particularly non-committal language descriptions, remains challenging.

To address the problem of differentiating informative hard samples from misleading ones in prompt-tuning, we propose a framework named Hard Sample Aware Prompt-Tuning (HardPT). We classify the samples into three categories: **easy**, **hard**, and **misleading** samples. **Easy** samples mean they are easily classified correctly by the model, while the **hard** samples are meant to be difficult for the model to learn correctly. **Misleading** samples refer to harmful samples during data annotation. Due to the lack of supervised signals for sample quality, we design a reinforcement learning network to solve the non-differentiable problem in hard sample identification. To better leverage the identified hard samples, we innovatively propose an adaptive contrastive learning method to strengthen the discrimination of the feature space without changing the original data distribution.

We conduct an extensive empirical study on various NLP tasks to demonstrate the capability of HardPT in few-shot scenarios. Remarkably, HardPT achieved state-of-the-art (SOTA) results across all evaluated NLP tasks, including pushing the SST-5 accuracy to 49.5% (1.1% absolute improvement), improving the QNLI accuracy by 1.9 percentage points, resulting in a noteworthy 74.6%. Moreover, HardPT improves the NMLI accuracy to reach 71.5 (0.7% absolute improvement), TACREV F_1-score to 28.2 (1.0 absolute improvement), and i2b2/VA F_1-score to 41.2 (1.3 absolute improvement). These results highlight the exceptional performance of HardPT in the few-shot scenarios.

Our key contributions can be summarized as follows:

- We introduce the problem of Hard Sample Aware Prompt-Tuning and propose a Reinforcement Learning network to automatically differentiate informative hard samples precisely from misleading samples.
- We propose adaptive contrastive learning to strengthen the discrimination of the feature space without changing the original data distribution.
- The extensive experiments show that HardPT achieves SOTA performance in few-shot scenarios.

2 Related work

Prompt-Tuning: Prompt-tuning (Brown et al., 2020; Zong et al., 2021; Lester et al., 2021; Han et al., 2021; Vu et al., 2022; Liang et al., 2022; Asai et al., 2022) is an efficient way to adapt pretrained language models (PLMs) to downstream tasks without tuning the parameters of PLMs. Depending on the type of prompt, prompt-tuning is divided into two categories: soft prompt and hard prompt. Soft prompt leverage trainable parameters as prompts (Lester et al., 2021; Vu et al., 2022; Asai et al., 2022), while hard prompt employs natural language strings as prompts. With the emergence of GPT-3 (Brown et al., 2020), the hard prompt has gained significant attention in recent years, particularly in the context of few-shot learning. And intensive efforts have been devoted to improving the prompts. Zong et al. (2021) utilize demonstrations to enhance prompt-tuning in few-shot scenarios and achieve improvement on various NLP datasets. Liang et al. (2022) introduce more demonstrations and utilize contrastive learning to compare different demonstrations on the same datasets. Han et al. (2021) propose to construct prompts automatically by combining sub-prompts with logic rules. However, the impact of the samples remains underexplored in prompt-tuning.

Contrastive Learning: Contrastive learning (CL) (Chopra et al., 2005; Yan et al., 2021; Gao et al., 2021b; Li et al., 2022) is an effective method for representation learning that brings samples of the same class closer together while pushing those of different classes apart in the representation space. CL can be divided into unsupervised CL and supervised CL according to whether the pretext task requires labeled data. Unsupervised CL has gained widespread popularity because it reduces the need for labeled data for the model. Yan et al. (2021) utilize CL by generating two distinct augmented versions of the same sentence. They employ four methods as a data augmentation module at the embedding layer. This approach effectively leverages

