Adversarial Soft Prompt Tuning for Cross-Domain Sentiment Analysis

Hui Wu[12] and **Xiaodong Shi**[123]*
[1]Department of Artificial Intelligence, School of Informatics, Xiamen University, China
[2]National Institute for Data Science in Health and Medicine, Xiamen University, China
[3]Key Laboratory of Digital Protection and Intelligent Processing of Intangible Cultural Heritage of Fujian and Taiwan (Xiamen University), Ministry of Culture and Tourism, China
huistudent@stu.xmu.edu.cn, mandel@xmu.edu.cn

Abstract

Cross-domain sentiment analysis has achieved promising results with the help of pre-trained language models. As GPT-3 appears, prompt tuning has been widely explored to enable better semantic modeling in many natural language processing tasks. However, directly using a fixed predefined template for cross-domain research cannot model different distributions of the [MASK] token in different domains, thus making underuse of the prompt tuning technique. In this paper, we propose a novel **Ad**versarial **S**oft **P**rompt **T**uning method (AdSPT) to better model cross-domain sentiment analysis. On the one hand, AdSPT adopts separate soft prompts instead of hard templates to learn different vectors for different domains, thus alleviating the domain discrepancy of the [MASK] token in the masked language modeling task. On the other hand, AdSPT uses a novel domain adversarial training strategy to learn domain-invariant representations between each source domain and the target domain. Experiments on a publicly available sentiment analysis dataset show that our model achieves new state-of-the-art results for both single-source domain adaptation and multi-source domain adaptation.

1 Introduction

In recent years, with the emergence of a series of large-scale pre-trained language models (PLMs), such as GPT (Radford et al., 2018, 2019), BERT (Devlin et al., 2019), and RoBERTa (Liu et al., 2019), fine-tuning PLMs has achieved promising results on a wide range of natural language processing (NLP) tasks. However, as PLMs become larger and larger, fine-tuning larger PLMs becomes more challenging in most real-world applications. More recently, Brown et al. (2020) show that designing task descriptions (a.k.a. prompts) can make accurate predictions without updating any of the parameters of GPT-3 (which has 175B parameters). This inspires a new PLM-tuning method named "**prompt tuning**". Such prompt tuning method has achieved state-of-the-art results on text classification and natural language inference (Schick and Schütze, 2020; Schick et al., 2020; Gao et al., 2020), relation classification (Han et al., 2021), and natural language generation (Li and Liang, 2021).

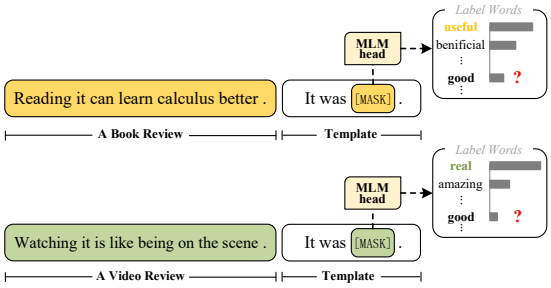

Figure 1: How domain discrepancy affects prompt tuning. Examples of a book review on the top and a video review on the bottom.

It is common to use a predefined template (e.g., "It was [MASK].") in prompt tuning for binary sentiment analysis, and the classification results of positive or negative depend on the probabilities of predefined label words (e.g., "{good, bad}") in the masked language modeling (MLM) task. However, the distributions of MLM prediction results can be different for different domains. An example is shown in Figure 1, the discrepancy between book-domain review and video-domain review leads to different possibilities of label words. The high-frequency label word in book-domain review is "*useful*", and video-domain review is "*real*", neither of which is in the predefined "{good, bad}". Therefore, it is unreasonable to predict predefined label words with fixed templates (a.k.a. hard prompts) for different domain datasets.

The intuition is that the feature distributions corresponding to the [MASK] position learned from the hard prompt are distinct among different do-

*Corresponding author.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 2438 - 2447

mains. And the discrepancy among different domains can have serious effects on the cross-domain setting where we train a classifier on source domain data, e.g., the book reviews, and test it on the target domain, e.g., the video review. So domain adaptation (Ben-David et al., 2007; Mansour et al., 2009) based on cluster hypothesis (Zhu and Goldberg, 2009) becomes a key point of the cross-domain research.

In order to improve the cross-domain sentiment analysis with the help of PLMs, we propose AdSPT: an **Ad**versarial **S**oft **P**rompt **T**uning method, which sheds new light on solving the domain adaptation problem. Specifically, we use soft prompts composed of multiple learnable vectors and the [MASK] token instead of hard templates for tuning. For different domains, we use independent soft prompts to represent domain-specific information, thus making them have the *domain-aware* knowledge. With different domain soft prompts, the MLM head classifier can mitigate the domain discrepancy of the [MASK] token. To enhance the effectiveness of the target domain, we design a novel adversarial training strategy to learn the *domain-invariant* knowledge of the [MASK] token, which can be seen as a two-player minimax game between the target domain and each source domain under multi-source domain adaptation setting. As a result, the collaborative effect of soft prompt tuning and domain adversarial training can more properly predict the feature distribution of the [MASK] token on the ground of domain-specific soft prompts and the domain invariance of the [MASK] token.

