TPTCD: A Prompt Tuning based Two-Step Framework for Cross-Domain Text Classification

Anonymous ACL submission

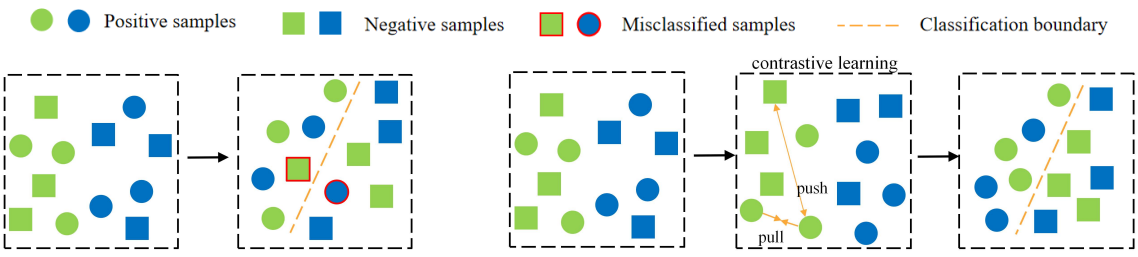
Abstract

In recent years, with the rapid development of large models, prompt tuning has shown strong performance in cross-domain text classification. Nevertheless, it still faces the following two issues: (i) prompt tuning based methods usually do not align the same class of labels in different domains; and (ii) they probably focus on some simple samples in the source domain, which may hinder the improvement of model's adaptability. To alleviate these issues, we propose a new method, called Two-step Prompt Tuning with supervised Contrastive learning and feature Disentanglement (TPTCD). Specifically, to enable the model to align the same labels in different domains , we innovatively combine soft prompt tuning with supervised contrastive learning, leveraging their merits. Furthermore, to improve the domain adaptation ability, we propose a novel adversarial enhanced Variational Auto Encoder (VAE) for feature disentanglement, which enables the model to learn more effective features in both source and target domains. Extensive experiments based on benchmark datasets demonstrate the competitiveness of our proposed method, compared against state-of-the-art cross-domain text classification models.

1 Introduction

With the continuous development of natural language processing (NLP) technology (Zhang et al., 2022), artificial intelligence applied to text has become more and more mature (Li et al.; Junfan et al., 2024). In recent years, some efficient large pre-trained language models have emerged, such as GPT (Radford et al., 2018), XLNET (Yang et al., 2019), BERT (Kenton and Toutanova, 2019), and RoBerta (Liu and Stoyanov, 2019). Text classification, as an important subtask of web content mining and NLP, plays an indispensable role in various aspects of our real life, such as rumor detection (Malhotra and Vishwakarma, 2020), online recommendation (Liu et al., 2019), sentiment analysis (Neogi et al., 2021), spam email detection (Mansoor et al., 2021), and so on. Aforementioned models have achieved excellent performance on these downstream tasks. Nevertheless, they usually rely on manually annotated data, and the distribution and availability of labeled data are inconsistent and vary greatly among different domains (Wu and Guo, 2020; Blitzer et al., 2007). To alleviate this challenge, unsupervised domain adaptation (UDA) was proposed (Ganin and Lempitsky, 2015). A lot of models have been developed based on UDA, including domain adversarial training methods (Du et al., 2020; Zou et al., 2021), pivot-based methods (Ziser and Reichart, 2018; Ben-David et al., 2020), semantic representation based methods (Li et al., 2022), statistical measurement minimization based methods (Guo et al., 2020), etc. To extract domain-invariant features (a.k.a., common features), these methods mainly minimize the difference between source domain and target domain in the feature space.

With the rapid development of large models in recent years, prompt tuning based pre-trained models (Brown et al., 2020) have achieved excellent results in the field of NLP. For example, Wu (Wu and Shi, 2022) propose AdSPT, which shows that soft prompt tuning performs well. Nevertheless, these prompt tuning based methods still face two severe challenges/issues in cross-domain text classification. First, common methods often did not consider the consistency between samples from the same class in different domains. In other words, they did not fully pull the samples with the same class closer together and push samples with different classes further apart, recurring misclassified samples, as shown in Figure 1a. Second, the models may perform simple and shortcut learning in the source domain (i.e., relying on some simple features in the source domain), while they ignore the domain-specific features crucial for domain adapta-



(a) common model (b) model with SCL

Figure 1: Common model and model with supervised contrastive learning (SCL); two colors denote two different domains.

tion, which will hinder the improvement of model's adaptation ability.