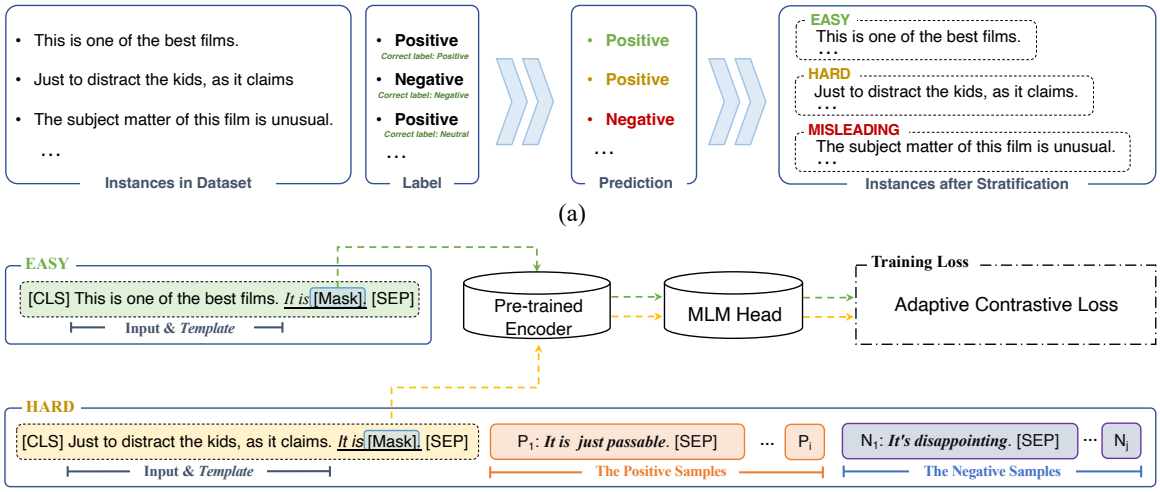

Figure 1: The model architecture of HardPT can be divided into two steps. (a) Samples are classified into easy, hard, and misleading categories by an agent. The labels in the Prediction column indicate the model's prediction accuracy, with green indicating correct, yellow indicating challenging, and red indicating incorrect. (b) To enhance learning on hard samples, a contrast batch is constructed comprising positive and negative samples, represented by the orange and blue boxes respectively. The PLM is employed to classify the hard samples, and the adaptive contrastive loss is used as the training loss.

CL to enhance the diversity and quality of the augmented data. With the help of labeled data, supervised CL can further improve the quality of features. Li et al. (2021) use CL to enhance the performance in sentiment classification tasks. PairSCL (Li et al., 2022) constructs a cross-attention module to enhance feature representations by capturing relationships and similarities among samples. Since all datasets in this paper are labeled, we use the supervised CL method to strengthen the representation of hard samples.

Reinforcement Learning: In order to achieve high performance in model training, a substantial quantity of labeled samples are typically required. However, this abundance of data is often accompanied by a significant amount of noise. The absence of labels in unsupervised learning scenarios can make it difficult to distinguish noise from relevant information. Reinforcement learning (RL) (Sutton and Barto, 2018) techniques have emerged as a promising solution to this challenge. Feng et al. (2018) and Zeng et al. (2018) employ RL to select high-quality training sentences, while Qin et al. (2018) leverage it to identify false-positive samples. Chen et al. (2020) treat a Deep Q-Network (DQN) module as a label denoiser, effectively selecting the most reliable labels. We use REINFORCE (Williams, 1992), a classical algorithm that is also used in Zhang et al. (2021), to obtain the assessment of sample quality.

3 Method

Previous research has demonstrated the effectiveness of prompt-tuning as a practical approach to address few-shot learning in NLP. However, many existing methods only focus on categorizing data into easy and hard samples, overlooking the crucial distinction between hard samples and mislabeled misleading samples. In the context of few-shot learning, failure to distinguish between these two types of samples can lead to underutilization of the value inherent in hard samples and result in a decline in model performance. Therefore, it is imperative to address this issue to fully leverage the potential of hard samples in the few-shot scenario.

To illustrate and address these problems more clearly, this section begins by introducing the fundamental paradigm of prompt-tuning in Section 3.1. Section 3.2 describes the HardPT algorithm, which mainly includes the detection and utilization methods of hard samples. Towards the end of this section, we outline the training process of HardPT.

3.1 Basic Paradigm of Prompt-Tuning

Prompt-tuning: Based on the fine-tuning method for NLP, the sentence is transformed into $x = \{\texttt{[CLS]}, t_1, t_2, \cdots, t_n, \texttt{[SEP]}\}$ by adding a special token `[CLS]` before the first token and `[SEP]` after the last token. `[CLS]` is encoded to a feature for

classification, which aggregates vital information for specific tasks, and [SEP] is a separator between two sentences. Prompt-tuning transforms the NLP tasks into cloze tasks by designing a template containing single or multiple masks connected to the original sentence x. The template can be expressed as $x_{prompt} = \{t_{temp_1}, \cdots, \texttt{[MASK]}, \cdots, t_{temp_m}\}$, and the final input of the PLM is $\mathcal{T}(x) = \{\texttt{[CLS]}, t_1, t_2, \cdots, t_n, t_{temp_1}, \cdots, \texttt{[MASK]}, \cdots, t_{temp_m}, \texttt{[SEP]}\}$. We can infer the classes of samples according to the hidden vector at the [MASK] position through MLP or other classifiers.

3.2 HardPT

In the context of few-shot learning, we contend that harnessing the full potential of hard samples, which the model finds challenging to learn, is just as crucial as accurately classifying easy-to-learn samples and excluding misleading samples that hinder model training. To effectively utilize hard samples, we need to address the following two problems in sequence: identifying hard samples from the datasets and devising effective learning methods that allow the pre-trained language model to incorporate the knowledge contained within them.

The initial challenge in effectively leveraging hard samples is the task of identifying them, especially in scenarios where supervised signals for assessing sample quality are unavailable. To address this issue, we propose an RL-based module, as depicted in Fig.1 (a), that employs agents to tackle unsupervised classification problems related to sample quality.

The main concept is to update the decision-making network based on the better performance observed in the previous exploration, aiming to achieve higher expected returns. These performance improvements are treated as "labels" for training the network.