In experiments, we evaluate on a publicly available sentiment analysis dataset for both single-source domain adaptation and multi-source domain adaptation. Our results show the effectiveness of collaboratively leveraging domain-specific soft prompts tuning and domain adversarial training. To summarize, the main contributions of this work are as follows:

(1) In prompt tuning, we adopt separate soft prompts to learn embeddings enriched with the domain knowledge, thus alleviating the domain discrepancy of the [MASK] position.

(2) We design a novel adversarial training strategy to learn the domain-invariant representation of the [MASK] position.

(3) Experiments on the Amazon reviews dataset show our method AdSPT obtains the average accuracy 93.14% (0.46 absolute improvement) under single-source domain adaptation and the average accuracy 93.75% (0.81 absolute improvement) under multi-source domain adaptation.

2 Related Work

Prompt tuning. Fine-tuning PLMs with task-specific heads on downstream tasks has become the main paradigm and yields strong performance on many NLP tasks (Peters et al., 2018; Devlin et al., 2019; Radford et al., 2019). But there is a big gap between the fine-tuning objectives of downstream tasks and the pre-training objectives of PLMs, which could limit the exploitation of knowledge in PLMs (Liu et al., 2021b). Subsequently, GPT-3 (Brown et al., 2020) brings a new paradigm "prompt tuning" for downstream tasks, which leverages natural-language prompts and task demonstrations as context to make downstream tasks similar to language modeling.

Early works explore manually defined templates (a.k.a. hard templates) for text classification and natural language inference (Schick and Schütze, 2020, 2021). However, suitable templates require strong domain knowledge. Therefore, some automatically generated hard templates are explored (Shin et al., 2020; Gao et al., 2020; Ben-David et al., 2021). Since prompt construction is to find a method that allows PLMs to effectively perform downstream tasks, it is not necessary to limit templates to human-interpretable natural language. Some works attempt to perform prompting directly with several learnable vectors, such as soft prompt (Lester et al., 2021; Vu et al., 2021), prefix-tuning (Li and Liang, 2021) and P-tuning V2 (Liu et al., 2021a). Moreover, Schick et al. (2020) explore automatically identifying label words. Hu et al. (2021) use an external knowledge base to expand label words. This paper focuses on improving the cross-domain sentiment analysis via different soft prompts of different domains.

Domain Adaptation. Research on *domain adaptation* (DA) uses labeled or unlabeled target data to transfer labeled source information to a specific target domain (Pan and Yang, 2009; Mansour et al., 2009). Popular methods for unsupervised DA are based on domain discrepancy optimizing based on adversarial training (Ganin et al., 2016; Zhao et al., 2018; Saito et al., 2018). As for cross-domain sentiment analysis, some early works use pivot-based methods to capture the shared feature representation of different domains (Yu and Jiang, 2016; Ziser

and Reichart, 2018; Li et al., 2018; Peng et al., 2018). Some other works adopt different adversarial learning methods to learn the domain-common sentiment knowledge (Li et al., 2017; Qu et al., 2019; Li et al., 2019).

Recently, with the promising performance of PLMs in NLP, many works on cross-domain sentiment analysis focus on how to improve langange model pre-training and fine-tuning, e.g., Du et al. (2020) use a target domain MLM task and a domain-distinguish task in pre-training; Zhou et al. (2020) utilize several pre-training tasks based on existing lexicons and annotations. Different from these works, our method is the first to use the combination of soft prompt tuning and adversarial training to solve the DA problem.

3 Problem Formulation

In this paper, we study cross-domain sentiment analysis in the unsupervised domain adaptation setting which contains two scenarios: a source domain and a target domain or multiple source domains and a target domain. Given $m(m \geq 1)$ source domains, the l-th ($l \in [1, \ldots, m]$) source domain contains an annotated dataset $\mathcal{S}_l = \{\boldsymbol{x}_i^s, y_i^s\}_{i=1}^{N_l^s}$, where $\boldsymbol{x}_i^s = [w_1^s, \ldots, w_n^s]$ is a input sentence with n words, y_i^s is the corresponding polarity label, and N_l^s represents the number of examples of the l-th source domain. In the target domain, there is an unannotated dataset $\mathcal{T} = \{\boldsymbol{x}_i^t\}_{i=1}^{N^t}$, where $\boldsymbol{x}_i^t = [w_1^t, \ldots, w_n^t]$ is an unlabeled sentence of the target domain and N^t is the number of the unlabeled data. The goal of cross-domain sentiment analysis is to learn a function $\mathcal{F}$ that could both retain in-domain knowledge for different domains and also learn the domain invariance between the target domain and each source domain to better predict the polarity of unlabeled sentences from the target domain.

4 Method

In this section, we first introduce a soft prompt tuning method for sentiment classification that utilizes soft prompts to capture domain-specific knowledge. Then we present a domain adversarial training method for domain adaptation. Finally, we describe the overall learning procedure.

4.1 Soft Prompt Tuning for Sentiment Classification

Prompt tuning is an approach to add extra information for PLMs by reformulating downstream tasks as cloze questions. The primary components include a template and a set of label words, where the template is a background description of current task and the label words are the high-probability vocabulary predicted by PLMs in the current context. In the binary sentiment classification, we denote the input sentence as $\boldsymbol{x} = [w_1, \ldots, w_n]$, the output label as y. Here $y \in \mathcal{Y}$, and the label space $\mathcal{Y} = \{\text{positive}, \text{negative}\}$.