To address the above issues, this paper presents a novel cross-domain text classification method, called Two-step Prompt Tuning with supervised Contrastive learning and feature Disentanglement (TPTCD). In the first step, we adopt supervised contrastive learning (SCL) together with soft prompt tuning to improve the model's class aware ability. That is, it can pull the samples with the same class closer together and push samples with different classes further apart, as shown in Figure 1b. Particularly, we use a memory bank to improve the effectiveness of SCL, and use domain adversarial training to learn domain-invariant features. In the second step, we propose a novel adversarial training method to further promote feature disentanglement. This training method is based on an enhanced variational auto encoder (VAE), which can learn more effective features. This way, the combination of these two steps fully learns class-aware features and domain-invariant features, producing impressive performance. To summarize, our contributions are as follows:

- We innovatively introduce supervised contrastive learning to improve the existing soft prompt tuning model, which can enhance the model's class-aware ability. To the best of our knowledge, this is the first cross-domain text classification work that combines the merits of supervised constrastive learning and prompt tuning.
- We propose a novel adversarial training strategy based on VAE, which can better promote feature disentanglement, learning more valuable features.
- We conduct extensive experiments based on benchmark datasets, and the experimental results demonstrate that our method achieves an extremely impressive performance, compared against state-of-the-art methods.

2 RELATED WORK

Prompt Tuning. As shown in (Brown et al., 2020), prompt tuning achieves better results than fine-tuning on many downstream tasks. Some of initial studies explore hard templates for text classification and natural language inference (Schick and Schütze, 2021). However, for cross-domain text classification, weak domain knowledge may hinder the selection of suitable templates. Therefore, some researchers study how to generate hard template automatically based on different domains (Ben-David et al., 2022). Later, researchers realize that it is unnecessary to limit templates to human-interpretable natural language, which may affect the performance of the model (Qin and Eisner, 2021). With this in mind, researchers attempt to address the limitations of hard templates. For example, soft prompts (Vu et al., 2022; Gu et al., 2022), P-tuning V2 (Liu et al., 2022) and prefix tuning (Li and Liang, 2021) use several learnable vectors as prompt words, instead of specific words. Our paper is to enhance soft prompts tuning, so as to improve the overall performance for cross-domain text classification.

Contrastive Learning. Contrastive learning (CL) has achieved good results in the field of deep learning (Chen et al., 2020; Azuma et al., 2023; Junfan et al., 2024). On the basis of contrastive learning, the supervised contrastive learning (SCL) uses label information to optimize the representation learning ability of the model (Khosla et al., 2020; Luo et al., 2022). SCL considers samples with the same label as positive pairs and samples with different labels as negative pairs. In this paper, we combine SCL and prompt tuning innovatively

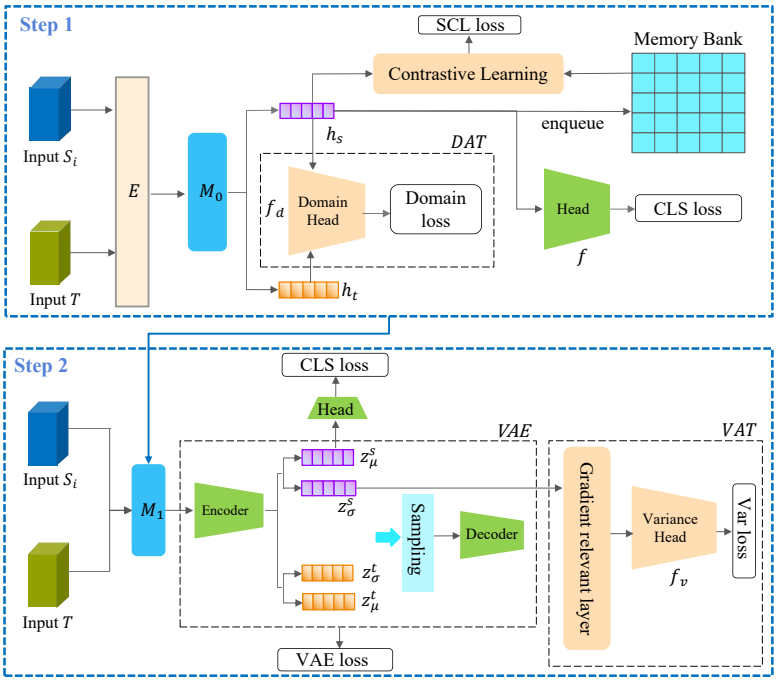


Figure 2: Framework of our model. Here E means the embedding layer; M_0 and M_1 mean feature extractors; SCL means supervised contrastive learning; DAT means domain adversarial training; and VAT means variance adversarial training.

to improve the overall classification ability of the model.

Feature Disentanglement. Feature disentanglement refers to the process of representing different features separately in data representation. Feature disentanglement has been used in many fields. For example, some researchers used it to separate the color and orientation of images (Chen et al., 2016), to separate the content and style of text (John et al., 2019; Pergola et al., 2021). Also, some researchers used features disentanglement in text classification for higher accuracy (Huang et al., 2021; Song et al., 2024). Compared to these works, our paper proposes a novel adversarial training variational auto encoder to promote feature disentanglement, which efficiently separates invalid features and valid ones for accurate classification in training samples.