The agent's reward function is defined as the increment of the F_1-score on the validation set, representing the change of F_1-score when training with or without this module across all samples. During the pre-training stage, each reward obtained triggers a fine-tuning process of the PLM from scratch to prevent error accumulation. The agent utilizes the cross-entropy to update, and the policy update method is expressed by the following formula:

$$\begin{aligned}&\pi_{i+1}(a \mid \boldsymbol{o}) \\ &= \operatorname{argmin}\{-\mathbb{E}_{\mathrm{z}\sim\pi_{\mathrm{i}}(\mathrm{a}|\mathrm{o})}[\phi(\mathrm{z}) \geq \psi_{\mathrm{i}}]\log\pi_{\mathrm{i}+1}(\mathrm{a} \mid \mathrm{o})\}\end{aligned} \tag{1}$$

where $\boldsymbol{o}$ represents the observation at the present moment, π_i represents the policy at the current moment $\boldsymbol{i}$, and $F(\phi(z) \geq \psi_i)$ is defined as the labeling function. $\phi(z)$ represents the increment of F_1-score for one sample set. ψ_i represents the reward threshold at the present moment $\boldsymbol{i}$. ψ_i is a hyperparameter determined by quantile statistics.

$$\begin{cases} F(\phi(z)) = 1, \text{if } \phi(\mathrm{z}) \geq \psi_{\mathrm{i}} \\ F(\phi(z)) = 0, \text{if } \phi(\mathrm{z}) < \psi_{\mathrm{i}} \end{cases} \tag{2}$$

In the training set, each annotated instance is associated with a ground-truth label and a predicted label in each training epoch. By comparing the predicted labels with the ground-truth labels, we construct a set called $\mathcal{N}_{\text{correct}}$ to gather samples with accurate predictions. Additionally, the agent generates a set of hard samples based on its observations, referred to as $\mathcal{N}_{\text{hard}}$.

The second step involves maximizing the utility of the hard samples identified in the first step. Drawing inspiration from CL, our objective is to reduce the distance between hard samples and easy samples with the same label in the feature space. Given that our scenarios involve supervised classification problems, we propose a novel module based on supervised CL. This module addresses the challenge of limited samples in few-shot scenarios by incorporating various methods for constructing multiple positive and negative samples.

For each hard sample x^i_{hard} selected by the agent, we employ three approaches to construct positive and negative samples. The first approach involves random sampling within the same batch. In this method, samples with the same labels as the hard sample are considered positive samples, while the remaining samples are considered negative samples. The second approach uses back-translation. It involves translating data from one language to another and then translating it back to the original language. This process introduces slight differences in expression while maintaining semantic similarity, making it an effective method for constructing positive samples in NLP tasks. These two methods are tailored for sentiment analysis and natural language inference tasks. The last approach involves entity replacement which is designed specifically for relation extraction tasks. Two samples with the same label are selected, and the head entities and tail entities of each sample are interchanged to generate positive and negative samples. By employing these approaches, the batch size of CL can

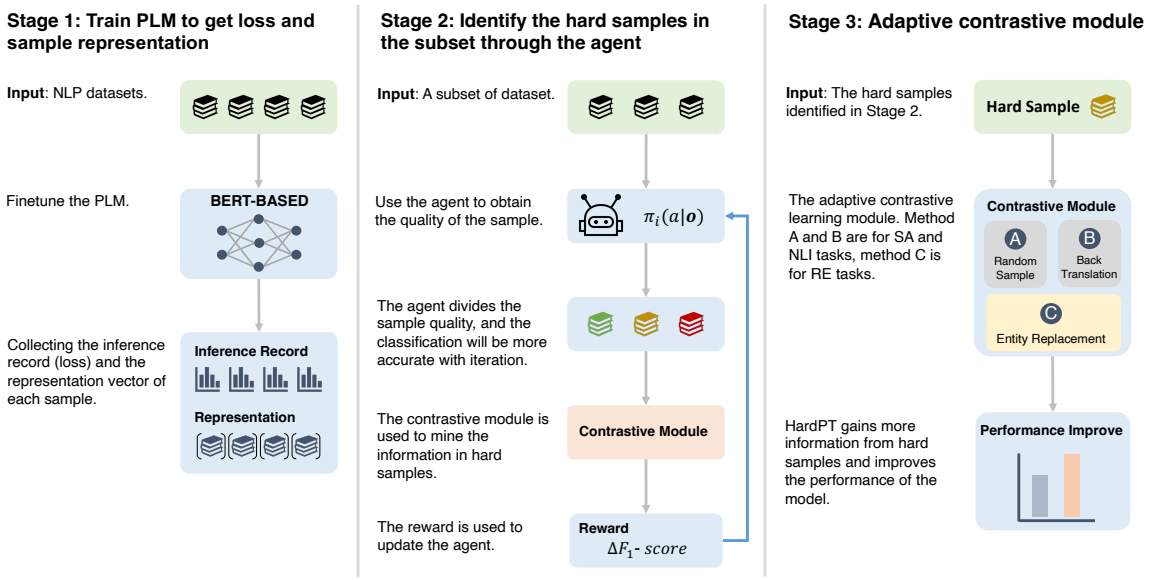

Figure 2: The training process of HardPT consists of three stages. In Stage 1, the training features of hard samples are collected. Stage 2 involves the creation of a few-shot learning scenario and utilizes the features obtained in the previous stage to pre-train the agent. Finally, in Stage 3, positive and negative samples are generated for the hard samples identified in Stage 2 and fed into the network for the final end-to-end training.

be increased.