Prompt tuning formalizes the classification task into a MLM task. Given a PLM $\mathcal{M}$ and its vocabulary $\mathcal{V}$, a prompt consists of a template function $T(\cdot)$ that converts the input sentence $\boldsymbol{x}$ to a prompt input $\boldsymbol{x}_{prompt} = T(\boldsymbol{x})$ with the [MASK] token and a set of label words $\mathcal{V}^* \subset \mathcal{V}$, which are connected with the label space through a mapping function $v : \mathcal{Y} \mapsto \mathcal{V}^*$. As shown in Figure 2, the soft prompted input $\boldsymbol{x}_{prompt}$ contains the embeddings of the original sentence $\mathbf{e}(\boldsymbol{x})$, k learnable vectors $[\mathbf{h}_0, \ldots, \mathbf{h}_{k-1}]$, the embedding of the [MASK] token $\mathbf{e}(\text{“[MASK]”})$, and the embeddings of two positional tokens $\mathbf{e}(\text{“[CLS]”})$ and $\mathbf{e}(\text{“[SEP]”})$. So the actual input of $\mathcal{M}$ is represented as:

$$\boldsymbol{x}_{prompt} = \big[\mathbf{e}(\text{“[CLS]”}), \mathbf{e}(\boldsymbol{x}), \mathbf{h}_0, \ldots, \mathbf{h}_{k-1}, \mathbf{e}(\text{“[MASK]”}), \mathbf{e}(\text{“[SEP]”})\big] \quad (1)$$

where $\mathbf{e}(\cdot)$ represents the embedding function of $\mathcal{M}$.

Here we can denote a PLM $\mathcal{M}$ as a function mapping from $\boldsymbol{x}_{prompt}$ to the feature representation and vocabulary distribution of the [MASK] token, represented as:

$$\mathbf{h}_{[\text{MASK}]}, \mathbf{s}_{[\text{MASK}]} = \mathcal{M}(\boldsymbol{x}_{prompt}) \quad (2)$$

where $\mathbf{h}_{[\text{MASK}]} \in \mathbb{R}^h$ and $\mathbf{s}_{[\text{MASK}]} \in \mathbb{R}^{|\mathcal{V}|}$ are the hidden representation and vocabulary distribution of the [MASK] token respectively, and $\mathbf{s}_{[\text{MASK}]} = f(\mathbf{h}_{[\text{MASK}]})$ is obtained by the MLM head function f.

The probability $p(y|\boldsymbol{x})$ is formalized according to the distribution of the label word $w \in \mathcal{V}^*$ w.r.t. the [MASK] position. In binary sentiment classification, we set the label words as $\mathcal{V}^* =$

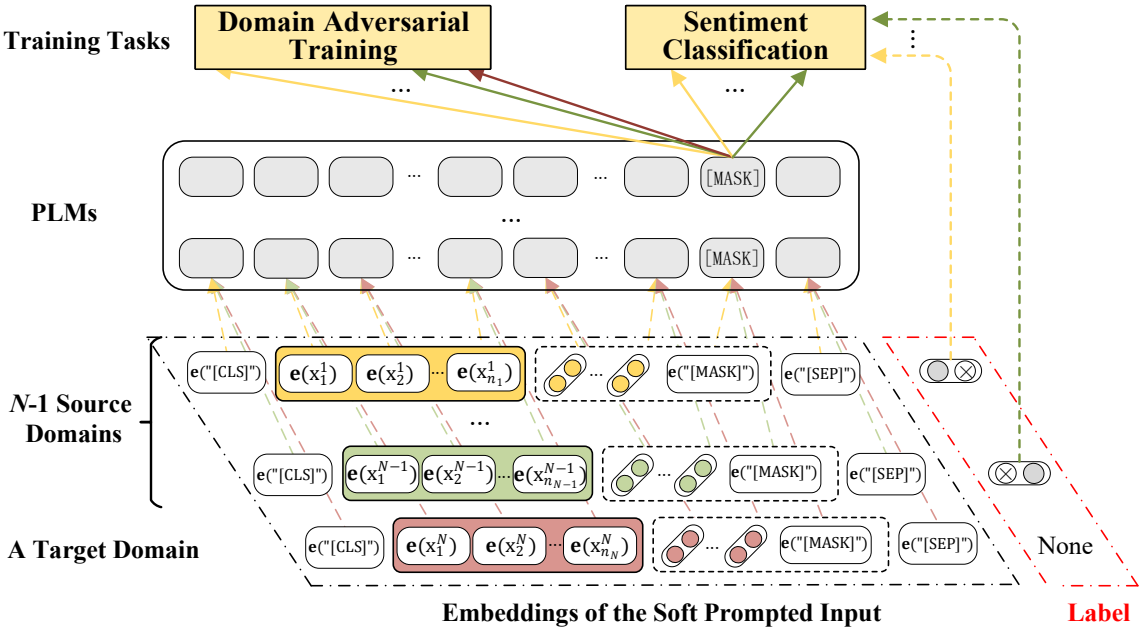

Figure 2: Overall structure of the proposed method.