3 PROBLEM FORMULATION

Assuming we have labeled data from K source domains $\{D_i^s\}_{i=1}^K$. For each source domain, the labeled data is represented as $S_i = \{x_j^i, y_j^i\}_{j=1}^{N_i}$, where $x_j^i = [w_1^i, w_2^i, ..., w_m^i]$ is an input sample with m words in the i-th domain, y_j^i indicates the sentiment label of the sample, and N_i^s is the number of samples in this domain. On the other hand, we use D^t and $T = \{x_j^t\}_{j=1}^{N^t}$ to represent the target domain and the unlabel dataset respectively, where $x_j^t = [w_1^t, w_2^t, ..., w_n^t]$ represents an unlabeled sample with n words in the target domain, N^t is the number of samples in the target domain. Specifically, the input text x_j^i (or x_j^t) is initially processed by the feature extractor M_0, which will produce a representation h. Subsequently, based on the feature representation h obtained before, the classification head $f(\cdot)$ is used for text classification.

4 METHOD

4.1 Overview of Our Solution

Figure 2 presents the framework of our proposed method. It can be seen that the framework consists of two parts. In the first step, we adopt PLM-based soft prompt tuning for feature extraction, which can generate feature representations for samples in the source domain and unlabeled samples in the target domain. Then, domain adversarial training is performed on all sample representations to extract domain-invariant features . Meanwhile, we perform supervised contrastive learning on source domain samples to enhance the model's class-aware ability. In addition, a memory bank is used to improve the effectiveness of contrastive learning.

In the second step, to generate efficient feature representations, we adopt the model M_1 as the

feature extractor, where M_1 denotes the best performance model trained in the first step (see the arrow pointing to M_1). Then, we use variational auto encoder for feature disentanglement, which separates valuable and valueless features (see the dashed box titled VAE). Meanwhile, we propose a novel adversarial training method on the valueless features for facilitating deeper feature disentanglement and further separating more valuable features (see the dashed box titled VAT). In what follows, we present the details (Sections 4.2 and 4.3).

4.2 Soft Prompt Tuning with SCL

Soft prompt tuning transforms the classification task into a fill-in-the-blank task by adding non-fixed prompt words to the samples (Brown et al., 2020). A recent study (Wu and Shi, 2022) demonstrated the effectiveness of soft prompt tuning in cross-domain text classification. Nevertheless, prompt tuning based methods still face challenges (recall Section 1). To alleviate the challenges, we attempt to combine soft prompt tuning with supervised contrastive learning (SCL), which can allow us to enhance the model's class-aware ability. The rationale behind our idea is to pull the samples with the same class closer together and push samples with different classes further apart.

Formally, the input samples of both domains are denoted as $x = [w_1, w_2, ..., w_n]$, and the resulting classification label is y which belongs to a binary label space $Y = \{positive, negative\}$. For each x above, we need to add soft prompt words at the end, and the resulting prompt template function is below:

$$x_p = [E(w_1), ..., E(w_n), P_1, ..., P_l, E([mask])] \tag{1}$$

where $E(\cdot)$ is the embedding layer of the pre-trained language model M_0, P_i ($i \in [1, l]$) is the soft prompt token embedding, l denotes the number of prompt tokens. The above template transforms the input x into an actual prompt input x_p with prompt words and a $[mask]$ token. Next, we input x_p into M_0 to get the hidden representation $h \in R^d$, where d is the hidden layer dimension of M_0:

$$h = M_0(x_p) \tag{2}$$

Later, the classification head $f(\cdot)$ will generate the probability distribution q of all words of the $[mask]$ token based on h:

$$q = f(h) \tag{3}$$

Given a selection function $S(\cdot)$, the function selects the probability q' corresponding to the two label words from q:

$$q'(y) = S(q) \tag{4}$$

where the label words can be set to $V \in \{good, bad\}$, referred to (Wu and Shi, 2022). Last, the final prediction probability p is obtained via softmax normalization:

$$p = \frac{\exp(q'(y))}{\sum_{y \in Y} \exp(q'(y))} \tag{5}$$

For cross-domain text classification, given a source domain dataset $S_i = \{x_j^i, y_j^i\}_{j=1}^{N_i}$, the binary cross entropy loss is used to optimize the feature extractor M_0 and the classification head $f(\cdot)$:

$$L_{cls}(S_i; M_0, f) = -\sum_{j=1}^{N_i} \sum_{k=1}^{|Y|} y_j^k \log p_j^k \tag{6}$$