In Fig.1 (b), the operation after sample construction is depicted. The positive sample and negative sample are concatenated behind the hard sample, forming a complete input fed into the PLM. The goal is to minimize the loss ℓ through backpropagation once the sentence-level feature is obtained. The adaptive contrastive loss is defined as follows:

$$\frac{1}{k}\sum_{i=1}^{k}\left[-\lambda\frac{\exp(D_\theta(x^i_{\text{hard}}, x^i_{\text{neg}}))}{\exp(D_\theta(x^i_{\text{hard}}, x^i_{\text{pos}})) + \exp(D_\theta(x^i_{\text{hard}}, x^i_{\text{neg}}))}\right] + \beta[L_{CE}(x^i_{\text{pos}}) + L_{CE}(x^i_{\text{neg}}) + L_{CE}(x^i_{\text{hard}})] \tag{3}$$

where λ and β are trainable parameters designed to strike a balance between correction and prediction. This module aims to align representations of samples with the same label and separate representations of samples with different labels. The supervised contrastive loss is formed using cross-entropy loss:

$$L_{CE}(x^i_*) = \sum_{C=1}^{M} y^i_C \log p^i_C \tag{4}$$

where k is the total number of samples. $*$ can be replaced by x^i_{hard}, x^i_{pos}, and x^i_{neg}, which are the i-th hard, positive, and negative sample, respectively. M is the total number of categories. y^i_C is a function, when the label of x^i is C, $y^i_C = 1$, otherwise $y^i_C = 0$. $\log p^i_C$ means the probability of observation i belonging to C.

$$\begin{aligned}\vec{S}_i &= \mathcal{F}^\theta_{\text{PLM}}(S_i)\\ \vec{S}_j &= \mathcal{F}^\theta_{\text{PLM}}(S_j)\\ D_\theta(\vec{S}_i, \vec{S}_j) &= 1 - \frac{\vec{S}_i \cdot \vec{S}_j}{\|S_i\|_2 \, \|S_j\|_2}\end{aligned} \tag{5}$$

$\mathcal{F}^\theta_{\text{PLM}}(S_i)$ represents the encoder of the PLM for the `[MASK]` in the input sentence and θ is the trainable parameters. $D_\theta(\vec{S}_i, \vec{S}_j)$ measures the dissimilarity between two vectors $\vec{S}_i$ and $\vec{S}_j$.

3.3 Training Process

As depicted in Fig.2, we divide the overall training method into three stages. In the first stage, the BERT-base model is employed to train all samples. The objective here is to collect the loss of samples during training and extract the encoded features using the PLM. In the second stage, we pre-train the hard sample identification module. We randomly sample from the dataset to create a subset of data that adheres to the few-shot scenario and utilize the agent to identify hard samples within this subset. After multiple epochs, we calculate the agent's reward, which corresponds to the increase in the F_1-score achieved by using the RL module. In the third stage, we perform end-to-end training on the few-shot scenarios of the dataset. Both the agents and PLM are fine-tuned using the information obtained from the previous two stages. For the

	SST-2	SST-5	MR	MPQA	MNLI	QNLI	RTE
BERT	87.7	41.7	79.5	69.6	66.3	66.9	57.4
BERT+Prompt-Tuning	89.1	42.3	83.3	75.3	69.2	67.8	55.1
BERT+Focal loss	76.5	39.8	74.2	50.8	61.6	61.6	53.3
SCL+Prompt-Tuning	91.0	42.8	84.6	85.2	70.3	67.7	64.2
LM-BFF† (Gao et al., 2021a)	91.8	43.9	87.2	84.3	69.7	68.4	68.9
Demo-Tuning† (Liang et al., 2022)	93.2	48.1	88.1	85.8	70.8	72.7	70.2
HardPT	**93.6**	**49.2**	**88.5**	**86.3**	**71.5**	**74.6**	**71.1**

Table 1: The performance of SA and NLI tasks in a few-shot setting with $K = 16$, where K represents the number of samples selected for each category. We randomly sample from the dataset and average the performance. The datasets marked as "†" indicate the reproduction results obtained from the original codes. Accuracy is used as the evaluation metric.

	TACRED	TACREV	Re-TACRED	SemEval	i2b2/VA	DDI
BERT	20.6	25.4	47.4	60.3	22.8	26.7
BERT+Prompt-Tuning	26.6	24.2	50.1	68.0	37.2	35.4
PTR† (Han et al., 2021)	30.7	27.2	51.8	79.1	39.9	38.1
HardPT	**31.1**	**28.2**	**52.1**	**79.9**	**41.2**	**38.7**

Table 2: The performance of RE tasks in a few-shot setting with $K = 16$ and the evaluation metric is F_1-score. For the general scenario, we use the same templates as PTR. Additionally, we design biomedical templates, and their details are in Appendix A.

hard samples identified by the agent, we employ the adaptive contrastive learning method to extract valuable information from these samples. It is to be noticed that the first stage need not be limited to the datasets of current tasks, other available corpora within the domain could also help sample quality representation.