$\{\text{good}, \text{bad}\}$. So,

$$p(y|\boldsymbol{x}) = p(\mathcal{V}_y^* \leftarrow [\text{MASK}]\,|\boldsymbol{x}_{prompt}) = \frac{\exp(\mathbf{s}_{[\text{MASK}]}(\mathcal{V}_y^*))}{\sum_{y' \in \mathcal{Y}} \exp(\mathbf{s}_{[\text{MASK}]}(\mathcal{V}_{y'}^*))} \quad (3)$$

Given an annotated dataset $\mathcal{S} = \{\boldsymbol{x}_i, y_i\}_{i=1}^N$, the training objective for soft prompt tuning is obtained using the binary cross-entropy loss,

$$\mathcal{L}_{class}(\mathcal{S}; \theta_{\mathcal{M},p,f}) = -\sum_{i=1}^{N} \Big[\log p(y_i|\boldsymbol{x}_i)^{\mathbb{I}\{\hat{y}_i=1\}} + \log(1 - p(y_i|\boldsymbol{x}_i))^{\mathbb{I}\{\hat{y}_i=0\}} \Big] \quad (4)$$

where $\hat{y}_i$ represents the ground truth label ranging from 1 as the positive label and 0 as the negative label). $\theta_{\mathcal{M},p,f}$ represents the overall trainable parameters of the PLM $\mathcal{M}$, several learnable vectors p and the MLM head function f.

4.2 Domain Adversarial Training

For the same task in different domains, domain adversarial training can not only transfer the generic knowledge from source domains to the target domain, but also train more domain-aware classifiers. As shown in Figure 2, domain adversarial training aims to make the feature distributions of the [MASK] position from different domains closer. More intuitively, it will encourage the MLM head classifer to obtain domain-invariant features across domains.

Based on the hidden representation $\mathbf{h}_{[\text{MASK}]}$ by the PLM, the detailed process of domain adversarial training is as follows: given m ($m \geq 1$) source domains, we assume that between each source domain $\mathcal{S}_l$ ($l \in [1, \ldots, m]$) and the target domain $\mathcal{T}$ have a domain discriminative function $g_l : \mathbb{R}^h \rightarrow \mathcal{D}$ that discriminates between the source domain and the target domain, where the domain label set is represented as $\mathcal{D} = \{0, 1\}$, 0 is the source domain label, and 1 is the target domain label. To this end, there are m domain discriminators, denoted as $\mathbf{g} = \{g_l\}_{l=1}^m$.

Given an input example $\boldsymbol{x}$ from either the l-th ($l \in [1, \ldots, m]$) source domain or the target domain, we first obtain the task-specific head representation $\mathbf{h}_{[\text{MASK}]}$ by $\mathcal{M}$ and then model the probability $p(d|\boldsymbol{x})$ for discriminating the domain label $d \in \mathcal{D}$ as:

$$p(d|\boldsymbol{x}) = \frac{\exp(g_l^d(\mathbf{h}_{[\text{MASK}]}))}{\sum_{d' \in \mathcal{D}} \exp(g_l^{d'}(\mathbf{h}_{[\text{MASK}]}))} \quad (5)$$

Given m source domain dataset $\hat{\mathcal{S}} = \{\mathcal{S}_l\}_{l=1}^m = \{\{\boldsymbol{x}_i^s\}_{i=1}^{N_l^s}\}_{l=1}^m$ and a target domain dataset $\mathcal{T} = \{\boldsymbol{x}_i^t\}_{i=1}^{N^t}$, where N_l^s is the number of samples in the l-th source domain and N^t is the number of samples in the target domain, the domain discriminative objective is to minimize the following cross-

entropy loss,

$$\begin{aligned}&\mathcal{L}_{domain}(\hat{\mathcal{S}}, \mathcal{T}; \theta_{\mathcal{M},p,\mathbf{g}}) \\ &= -\sum_{l=1}^{m}\sum_{i=1}^{N_l^s+N^t}\Big[\log p(d_i|\boldsymbol{x}_i)^{\mathbb{I}\{\hat{d}_i=1\}} \\ &\qquad + \log(1-p(d_i|\boldsymbol{x}_i))^{\mathbb{I}\{\hat{d}_i=0\}}\Big]\end{aligned} \tag{6}$$

where $\hat{d}_i$ represents the truth domain label and $\theta_{\mathcal{M},p,\mathbf{g}}$ represents the overall trainable parameters of the PLM $\mathcal{M}$, several learnable vectors p and m domain discriminators $\mathbf{g}$.

The domain adversarial training among m source domains and the target domain can be seen as a two-player minimax game where the domain classifiers $\mathbf{g} = \{g_l\}_{l=1}^m$ tend to minimize the domain discrimination loss so as to make the domain discriminators strong while the PLM $\mathcal{M}$ tends to maximize the domain discrimination loss so as to weaken the domain discrimination.