We adopt domain adversarial training (Ganin et al., 2016) to extract domain-invariant features, which can transfer general knowledge from the source domain to the target domain. Formally, given K source domains, we use a binary classifier to distinguish data from different domains, denoted as f_d. And the domain label set is represented as $D = \{0, 1\}$. Intuitively, domain discriminator f_d should be optimized to maximumly discriminate instances from different domains, while the shared feature extractor M_0 should be optimized to maximumly fool the discriminator. Specifically, the adversarial learning loss is below:

$$L_{adv}^d(S_i, T; M_0, f_d) = -\sum_{j=1}^{N_i+N^t} \sum_{k=1}^{|D|} y_d \log f_d(h_j^k) \tag{7}$$

where N_i and N^t is the number of samples in the source domain S_i and target domain T respectively, y_d is the domain label corresponding to the sample, and h_j is the hidden representation of the sample obtained by the feature extractor. As shown above, domain adversarial training is a minimax game, which can be expressed as:

$$\max_{M_0} \min_{f_d} \; L_{adv}^d(S_i, T; M_0, f_d) \tag{8}$$

To improve the model's class-aware ability, we introduce supervised contrastive learning (SCL) (He et al., 2020) into our framework. SCL

can help us pull representations of the same class together and exclude representations of different classes. This not only improves the class-awareness of the model, but also enables the model to learn a more uniform distribution of each class, thereby enhancing the discriminative power of the hidden representation. Typically, SCL uses samples with the same label in the same batch as positive examples and samples with different labels as negative examples.

In addition, since contrastive learning is extremely sensitive to the number of contrast samples, the learning effect is not good when the number is small. To obtain better results, we use a memory bank Q to increase the number of contrastive samples in each batch during training. The memory bank Q stores the sentence representations h and their corresponding labels y (for the computation in each batch). Particularly, the number of contrastive samples is the sum of samples in current batch and samples in the memory bank.

Since SCL requires samples to be labeled, we only perform it on source domain samples. The contrast loss can be formulated as:

$$L_{scl} = -\frac{1}{N_b} \sum_{h_i \in B} \sum_{h_p \in P(i)} \log \frac{\exp(h_i \cdot h_p / t)}{\sum_{h_n \in N(i)} exp(h_i \cdot h_n / t)} \tag{9}$$

where B refers to a batch of samples with size N_b, h_i is the sample of the current batch, h_p and h_n are the positive samples and negative samples belonging to the positive example set $P(i)$ and the negative example set $N(i)$, respectively. Finlay, we get the joint loss function L_1 of the first step, in which three losses mentioned before are used to optimize the model together:

$$L_1 = L_{cls} + L_{adv}^{d} + L_{scl} \tag{10}$$

4.3 Adversarial Enhanced VAE

To avoid shortcut learning and learn more effective features, we adopt the best checkpoint of the model in the first step as the feature extractor, and employ the variational auto encoder (VAE) (Kingma and DP, 2014) for feature disentanglement, which can allow us to differentiate effective and ineffective features from the potential feature space. Furthermore, a novel variance adversarial training (VAT) is used to enhance VAE-based feature disentanglement, so as to further improve the model's performance.

For the hidden feature representation h (obtained by the feature extractor) and a probabilistic latent variable z, two steps are performed in VAE: (1) encoding h using z, and (2) decoding h from z, where z follows a Gaussian distribution $N(0, 1)$. For each z, there are two functions μ and σ, which respectively determine the mean and variance of the Gaussian distribution corresponding to z. Then the accumulation of all Gaussian distributions in the integral domain becomes the original distribution $P(h)$, namely using infinite Gaussian to approximate the true distribution:

$$P(h) = \int P(h|z)P(z)\,\mathrm{d}z \tag{11}$$

where $P(z)$ is the prior distribution that follows $N(0, 1)$. It is difficult to compute $P(h)$, but we can apply variational inference to estimate this value. Intuitively, we want to approximate the posterior distribution $P(z|h)$ with a simpler distribution $Q(z|h)$. If we can determine the parameters of $Q(z|h)$ and ensure that it is very similar to $P(z|h)$, then we can sometimes use $Q(z|h)$ for approximate reasoning. As we know, KL divergence is a measure of the difference between two probability distributions. Therefore, if we want to ensure that $Q(z|h)$ is similar to $P(z|h)$, we can minimize the KL divergence between the two distributions, so the VAE loss according to the evidence lower bound (ELBO) is formulated as follows:

$$L_{vae} = -E_{Q(z|h)} \log P(h|z) + KL(Q(z|h)||P(z)) \tag{12}$$

where the first term represents the likelihood we reconstruct, and the second term ensures the similarity between the distribution Q (we learn) and the true prior distribution P. Here, $Q(z|h)$ can be expressed as $N(\mu, \mathrm{diag}(\sigma^2))$, where μ is the mean and σ^2 is the variance. According to (Fotopoulos, 2006; Song et al., 2024), μ and σ^2 are independent of each other and can be modeled by two independent linear transformations, so we use them to represent effective features and ineffective features, respectively. As shown in Figure 2, in order to make the encoder learn the knowledge in the target domain, we apply VAE loss on each unlabeled sample in the target domain. Formally, μ and σ^2 can be represented as z_μ and z_σ. Further, they can be represented in more detail as z_μ^s and z_σ^s in the source domain, z_μ^t and z_σ^t in the target domain, respectively. Here z_μ is the final feature representation used for classification.