4 Experiment

In this section, we present the experimental setup and results, which are divided into three parts. In Section 4.1, we provide a brief description of the public datasets utilized in this paper. The baseline models employed in the experiments are outlined in Section 4.2. In Section 4.3, we present the results and provide a detailed analysis. For additional information regarding the dataset, experimental conditions, and hyperparameter settings, please refer to Appendices A to C.

4.1 Datasets

To verify the effectiveness of HardPT, we select several representative NLP tasks including Sentiment Analysis (SA), Natural Language Inference (NLI), and sentence-level Relation Extraction (RE).

The SA and NLI tasks utilize datasets selected from the GLUE benchmark (Wang et al., 2019). The SA task involves classifying text based on personal subjective sentiment, categorizing it into positive, negative, or more categories. The SA task consists of binary classification datasets such as SST-2 (Socher et al., 2013), MR (Pang and Lee, 2005), and MPQA (Wiebe et al., 2005). The SST-5 dataset (Socher et al., 2013) is a well-known multi-classification dataset comprising five classes. The NLI task involves predicting the relationship between a given premise proposition and a hypothetical proposition, categorizing it as entailment, neutral, or contradiction. Several well-known NLI datasets we used include MNLI (Williams et al., 2018), QNLI (Rajpurkar et al., 2016), and RTE (Dagan et al., 2005; Bar-Haim et al., 2006; Giampiccolo et al., 2007, 2008; Bentivogli et al., 2009, 2010, 2011).

The sentence-level RE task involves classifying the relations between specified entities in a sentence. We utilized several classical datasets in our study. In the general domain, we employ TACRED (Zhang et al., 2017), TACREV (Alt et al., 2020), and Re-TACRED (Stoica et al., 2021). Additionally, we included the biomedical datasets DDI (Segura-Bedmar et al., 2013) and i2b2/VA (Uzuner et al., 2011).

4.2 Baselines

The baselines for the SA and NLI tasks include vanilla BERT, prompt-tuning based BERT, vanilla

BERT with focal loss, supervised contrastive learning (SCL) models (Gunel et al., 2020), LM-BFF (Gao et al., 2021a), and Demo-Tuning (Liang et al., 2022) (current SOTA). As for the RE task, in addition to vanilla BERT and prompt-tuning based models, we also include PTR as a baseline.

The vanilla BERT serves as a benchmark compared to traditional PLM-based models. The prompt-based model has gained popularity in applying PLMs to few-shot scenarios by constructing task-specific prompts. Focal Loss, originally proposed in the CV field, addresses the challenge of handling hard samples and has been adapted to PLMs to explore its potential. SCL is a fundamental framework based on CL in NLP, which has proven effective for few-shot learning. LM-BFF and Demo-tuning are two prompt-based methods for NLP tasks that have achieved SOTA performance on various established datasets. PTR, a classic prompt-based method, is specifically tailored for the RE task. Evaluation measures for the SA and NLI tasks are accuracy, while the RE task is evaluated using the F_1-score.

4.3 Experimental Results

The performance of HardPT and the baselines are presented in Table 1 and Table 2. The PLM-based prompt-tuning exhibits significant superiority over traditional fine-tuning methods across all tasks. This compelling result proves that prompt-tuning can further enhance the utilization efficiency of PLM in few-shot scenarios, enabling them to effectively absorb knowledge from the training set.

In SA and NLI tasks, using focal loss as the fine-tuning objective function of fine-tuning does not effectively address the challenge of hard samples within PLMs. This suggests that inference-based confidence measures are inadequate for assessing the difficulty of natural language samples. Moreover, altering the sample distribution through weighting in the loss function may result in training target deviations for PLMs. HardPT addresses these limitations by constructing positive and negative samples with a specific target, thereby enhancing the model's capabilities. Notably, HardPT achieves improved performance on PLMs under the few-shot setting while maintaining template simplicity and consistency. Our results, as shown in Table 1, demonstrate new SOTA performance across all evaluated datasets, with notable achievements including a 1.1% absolute accuracy improvement on SST-5 and a 1.9% absolute accuracy improvement on QNLI.

Dataset	Random Stratification	Loss Ranking	HardPT
SST-2	34.1	74.1	**93.6**
SST-5	20.6	35.3	**49.2**
MR	54.2	69.6	**88.5**
TACRED	52.7	62.9	**69.8**
TACREV	61.9	72.8	**79.1**
Re-TACRED	71.5	85.6	**90.5**
SemEval	73.5	84.2	**89.1**

Table 3: Comparison of hard sample identification methods in few-shot learning scenarios. Random stratification refers to the scenario where hard samples are not specifically selected. Loss ranking involves selecting hard samples based on the sorting of sample losses in the model.