Formally, the domain adversarial training objective w.r.t. to $\mathbf{g}$, p and $\mathcal{M}$ can be represented as:

$$\max_{\mathcal{M},p}\min_{\mathbf{g}}\mathcal{L}_{domain}(\hat{\mathcal{S}}, \mathcal{T}; \theta_{\mathcal{M},p,\mathbf{g}}) \tag{7}$$

4.3 Learning Procedure

Joint training objective. Given m source domains $\hat{\mathcal{S}}$ and a target domain $\mathcal{T}$, the sentiment classifier and the domain discriminator are jointly trained for optimizing the PLM $\mathcal{M}$, soft prompt embeddings p, MLM head function f and domain discriminators $\mathbf{g}$, and the final training objective is formally represented as:

$$\begin{aligned}\min_{\mathcal{M},p,f}\big\{&\lambda\mathcal{L}_{class}(\mathcal{S}; \theta_{\mathcal{M},p,f}) \\ &- \min_{\mathbf{g}}\mathcal{L}_{domain}(\hat{\mathcal{S}}, \mathcal{T}; \theta_{\mathcal{M},p,\mathbf{g}})\big\}\end{aligned} \tag{8}$$

where λ is a trade-off parameter. The sentiment classification objective $\mathcal{L}_{class}$ and the domain discrimination objective $\mathcal{L}_{domain}$ are defined in Eq. (4) and Eq. (6), respectively.

Training procedure. The iterative training procedure is summarized in Algorithm 1. In each iteration, the input samples of each source domain are first used for training the PLM $\mathcal{M}$, several learnable vectors p and the MLM head function f. The sentiment classification loss is computed in line 5. Then the samples of each source domain and the target domain are mapped to different domain discriminators to train the PLM $\mathcal{M}$, several learnable vectors p and the domain discriminator g_l. The corresponding domain discrimination loss is computed in line 6. The sentiment classification loss is used for updating the parameters of the PLM, several learnable vectors and the MLM head function (line 7, 10). The domain discrimination loss is used for updating the parameters of the PLM, several learnable vectors and the domain discriminators. Obviously, the parameters of the PLM and several learnable vectors be updated together by the above two losses.

Algorithm 1 Training Process of AdSPT.

Input: Training samples of m source domain dataset $\hat{\mathcal{S}} = \{\mathcal{S}_l\}_{l=1}^m = \{\{\boldsymbol{x}_i^s, y_i^s\}_{i=1}^{N_l^s}\}_{l=1}^m$ and a target domain dataset $\mathcal{T} = \{\boldsymbol{x}_i^t\}_{i=1}^{N^t}$; the number of training iterations n.
Output: Configurations of AdSPT $\theta_{\mathcal{M},p,f,\mathbf{g}}$
Initialize: PLM $\theta_{\mathcal{M}}$; soft prompt embeddings θ_p; MLM head function θ_f; domain discriminator $\{\theta_{g_l}\}_{l=1}^m$; learning rate η; trade-off parameter λ.

1: **while** Training steps not end **do**
2: **for** d in {Source, Target} **do**
3: **if** d = Source **then**
4: **for** l in $\{1, \ldots, m\}$ **do**
5: $\mathcal{L}_{class} \leftarrow \mathcal{L}_{class}(\mathcal{S}_l; \theta_{\mathcal{M},p,f})$
6: $\mathcal{L}_{domain} \leftarrow \mathcal{L}_{domain}(\mathcal{S}_l, \mathcal{T}; \theta_{\mathcal{M},p,g_l})$
 # Minimizing the MLM head classification loss
7: $\theta_f \leftarrow \theta_f - \nabla_{\theta_f}\mathcal{L}_{class}$
 # Minimizing the domain discrimination loss
8: $\theta_{g_l} \leftarrow \theta_{g_l} - \nabla_{\theta_{g_l}}\mathcal{L}_{domain}$
9: **end for**
 # Minimizing the sentiment classification loss
10: $\theta_{\mathcal{M},p} \leftarrow \theta_{\mathcal{M},p} - \nabla_{\theta_{\mathcal{M},p}}(\lambda\mathcal{L}_{class} - \mathcal{L}_{domain})$
11: **end if**
12: **end for**
13: **end while**

5 Experiments

In this section, we conduct experiments to evaluate the effectiveness of our methods. Our experiments are carried out on single-source domain adaptation and multi-source domain adaptation settings (§ 5.3). In addition, we also investigate how different components in the model impact the performance of cross-domain sentiment analysis with different settings.

5.1 Experimental Setup

Dataset. We evaluate on the **Amazon reviews** dataset (Blitzer et al., 2007), which has been widely used for cross-domain sentiment classification. This dataset contains reviews of binary categories from four domains: Books (B), DVDs

S → T	Fine-tuning				Prompt-tuning			
	BERT-DAAT	SENTIX$_{Fix}$	FT	FT + AT	PT(HARD)	PT(HARD) + AT	PT(SOFT)	AdSPT
B → D	89.70	91.30	88.96	89.70	89.75	90.75	90.50	**92.00**
B → E	89.57	93.25	86.15	87.30	91.75	92.45	93.05	**93.75**
B → K	90.75	**96.20**	89.05	89.55	91.90	92.70	92.75	93.10
D → B	90.86	91.15	89.40	89.55	90.90	91.50	91.75	**92.15**
D → E	89.30	93.55	86.55	86.05	91.75	92.75	93.55	**94.00**
D → K	87.53	**96.00**	87.53	87.69	91.05	92.35	92.50	93.25
E → B	88.91	90.40	86.50	87.15	90.00	91.90	91.90	**92.70**
E → D	90.13	91.20	87.98	88.20	92.10	92.55	93.25	**93.15**
E → K	93.18	**96.20**	91.60	91.91	92.90	93.55	93.95	94.75
K → B	87.98	89.55	87.55	87.65	89.15	90.75	91.75	**92.35**
K → D	88.81	89.85	87.30	87.72	90.05	91.00	91.35	**92.55**
K → E	91.72	93.55	90.45	90.25	92.15	92.50	93.10	**93.95**
Avg.	90.12	92.68	88.25	88.56	91.12	92.06	92.45	**93.14**