To further improve the performance of the model, we propose a novel adversarial method to enhance VAE, which allows us to extract more effective

classification features. For clarity, we call it the variance adversarial training (VAT). We know that z_σ (i.e., σ^2) is ineffective feature (recall the previous paragraph), usually its classification effect is poor. So there is an intuition that, the worse the classification effect of z_σ, the better the classification effect of z_μ (due to the existence of VAE loss). This case just means that the feature disentanglement is more thorough.

In order to achieve the above purpose, we first pass z_σ via the gradient reversal layer and then input it into a binary classifier f_v, as shown in Figure 2. The loss is as follows:

$$L^v_{adv}(S_i; M_1, V, f_v) = -\sum_{1}^{N_i}\sum_{1}^{|Y|} y \log f_v(z^s_\sigma) \tag{13}$$

where the notations (e.g., N_i) denote the same meanings presented earlier (e.g., N_i is the number of samples in the source domain). The adversarial training can be expressed as:

$$\max_{M_1,V} \min_{f_v} \; L^v_{adv}(S_i; M_1, V, f_v) \tag{14}$$

where S_i is the source domain dataset, M_1 is the feature extractor, V is the VAE layer (encoder and decoder), f_v is the binary classifier. Under VAT, if the classification effect of f_v is better, then the actual classification effect of z^s_σ obtained by M_1 and V will be worse.

For z^s_μ, we also use the classification head $f(\cdot)$ to predict it. The classification loss is the same as that in the first step. That is:

$$L_{cls}(S_i; M_1, f, V) = -\sum^{N_i}\sum^{|Y|} y \log f(z^s_\mu) \tag{15}$$

Finally, we get the joint loss function of the second step:

$$L_2 = L_{cls} + L_{vae} + L^v_{adv} \tag{16}$$

where these losses further optimize the model trained by the first step, and finally achieve an effective cross-domain text classification model.

5 EXPERIMENTS

5.1 Experimental Settings

Datasets. We adopt two public datasets to conduct experiments: the Amazon Reviews Dataset (Blitzer et al., 2007) and the FDU-MTL Dataset (Liu et al., 2017). The Amazon dataset is a classic public dataset which contains four domains: Books (B), DVDs (D), Electronics (E), and Kitchen (K). The FDU-MTL dataset consists of 16 domains, the first 14 domains are Amazon reviews of different products, and the remaining two domains are movie reviews from IMDB and MR datasets.

Implementation Details. In our experiments, the feature extractor used in the first step is the official pre-trained RoBERTa-base (Liu and Stoyanov, 2019) provided by Huggingface. The hidden layer size of Roberta is 768, and the max sequence length is 512. We set the batch size and memory bank size to 8 and 128 respectively. Adam (Kingma and Ba, 2015) is used as the optimizer, the learning rate of the optimization of the classification head $f(\cdot)$ is 1e-5; the learning rate of the optimization of the domain classifier f_d (or variance classifier f_v) is 5e-5. We train the model for 10 epochs at each step. Following previous work (Wu and Shi, 2022), the accuracy will be adopted as the evaluation criterion in this paper. All of our experiments are performed with Pytorch and HuggingFace Transformers on an NVIDIA GeForce RTX 4090 24 GB GPU.

5.2 Baselines

We compare our proposed model with state-of-the-art cross-domain classification methods, which are as follows: **DACL**, **DAAT**, **SENTIX**, **EADA**, **DASK**, **COBE**, **UDALM**, **AdSPT**, **RCA**, **TACIT**, **PL-Mix**.

5.3 Results on Amazon Dataset

In the single-source domain setting, the experimental results on the Amazon dataset are shown in Table 1. Overall, our model achieves state-of-the-art performance with an average accuracy of 94.18% across 12 cross-domain tasks. Also, we can see that our method achieves the best results on 8 (out of 12) single-source domain tasks, compared with other competitive methods. On the "E→K" and "K→B" tasks, although our method do not reach the highest accuracy, it is still very close to the highest value (0.01% lower). The average accuracy value of our model is 1.04% higher than the AdSPT (93.14%) and 0.52% higher than PL-Mix (93.66%). These results reveal that, in cross-domain text classification, pulling representations of different domains with the same labels together and using adversarial enhanced VAE can help soft prompt tuning to achieve a more excellent performance. In addition, we can see that the accuracy of the "E→D" and

Table 1: Accuracy of single-domain adaptation on Amazon Dataset.