As presented in Table 2, in RE tasks we achieve a notable improvement in the F_1-score for TACREV, with an increase of 1.0. Additionally, we observe an improvement of 1.3 in the F_1-score for i2b2/VA and a 0.6 improvement for DDI. These results on the i2b2/VA and DDI datasets highlight the strong transferability of HardPT, demonstrating its ability to deliver exceptional performance not only in the general domain but also in specific domains.

The results demonstrate that HardPT effectively distinguishes hard samples from misleading ones and utilizes CL to mine hard samples, thereby significantly enhancing sample utilization efficiency in few-shot scenarios on top of prompt-tuning.

5 Analysis

To validate the effectiveness of each component in HardPT and its robustness against noise, we conducted several controlled experiments. Firstly, we conducted ablation experiments to verify the impact of hard sample identification and the sample augmentation method in contrastive learning. Secondly, we performed experiments to assess the algorithm's robustness in the presence of noise.

5.1 Ablation Experiments

We conduct two ablation experiments: the first one validates the effectiveness of hard sample identification, and the second one examines the impact of using different languages for back-translation in the adaptive contrastive module.

Effectiveness of hard sample identification module. To validate the effectiveness of the hard sample identification module, we establish two control groups. The first group is the random stratification, where the samples are randomly divided into easy, hard, and misleading categories, without any specific identification of hard samples by the model. The second group involves partitioning the samples based on training loss ranking, which represents a basic mechanism for selecting potentially hard samples. In this case, we set the quantiles for easy and hard samples at 0.3 and 0.7, respectively.

Table 3 demonstrates that the hard sample identification used in HardPT outperforms both random stratification and loss ranking alone. This indicates that the agent has learned relevant sample quality features. There are two reasons why using loss ranking alone is less effective. Firstly, HardPT incorporates more input information by considering both sample features and training loss comprehensively, whereas loss ranking only focuses on the loss and overlooks sample features. Secondly, the loss ranking method may confuse hard samples with misleading samples. Both hard samples and misleading samples exhibit higher losses during training, but loss ranking methods fail to effectively differentiate between them.

Choice of back-translation method. In the adaptive contrastive learning module, we investigate the importance of language selection in constructing positive and negative samples. Given that the original dataset is in English, we compare the impact of using French, a language close to English, and Vietnamese, a language significantly different from English. As depicted in Table 4, employing Vietnamese as an intermediate language introduces considerable bias in back-translation, resulting in unnecessary additional noise. Hence, when designing the back-translation module, it is crucial to consider language differences, and using similar languages may be more suitable for constructing positive and negative samples.

5.2 Robustness Experiment

In order to mitigate the potential influence of mislabeled labels in the original dataset, we carefully select two datasets, SST-2 and SemEval, known for their high-quality labels. To simulate the impact of noise in real-world scenarios, we introduce artificial noise into the few-shot scenario of SST-2 and SemEval at a ratio of 10%. The results presented in Table 5 demonstrate that all models experience a decline in performance in the presence of noise. However, when compared to vanilla BERT and prompt-tuning based BERT, HardPT exhibits superior resistance to noise and greater robustness. These findings also indicate that the prompt-tuning method carries the risk of magnifying the influence of misleading samples in noisy scenarios.

Dataset	Vietnamese	French
SST-2	85.1	**93.6**
SST-5	39.6	**49.2**
MR	84.2	**88.6**

Table 4: The impact of different back translation methods on SA task performance. We examine the effect of using different intermediate languages, specifically French and Vietnamese, on the construction of positive and negative samples.

Model	SST-2 (with 10% noise)	SemEval (with 10% noise)
BERT	83.2 (-5.13%)	71.9 (-17.5%)
Prompt-tuning	80.9 (-9.20%)	75.3 (-15.5%)
HardPT	85.5 (-8.60%)	80.1 (-12.3%)

Table 5: Robustness verification results in the noisy scenario. To simulate real-world noisy datasets, SST-2 and SemEval are randomly injected with 10% noise, i.e., incorrect labels. We compared the performance of our model with two baselines: vanilla BERT and prompt-tuning based BERT.

6 Conclusions

This paper proposes HardPT, the first prompt-tuning framework for hard sample identification and utilization. HardPT focuses on the influence of sample quality on the model in the few-shot scenario based on prompt-tuning. Our method can distinguish hard samples from misleading samples without data quality labels, and mine the information contained in hard samples using contrastive learning based on the features of hard samples. An extensive empirical study on a series of NLP tasks demonstrates the capability of HardPT in few-shot scenarios. HardPT obtains new SOTA results on all evaluated NLP tasks, including pushing the SST-5 accuracy to 49.5% (1.1% point absolute improvement), QNLI accuracy to 74.6% (1.9% absolute improvement), NMLI accuracy to 71.5 (0.7% absolute improvement), TACREV F_1-score to 28.2 (1.0 absolute improvement), and i2b2/VA F_1-score to 41.2 (1.3 absolute improvement).