Table 1: Results of single-source domain adaptation on Amazon reviews. There are four domains, B: Books, D: DVDs, E: Electronics, K: Kitchen appliances. In the table header, S: Source domain; T: Target domain; FT: Fine-tuning; AT: Adversarial training; PT(HARD): Prompt-tuning with the hard prompt; PT(SOFT): Prompt-tuning with the soft prompt; + represents the combination, e.g., "PT(HARD) + AT" represents hard prompt tuning with the domain adversarial training. AdSPT is also called "PT(SOFT) + AT". We report mean performances over 5 fold cross-validation.

(D), Electronics (E) and Kitchen appliances (K). Each domain has totally 2,000 manually labeled reviews. We use different settings for single-source domain adaptation and multi-source domain adaptation. For each domain, there are 2000 labeled reviews, including 1000 positive and 1000 negative, and 4000 unlabeled reviews. Following previous work (Ruder and Plank, 2017), we randomly select a small part (20%) of examples in each domain as the development set to save the best training model and perform a 5 fold cross-validation.

In single-source domain adaptation, we follow previous work (Ziser and Reichart, 2018) to construct 12 cross-domain sentiment analysis tasks (corresponding to 12 ordered domain pairs). In multi-source domain adaptation, we choose three-domain data as multiple source domains and the remaining one as the target domain, e.g., "BDE → K". So there are 4 combinations, corresponding to 4 tasks.

Training details. In the **Amazon reviews** experiments, we adopt a 12-layer Transformer (Vaswani et al., 2017; Devlin et al., 2019) initialized with RoBERTa$_{\text{BASE}}$ (Liu et al., 2019) as the PLM. During the training, we train with batch size of 2 for 10 epoches. The optimizer is Adam with learning rate $2e^{-5}$ for the PLM optimization and $5e^{-5}$ for optimizing domain discriminators. All experiments are conducted with an NVIDIA GeForce RTX 2080 Ti.

5.2 Baselines

We compare our method against 2 state-of-the-art methods, and also design several variants of fine-tuning and prompt tuning as baselines to demonstrate the effectivenss of adversatial training strategy in soft prompt tuning for DA.

(1) **BERT-DAAT**(Du et al., 2020): Use BERT post-training for cross-domain sentiment analysis with adversarial training.

(2) **SENTIX**$_{Fix}$(Zhou et al., 2020): Pre-train a sentiment-aware language model by several pre-training tasks.

(3) **Fine-tuning**: Standard fine-tuning vanilla PLMs in the source domain labeled data, which use the hidden representation of [CLS] for classification.

(4) **Fine-tuning + AT**: Add the adversarial training operating on standard fine-tuning vanilla PLMs.

(5) **Prompt-tuning(Hard)**: Use a manually defined template "It is [MASK]" for prompt-tuning.

(6) **Prompt-tuning(Hard) + AT**: Add the adversarial training operating on Prompt-tuning(Hard).

Following previous work (Du et al., 2020; Zhou et al., 2020), we adopt the accuracy to evaluate the performance.

5.3 Main Results

Main results contain results of single-source domain adaptation (Table 1) and multi-source domain adaptation (Table 2).

S → T	Fine-tuning		Prompt-tuning			
	FT	FT + AT	PT(HARD)	PT(HARD) + AT	PT(SOFT)	AdSPT
BDE → K	89.70	91.30	91.50	92.25	93.25	**93.75**
BDK → E	90.57	91.25	91.30	93.00	93.75	**94.25**
BEK → D	88.56	89.05	90.75	91.25	92.00	**93.50**
DEK → B	89.86	91.75	92.00	92.25	92.75	**93.50**
Avg.	89.67	90.84	91.39	92.00	92.94	**93.75**

Table 2: Results of multi-source domain adaptation on Amazon reviews.

Results of Single-source Domain Adaptation. Table 1 shows our main experimental results under single-source domain adaptation. We can observe that our method AdSPT outperforms all other methods in most of single-source domain adaptation.

Compared with previous state-of-the-art methods, AdSPT is significantly superior to BERT-DAAT and SENTIX$_{Fix}$ on average (3.02 absolute improvement and 0.46 absolute improvement, respectively). More specifically speaking, prompt-tuning methods achieve better results than BERT-DAAT on most of single-source domain adaptation. This indicates that prompt tuning can stimulate pre-encoded knowledge in PLMs to solve the DA problem. But the performance of PT(HARD) and PT(HARD) + AT is lower than that of SENTIX$_{Fix}$ on average (91.12% v.s. 92.68% and 92.06% v.s. 92.68%), showing that the feature representation of the [MASK] token in hard prompt tuning learns more domain knowledge of source domains, which leads to degraded performance on the target domain. Conversely, PT(SOFT) is comparable to SENTIX$_{Fix}$ on average (92.45% v.s. 92.68%) and AdSPT achieves better results than SENTIX$_{Fix}$ on average (0.46 absolute improvement). It shows that soft prompt tuning not only learns domain-aware continuous vectors, but also weakens the domain discrepancy of the feature distribution of the [MASK] position. In addition, prompt-tuning methods are consistently superior to FT and FT + AT, either using a hard prompt, or soft prompt.