S → T	DAAT	EADA	DASK	SENTIX	COBE	UDALM	AdSPT	TACIT	PL-Mix	Ours
B → D	89.70	88.10	90.90	91.15	90.05	92.18	92.47	92.65	93.38	**93.87**
B → E	89.57	85.25	92.30	92.50	90.45	93.55	93.79	93.81	93.84	**94.36**
B → K	90.75	81.20	92.75	95.70	92.90	95.32	94.96	95.03	95.32	**95.82**
D → B	90.86	86.05	91.85	90.85	90.98	93.34	93.21	93.57	94.56	**94.74**
D → E	89.30	85.35	92.40	92.15	90.67	93.60	93.84	93.16	94.12	**94.52**
D → K	87.53	80.35	92.35	94.95	92.00	93.21	95.16	94.40	95.53	**95.90**
E → B	88.91	89.35	90.00	88.10	87.90	91.80	91.32	92.70	92.31	**92.92**
E → D	90.13	87.15	89.20	89.86	87.87	**93.38**	90.58	92.06	90.93	92.00
E → K	93.18	85.11	94.65	95.45	93.33	94.85	95.01	**95.87**	95.36	95.86
K → B	87.98	89.65	89.75	87.00	88.38	92.74	91.84	93.06	**93.18**	93.17
K → D	88.81	89.20	89.45	88.05	87.43	92.33	90.76	**91.97**	90.68	91.88
K → E	91.72	90.50	93.35	91.85	92.58	93.56	94.24	94.57	94.69	**95.09**
Avg.	89.87	86.44	91.58	91.47	90.38	93.32	93.10	93.57	93.66	**94.18**

"K→D" tasks are relatively low, which may be due to the small similarity between these domains (e.g., "K" and "D").

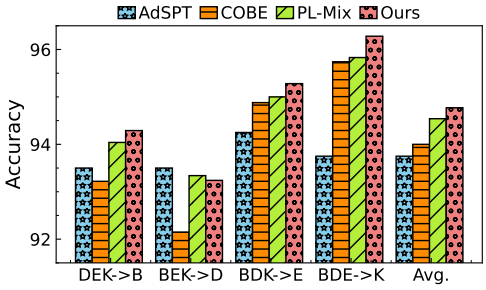


Figure 3: Accuracy of multi-source domain adaptation on Amazon dataset.

We also compare our method to other ones in multi-source domain setting, treating one domain as the target domain and the other three domains as one large source domain. Experimental results of multi-source domain adaptation on Amazon datasets are shown in Figure 3. It can be seen that our method achieves the best results on three tasks, only slightly lower than the other two methods on "BEK→D" task. Moreover, our method still achieves the best overall results with the average accuracy of 94.77% compared to the other three competitors, which further verifies the effectiveness of our proposed method. As for our results, we find the accuracy has increased by 0.59% compared to our single source domain experimental results. This is consistent with the common phenomenon. The underlying reason could be that, when the tasks are similar, adding the number of source domains will improve the performance of the target domain, due to the existence of commonalities among these domains.

5.4 Results on FDU-MTL Dataset

We conduct "15→1" multi-source domain experiment on FDU-MTL dataset, totaling 16 tasks. The experimental results are shown in Table 2. We can see that our method achieves state-of-the-art performance with an accuracy of 93.81% on average, eclipsing the performance of these baselines. Impressively, our model achieves the best results on 14 (out of 16) tasks, only slightly lower than RCA on the domain of camera and magazines. Our average accuracy is 4.71% higher than DACL, 3.78% higher than COBE, 1.61% higher of AdSPT, and 3.61% higher than that of RCA. These results again validate the competitiveness of our proposed method.

In addition, we find that there is a significant gap in the result of MR compared to other domains, which is 6.41% lower than the average accuracy. This could be due to the excessively long sentences and the smaller feature similarity between MR and other domains. Meanwhile, we find that the average accuracy on FUD-MTL dataset (i.e., 93.81%) is 0.96% lower than that of multi-source domain tasks on Amazon dataset (i.e., 94.77%). This could be due to that the addition of MR and IMDB has led to a decrease in overall similarity among these domains. This phenomenon may imply that the similarity among domains could be a more important factor which can affect model performance, compared to the number of source domains.

Table 2: Accuracy of multi-source domain adaptation on FDU-MTL.

Target	DACL	COBE	AdSPT	RCA	Ours
books	87.50	90.67	94.23	89.20	**96.97**
dvd	90.67	87.50	90.58	90.40	**91.64**
elec.	90.30	92.33	93.08	87.60	**95.14**
kit.	91.50	90.75	94.28	92.50	**95.81**
appa.	89.50	91.16	92.31	88.60	**94.66**
baby	92.00	93.17	94.95	92.70	**95.33**
camera	91.50	91.67	92.07	**95.80**	93.70
health	90.50	94.33	95.14	90.70	**96.92**
IMDB	87.30	86.58	91.35	89.50	**92.36**
maga	93.80	90.50	91.78	**95.40**	93.99
MR	76.00	78.91	85.77	78.30	**87.40**
music	86.30	89.17	91.11	87.70	**92.93**
softw.	90.50	90.82	93.61	92.90	**95.19**
sports	89.30	92.15	93.03	89.70	**93.65**
toys	91.30	92.33	92.84	92.40	**94.52**
video	88.50	88.50	89.04	90.00	**90.72**
Avg.	89.10	90.03	92.20	90.20	**93.81**

Table 3: Accuracy of ablation experiment on Amazon dataset.