Limitation

In HardPT, we focus on training specifically on hard samples while discarding misleading samples. However, it is worth acknowledging that these misleading samples may potentially contain valuable information. Additionally, finding quantifiable and interpretable evaluation metrics to accurately assess the model's ability to identify misleading and hard samples is a crucial challenge. In our future work, we plan to explore strategies for correcting mislabeled samples and develop evaluation metrics that accurately measure the accuracy of sample partitioning. Our aim is to maximize the utilization of all available information from the original dataset.

Acknowledgements

This work is jointly supported by the National Key R&D Program of China (No. 2022YFF1203002) and the Beijing Academy of Artificial Intelligence (BAAI).

References

Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. 2020. TACRED revisited: A thorough evaluation of the TACRED relation extraction task. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1558–1569. Association for Computational Linguistics.

Akari Asai, Mohammadreza Salehi, Matthew Peters, and Hannaneh Hajishirzi. 2022. ATTEMPT: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6655–6672, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognising textual entailment challenge.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2010. The sixth PASCAL recognizing textual entailment challenge. In *Proceedings of the Third Text Analysis Conference, TAC 2010, Gaithersburg, Maryland, USA, November 15-16, 2010*. NIST.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2011. The seventh PASCAL recognizing textual entailment challenge. In *Proceedings of the Fourth Text Analysis Conference, TAC 2011, Gaithersburg, Maryland, USA, November 14-15, 2011*. NIST.

Luisa Bentivogli, Bernardo Magnini, Ido Dagan, Hoa Trang Dang, and Danilo Giampiccolo. 2009. The fifth PASCAL recognizing textual entailment challenge. In *Proceedings of the Second Text Analysis Conference, TAC 2009, Gaithersburg, Maryland, USA, November 16-17, 2009*. NIST.

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. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Tiantian Chen, Nianbin Wang, Ming He, and Liu Sun. 2020. Reducing wrong labels for distantly supervised relation extraction with reinforcement learning. *IEEE Access*, 8:81320–81330.

Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pages 539–546.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges, Evaluating Predictive Uncertainty, Visual Object Classification and Recognizing Textual Entailment, First PASCAL Machine Learning Challenges Workshop, MLCW 2005, Southampton, UK, April 11-13, 2005, Revised Selected Papers*, volume 3944 of *Lecture Notes in Computer Science*, pages 177–190. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th*

AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, pages 5779–5786. AAAI Press.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021a. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021b. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 6894–6910. Association for Computational Linguistics.

Danilo Giampiccolo, Hoa Trang Dang, Bernardo Magnini, Ido Dagan, Elena Cabrio, and Bill Dolan. 2008. The fourth PASCAL recognizing textual entailment challenge. In *Proceedings of the First Text Analysis Conference, TAC 2008, Gaithersburg, Maryland, USA, November 17-19, 2008*. NIST.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL@ACL 2007 Workshop on Textual Entailment and Paraphrasing, Prague, Czech Republic, June 28-29, 2007*, pages 1–9. Association for Computational Linguistics.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. *ArXiv*, abs/2011.01403.

Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2021. PTR: prompt tuning with rules for text classification. *CoRR*, abs/2105.11259.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shuang Li, Xuming Hu, Li Lin, and Lijie Wen. 2022. Pair-level supervised contrastive learning for natural language inference. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8237–8241.

Zhengyan Li, Yicheng Zou, Chong Zhang, Qi Zhang, and Zhongyu Wei. 2021. Learning implicit sentiment in aspect-based sentiment analysis with supervised contrastive pre-training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 246–256, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xiaozhuan Liang, Ningyu Zhang, Siyuan Cheng, Zhen Bi, Zhenru Zhang, Chuanqi Tan, Songfang Huang, Fei Huang, and Huajun Chen. 2022. Contrastive demonstration tuning for pre-trained language models. *CoRR*, abs/2204.04392.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2999–3007. IEEE Computer Society.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, pages 115–124. The Association for Computer Linguistics.

Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Robust distant supervision relation extraction via deep reinforcement learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2137–2147. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392. The Association for Computational Linguistics.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. Semeval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013). In *Proceedings of the 7th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2013, Atlanta, Georgia, USA, June 14-15, 2013*, pages 341–350. The Association for Computer Linguistics.

Abhinav Shrivastava, Abhinav Gupta, and Ross B. Girshick. 2016. Training region-based object detectors with online hard example mining. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 761–769. IEEE Computer Society.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment

treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1631–1642. ACL.

George Stoica, Emmanouil Antonios Platanios, and Barnabás Póczos. 2021. Re-tacred: Addressing shortcomings of the TACRED dataset. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13843–13850. AAAI Press.

Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. 2020. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Conference on Empirical Methods in Natural Language Processing*.

Özlem Uzuner, Brett R. South, Shuying Shen, and Scott L. DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *J. Am. Medical Informatics Assoc.*, 18(5):552–556.

Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou', and Daniel Cer. 2022. SPoT: Better frozen model adaptation through soft prompt transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5039–5059, Dublin, Ireland. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Lang. Resour. Evaluation*, 39(2-3):165–210.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. Consert: A contrastive framework for self-supervised sentence representation transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 5065–5075. Association for Computational Linguistics.