In prompt-tuning, soft prompt tuning methods achieve better performances than corresponding hard prompt tuning methods (1.33 absolute improvement and 1.08 absolute improvement, respectively). This indicates these separate soft prompts can flexibly learn in-domain knowledge of different domains, which makes the feature representation of the [MASK] token more suitable for predicting the predefined label words. So soft prompt is more applicable to the DA problem than a hard prompt. When we add a domain adversarial training operation on soft prompt tuning, AdSPT achieves the new start-of-the-art result on average. It shows that the domain adversarial training strategy can enhance the domain-invariant feature of the [MASK] token among different domain datasets.

Results of Multi-source Domain Adaptation. Table 2 shows our main experimental results under multi-source domain adaptation.

Compared with fine-tuning methods, variants of prompt tuning achieve better performances (over at least 0.55 absolute improvement on average). This is mainly because prompt tuning uses the feature representation of [MASK] token for classification, rather than the feature representation of [CLS] token. On the one hand, fine-tuning is difficult to train the domain-specific classifier accurately from scratch on the unlabeled dataset. On the other hand, prompt tuning is used to classify by predicting the feature distribution of the [MASK] token in the set of label words, which can activate some prior knowledge in PLMs.

Compared with hard prompt tuning methods, soft prompt tuning methods achieve significant improvements on average (92.94% v.s. 91.39% and 93.75% v.s. 92.94%). Constructing the sophisticated hard template not only requires expertise knowledge and time, but the unified predefined hard template leads to the domain discrepancy of the feature representation of the [MASK] position that is unsuitable for multi-domain adaptation.

Besides, PT(HARD) + AT achieves a better result than PT(HARD) on average (0.61 absolute improvement), which shows the domain adversarial training can obtain domain-invariant features among different domains by domain discriminators for DA. So when adding the domain adversarial training into soft prompt tuning, AdSPT achieves the best results under multi-source domain adaptation setting. This shows the effectiveness of the collaboration of soft prompt tuning and the domain adversarial training strategy. In the domain ad-

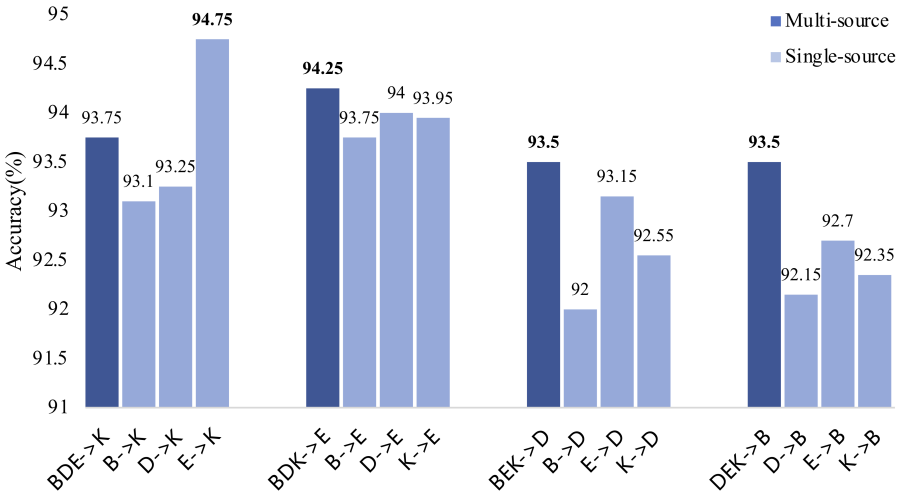

Figure 3: Analysis of multi-source and single-source

versarial training, using the feature representation of the [MASK] token to obtain domain invariance is better for predicting the predefined set of label words.

5.4 Analysis

Multi-source v.s. Single-source. We make more detailed comparisons to explore the effect of multi-source domain adaptation and single-source domain adaptation settings. Figure 3 illustrates the influence of multi-source and single-source on the predicted results of the same target domain. When the target domain is "E", "D", or "B", multi-source achieves better results in the target domain than single-source, showing that in most cases, multi-source domain adaptation is superior to single-source domain adaptation in cross-domain research. However, when the target domain is "K", the result of "E → K" is superior to that of "BDE → K" (94.75% v.s. 93.75%). It is mainly because the feature distribution of "E" and "K" is closer.

Effect of Soft Prompts. As stated in previous works (Gao et al., 2020), the choice of hard templates may have a huge impact on the performance of prompt tuning. In this subsection, we carry out experiments in "BDE → K" and "B → K" respectively to investigate the influence of different soft prompts under multi-source domain adaptation and single-source domain adaptation settings.

As shown in Figure 4, we use 6 different soft prompts (by changing the number of prompt tokens k). The results demonstrate that the choice of templates exerts a considerable influence on the performance of prompt tuning. For soft prompts, surprisingly, prompt tuning yields the best result with the fewest special tokens. Here $k = 3$.