S→T	-SCL	-VAE	AdSPT	PLMix	Ours
B→D	92.73	93.67	92.47	93.38	**93.87**
B→E	94.04	**94.58**	93.79	93.84	94.36
B→K	95.04	95.57	94.96	95.32	**95.82**
D→B	94.19	94.54	93.21	94.56	**94.74**
D→E	94.37	**94.54**	93.84	94.12	94.52
D→K	95.36	95.66	95.16	95.53	**95.90**
E→B	92.54	92.68	91.32	92.31	**92.92**
E→D	90.88	91.55	90.58	90.93	**92.00**
E→K	95.45	95.73	95.01	95.36	**95.86**
K→B	92.26	92.67	91.84	**93.18**	93.17
K→D	90.85	91.43	90.76	90.68	**91.88**
K→E	94.53	94.90	94.24	94.69	**95.09**
Avg.	93.52	93.96	93.10	93.66	**94.18**

5.5 Ablation Study

To verify the contributions of each module to the overall performance of our method, we conduct corresponding ablation experiments on Amazon dataset. Experimental results of ablation study are shown in Table 3, where -SCL represents not using supervised contrastive learning in the first step, and -VAE represents removing the entire second step. From this table, it can be seen that the performance of our model has decreased in both cases, with average accuracy of 93.52% (-0.66%) and 93.96% (-0.22%) respectively. Meanwhile, in almost all tasks, the accuracy of our method decreases after removing a certain module. These indicate that each module has made a positive contribution to our model for cross-domain text classification.

5.6 Visualization

To analyze the impact of our method on data distribution alignment, we adopt t-SNE to map the high-dimensional feature representations (trained by the model) to low dimensions for feature visualization, and results are shown in Figure 4. From the figure, we can see that before training, both negative and positive features are mixed together and arranged in a scattered manner, with feature points spread throughout the entire space. The feature representation trained by our method is significantly more compact, with clear classification boundary, indicating that our method has transferred source domain knowledge to the target domain, and can align features of the same class well.

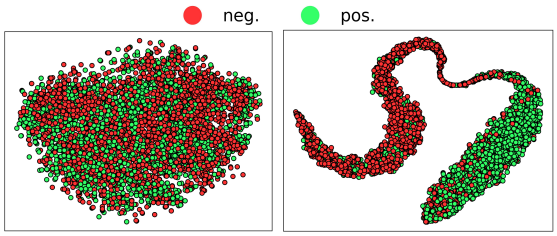


Figure 4: Feature visualization conducted on the Amazon dataset. The visualization is shown under "D→K" task, with untrained features and trained features

6 CONCLUSION

In this paper, we proposed a two-step framework for cross-domain text classification. In the first step, we innovatively introduced supervised contrastive learning to improve the existing soft prompt tuning model, which can enhance the model's class-aware ability. In the second step, we proposed a novel adversarial enhanced VAE to achieve deep feature disentanglement, which enables the model to learn more valuable features. We compared our method with a set of state-of-the-art cross-domain text classification models, based on two public datasets (the Amazon dataset and the FDU-MTL dataset). The experimental results consistently revealed that our method achieved a better performance compared against the competitors in both single-domain setting and multi-domain setting.

Limitations

At present, large models have become a popular research tool, but our method lacks application to large models. Using large models may bring better results. In addition, we have not yet considered the issue of class imbalance in the dataset, which may affect the performance of the model. These two points will also be the focus of our future research.

References

Chiori Azuma, Tomoyoshi Ito, and Tomoyoshi Shimobaba. 2023. Adversarial domain adaptation using contrastive learning. *Engineering Applications of Artificial Intelligence*, 123:106394.

Eyal Ben-David, Nadav Oved, and Roi Reichart. 2022. Pada: Example-based prompt learning for on-the-fly adaptation to unseen domains. *Transactions of the ACL*, 10:414–433.

Eyal Ben-David, Carmel Rabinovitz, and Roi Reichart. 2020. Perl: Pivot-based domain adaptation for pre-trained deep contextualized embedding models. *Transactions of the ACL*, 8:504–521.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of ACL*, pages 440–447.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of NIPS*, page 1877–1901.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of NIPS*, pages 2172–2180.

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 ACL*, pages 4019–4028.

S. B. Fotopoulos. 2006. All of nonparametric statistics. *Technometrics*.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of ICML*, pages 1180–1189.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2022. Ppt: Pre-trained prompt tuning for few-shot learning. In *Proceedings of ACL*, pages 8410–8423.

Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2020. Multi-source domain adaptation for text classification via distancenet-bandits. In *Proceedings of AAAI*, pages 7830–7838.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of CVPR*, pages 9729–9738.