Xiangrong Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Large scaled relation extraction with reinforcement learning. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5658–5665. AAAI Press.

Junzi Zhang, Jongho Kim, Brendan O'Donoghue, and Stephen Boyd. 2021. Sample efficient reinforcement learning with reinforce. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12):10887–10895.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 35–45. Association for Computational Linguistics.

Chuang Zhu, Wenkai Chen, Ting Peng, Ying Wang, and Mulan Jin. 2022. Hard sample aware noise robust learning for histopathology image classification. *IEEE Trans. Medical Imaging*, 41(4):881–894.

Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors. 2021. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*. Association for Computational Linguistics.

A Datasets

Table A and Table B present the statistics of each dataset used in this study. They provide information on the task type, as well as the number of samples in the training and testing datasets. Specifically, Table A displays the datasets information in SA and NLI tasks, while Table B presents the datasets information in RE tasks. The templates of i2b2/VA dataset are shown in Table C. The template of DDI dataset is "The relationship of $\texttt{[MASK]}_1$ and $\texttt{[MASK]}_2$ is $\texttt{[MASK]}_3$".

Dataset	Task type	#Train	#Test	$\lvert\mathcal{Y}\rvert$
SST-2	sentiment	6,920	872	2
SST-5	sentiment	8,544	2,210	5
MR	sentiment	8,662	2,000	2
MPQA	opinion polarity	8,606	2,000	2
MNLI	NLI	392,702	9,815	3
QNLI	NLI	104,743	5,463	2
RTE	NLI	2,490	277	2

Table A: The statistics of the datasets used in this work. $\lvert\mathcal{Y}\rvert$ denotes number of classes. In our few-shot setting, we only sample $K \times \lvert\mathcal{Y}\rvert$ examples in $\mathcal{D}_{\text{train}}$.

Dataset	Task type	#Train	#Val	#Test	$\lvert\mathcal{Y}\rvert$
TACRED	RE	68,124	22,631	15,509	42
TACREV	RE	68,124	22,631	15,509	42
RE-TACRED	RE	58,465	19,584	13,418	40
SEMEVAL	RE	6,507	1,493	2,717	19
I2B2/VA	RE	8,184	2,047	19,114	9
DDI	RE	22,232	5,559	5,716	5

Table B: The RE datasets evaluated in this work. $\lvert\mathcal{Y}\rvert$ denotes number of classes. In our few-shot setting, we only sample $K \times \lvert\mathcal{Y}\rvert$ examples in $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{val}}$, respectively.

Class Label	$\texttt{[MASK]}_1$	$\texttt{[MASK]}_2$	$\texttt{[MASK]}_3$
TrIP	treatment	is beneficial for	problem
TrWP	treatment	is useless for	problem
TrCP	treatment	is cause for	problem
TrAP	treatment	is treatment for	problem
TrNAP	treatment	is avoided because	problem
TeRP	test	has revealed the	problem
TeCP	test	is for detecting	problem
PIP	problem	is relevant of	problem
None	entity	is irrelevant of	entity

Table C: The relations contained in the i2b2/VA dataset, and specific templates corresponding to each relation. Combined with the template, the input to the model is: "<*S*>. The $\texttt{[MASK]}_1$ $\texttt{[MASK]}_2$ $\texttt{[MASK]}_3$."

B Environment of Experiments

The experimental environment is equipped with 32 V100 GPUs, and approximately 5000 GPU hours are allocated on average to train a single model.

C Parameters of Experiments

Hyperparameters: We maintain consistent and neutral settings across all experiments to mitigate potential bias in the experimental results attributed to hyperparameters. Our model experiments employ fixed hyperparameters, while other models adhere to the original settings outlined in their respective papers.

In the experiments, our batch size is set to 16. All the models used AdamW as an optimizer. The learning rates of the agent model are set to 3e-6 and 3e-5 in GLUE benchmarks and RE tasks, respectively. In contrastive learning, the learning rates are set to 3e-4 and 1e-6. Initially, the trainable parameters in the loss function, responsible for balancing the contrastive loss and cross-entropy loss, are set to 1.0.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
The description of limitations has been placed at the end of the paper, please see Limitation section for more details.

☑ A2. Did you discuss any potential risks of your work?
The description of limitations has been placed at the end of the paper, please see Limitation section for more details.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
In the Abstract and Introduction section, we summarize the main idea and core idea of this paper.

☒ A4. Have you used AI writing assistants when working on this paper?
We did not use any AI writing assistants that need to be disclosed in the article.

B ☑ Did you use or create scientific artifacts?

We used the datasets produced by predecessors, such as part of GLUE, and the data used was explained in the first subsection of the Experiment.

☑ B1. Did you cite the creators of artifacts you used?
We cite that work in our citation.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
We have made statistics and explanations of the data used in the appendix.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

The results of Experiments are described in detail.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
We describe this in the Appendix.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
We describe this in the Appendix.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Relevant statistics about the experimental results are explained in the Experiment.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.