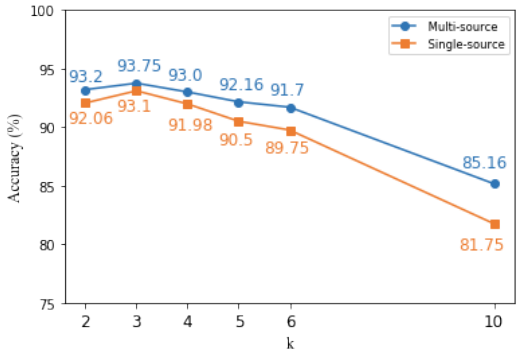

Figure 4: Results of different soft prompts k on "BDE → K" and "B → K"

6 Conclusion

In this paper, we proposed a novel **Ad**versarial **S**oft **P**rompt **T**uning method (AdSPT) for cross-domain sentiment analysis. Firstly, we use domain-specific soft prompts instead of hard templates to represent domain-specific knowledge. The domain-specific soft prompts can alleviate the domain discrepancy w.r.t. the [MASK] representations by MLM task. Meanwhile, we also design a novel adversarial training strategy to learn the domain-invariant knowledge of the [MASK] token among different domains. Experiments on the Amazon reviews dataset achieve state-of-the-art performance.

Acknowledgements

We thank the anonymous reviewers for their helpful comments and suggestions. This work is supported by the Project of Technological Innovation 2030 "New Generation Artificial Intelligence" (Grant no. 2020AAA0107904), the Major Scientific Research Project of the State Language Commission in the 13th Five-Year Plan (Grant nos. WT135-38), and the Key Support Project of NSFC-Liaoning Joint Foundation (Grant no. U1908216).

References

Eyal Ben-David, Nadav Oved, and Roi Reichart. 2021. Pada: A prompt-based autoregressive approach for adaptation to unseen domains.

Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. 2007. Analysis of representations for domain adaptation. *Advances in Neural Information Processing Systems*, 19:137.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. pages 1877–1901.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, and Jianxin Liao. 2020. Adversarial and domain-aware bert for cross-domain sentiment analysis. In *Proceedings of the 58th annual meeting of the Association for Computational Linguistics*, pages 4019–4028.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. pages 3816–3830.

Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2021. Ptr: Prompt tuning with rules for text classification. *arXiv preprint arXiv:2105.11259*.

Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Juanzi Li, and Maosong Sun. 2021. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. *arXiv preprint arXiv:2108.02035*.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. pages 4582–4597.

Zheng Li, Xin Li, Ying Wei, Lidong Bing, Yu Zhang, and Qiang Yang. 2019. Transferable end-to-end aspect-based sentiment analysis with selective adversarial learning. *arXiv preprint arXiv:1910.14192*.

Zheng Li, Ying Wei, Yu Zhang, and Qiang Yang. 2018. Hierarchical attention transfer network for cross-domain sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Zheng Li, Yun Zhang, Ying Wei, Yuxiang Wu, and Qiang Yang. 2017. End-to-end adversarial memory network for cross-domain sentiment classification. In *IJCAI*, pages 2237–2243.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2021a. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021b. Gpt understands, too. *arXiv:2103.10385*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. 2009. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*.

Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359.

Minlong Peng, Qi Zhang, Yu-gang Jiang, and Xuan-Jing Huang. 2018. Cross-domain sentiment classification with target domain specific information. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2505–2513.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Xiaoye Qu, Zhikang Zou, Yu Cheng, Yang Yang, and Pan Zhou. 2019. Adversarial category alignment network for cross-domain sentiment classification. In *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 2496–2508.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Sebastian Ruder and Barbara Plank. 2017. Learning to select data for transfer learning with bayesian optimization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 372–382.

Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3723–3732.

Timo Schick, Helmut Schmid, and Hinrich Schütze. 2020. Automatically identifying words that can serve as labels for few-shot text classification. pages 5569–5578.

Timo Schick and Hinrich Schütze. 2020. Exploiting cloze questions for few shot text classification and natural language inference. pages 255–269.

Timo Schick and Hinrich Schütze. 2021. It's not just size that matters: Small language models are also few-shot learners. pages 2339–2352.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV au2, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou, and Daniel Cer. 2021. Spot: Better frozen model adaptation through soft prompt transfer. *arXiv preprint arXiv:2110.07904*.

Jianfei Yu and Jing Jiang. 2016. Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 236–246.

Han Zhao, Shanghang Zhang, Guanhang Wu, José MF Moura, Joao P Costeira, and Geoffrey J Gordon. 2018. Adversarial multiple source domain adaptation. *Advances in Neural Information Processing Systems*, 31:8559–8570.

Jie Zhou, Junfeng Tian, Rui Wang, Yuanbin Wu, Wenming Xiao, and Liang He. 2020. Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 568–579.

Xiaojin Zhu and Andrew B Goldberg. 2009. Introduction to semi-supervised learning. *Synthesis lectures on artificial intelligence and machine learning*, 3(1):1–130.

Yftah Ziser and Roi Reichart. 2018. Pivot based language modeling for improved neural domain adaptation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1241–1251.