Yufan Huang, Yanzhe Zhang, Jiaao Chen, Xuezhi Wang, and Diyi Yang. 2021. Continual learning for text classification with information disentanglement based regularization. In *Proceedings of NAACL-HLT*, pages 2736–2746.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2019. Disentangled representation learning for non-parallel text style transfer. In *Proceedings of ACL*, pages 424–434.

Chen Junfan, Richong Zhang, Yaowei Zheng, Qianben Chen, Chunming Hu, and Yongyi Mao. 2024. Dualcl: Principled supervised contrastive learning as mutual information maximization for text classification. In *Proceedings of WWW*, pages 4362–4371.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673.

Diederik P Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic gradient descent. In *Proceedings of ICLR*.

Diederik P Kingma and Welling DP. 2014. Auto-encoding variational bayes. In *Proceedings of ICLR*.

Miaomiao Li, Jiaqi Zhu, Yang Wang, Yi Yang, Yilin Li, and Hongan Wang. Ruleprompt: Weakly supervised text classification with prompting plms and self-iterative logical rules. In *Proceedings of WWW*, pages 4272–4282.

Shichen Li, Zhongqing Wang, Xiaotong Jiang, and Guodong Zhou. 2022. Cross-domain sentiment classification using semantic representation. In *Findings of the ACL*, pages 289–299.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of ACL-IJCNLP*, pages 4582–4597.

M.; Goyal N.; Du J.; Joshi M.; Chen D.; Levy O.; Lewis MZettlemoyer L.; Liu, Y.; Ott and V. Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Pengfei Liu, Xipeng Qiu, and Xuan-Jing Huang. 2017. Adversarial multi-task learning for text classification. In *Proceedings of ACL*, pages 1–10.

Wei Liu, Zhi-Jie Wang, Bin Yao, and Jian Yin. 2019. Geo-alm: POI recommendation by fusing geographical information and adversarial learning mechanism. In *Proceedings of IJCAI*, pages 1807–1813.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of ACL*.

Yun Luo, Fang Guo, Zihan Liu, and Yue Zhang. 2022. Mere contrastive learning for cross-domain sentiment analysis. In *Proceedings of ICCL*, pages 7099–7111.

Bhaye Malhotra and Dinesh Kumar Vishwakarma. 2020. Classification of propagation path and tweets for rumor detection using graphical convolutional networks and transformer based encodings. In *Proceedings of IEEE international conference on multimedia big data*, pages 183–190.

RAZA Mansoor, Nathali Dilshani Jayasinghe, and Muhana Magboul Ali Muslam. 2021. A comprehensive review on email spam classification using machine learning algorithms. In *Proceedings of ICOIN*, pages 327–332.

Ashwin Sanjay Neogi, Kirti Anilkumar Garg, Ram Krishn Mishra, and Yogesh K Dwivedi. 2021. Sentiment analysis and classification of indian farmers' protest using twitter data. *International Journal of Information Management Data Insights*, 1(2):100019.

Gabriele Pergola, Lin Gui, and Yulan He. 2021. A disentangled adversarial neural topic model for separating opinions from plots in user reviews. In *Proceedings of NAACL-HLT*, pages 2870–2883.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying lms with mixtures of soft prompts. In *Proceedings of NAACL-HLT*, pages 5203–5212.

Alec Radford, Karthik Narasimhan, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *Technical report, OpenAI blog*.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of ECACL*, pages 255–269.

Rui Song, Fausto Giunchiglia, Yingji Li, Mingjie Tian, and Hao Xu. 2024. Tacit: A target-agnostic feature disentanglement framework for cross-domain text classification. In *Proceedings of AAAI*, pages 18999–19007.

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 ACL*, pages 5039–5059.

Hui Wu and Xiaodong Shi. 2022. Adversarial soft prompt tuning for cross-domain sentiment analysis. In *Proceedings of ACL*, pages 2438–2447.

Yuan Wu and Yuhong Guo. 2020. Dual adversarial co-learning for multi-domain text classification. In *Proceedings of AAAI*, pages 6438–6445.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Proceedings of NIPS*, pages 5754–5764.

Yu Zhang, Zhihong Shen, Chieh-Han Wu, Boya Xie, Junheng Hao, Ye-Yi Wang, Kuansan Wang, and Jiawei Han. 2022. Metadata-induced contrastive learning for zero-shot multi-label text classification. In *Proceedings of WWW*, pages 3162–3173.

Yftah Ziser and Roi Reichart. 2018. Pivot based language modeling for improved neural domain adaptation. In *Proceedings of NAACL-HLT*, pages 1241–1251.

Han Zou, Jianfei Yang, and Xiaojian Wu. 2021. Unsupervised energy-based adversarial domain adaptation for cross-domain text classification. In *Findings of the ACL-IJCNLP*, pages 1208–1218.