Learning Beyond Domains: Misleading Prompts and Pseudo-Label Contrast for Text Domain Generalization

Qizhi Li[1], **Xuyang Wang**[1], **Yingke Chen**[3], **Ming Yan**[4, 5], **Dezhong Peng**[1, 2], **Xi Peng**[1, 2], **Xu Wang**[1, 5*]

[1]The College of Computer Science, Sichuan University, Chengdu, China
[2]Tianfu Jincheng Laboratory, Chengdu, China
[3]The Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne, United Kingdom
[4]School of Software, Xinjiang University, Urumqi, China.
[5]Centre of Frontier AI Research (CFAR), A*STAR, Singapore
liqz@stu.scu.edu.cn, xywang@stu.scu.edu.cn, yingke.chen@northumbria.ac.uk, yanmingtop@gmail.com, pengdz@scu.edu.cn, pengx.gm@gmail.com, wangxu.scu@gmail.com

Abstract

Recent advancements in Pre-trained Language Models (PLMs) have significantly enhanced performance across various Natural Language Processing (NLP) tasks. However, the variability in data distributions across different domains presents challenges in generalizing these models to unseen domains. Domain generalization offers a promising solution, but existing text domain generalization methods typically rely on adversarial training to learn domain-invariant features, which often leads to models with high computational and memory overhead. To address this issue, this paper proposes a novel solution named Generalization via Prompts and Contrastive Learning (GenPromptCL) to enhance the generalization capability in domain generalization. GenPromptCL consists of two key components: Domain-Misleading Prompt Learning (DMPL) and Pseudo Label-based Contrastive Learning (PCL). Specifically, DMPL disrupts domain labels randomly, misleading the model into producing incorrect domain labels. This forces the model to learn domain-invariant features. Meanwhile, PCL generates pseudo labels within a single mini-batch, enabling the model to learn both intra-class and inter-class discriminative representations with low time and space complexity. Extensive experimental results demonstrate that GenPromptCL achieves state-of-the-art performance on three distinct text classification tasks (sentiment analysis, rumor detection, and natural language inference) while significantly improving model operation efficiency.

Code — https://github.com/Balding-Lee/GenPromptCL

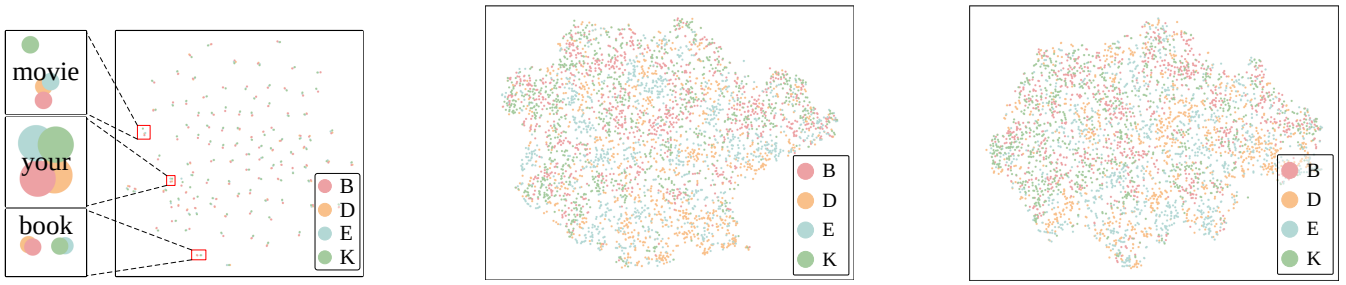
Introduction

Domain generalization (DG) aims to train a model on multiple source domains with varying distributions, enabling it to generalize well to unseen target domains (Wang et al. 2023a). While DG has been extensively studied in the field of computer vision (CV)—where humans can easily distinguish between domains in different images (e.g., rainy, foggy, sunny) (Wang et al. 2023c; Yao et al. 2024; Yin et al. 2025; Liu et al. 2024; Wen et al. 2024). In contrast, DG has garnered less attention in natural language processing (NLP) compared to CV, primarily because pre-trained language models (PTLMs) (Devlin et al. 2019; Liu et al. 2019; Lei et al. 2022) already demonstrate strong performance in general scenarios.

However, recent studies (Guo and Yu 2022; Ding et al. 2022; Ling et al. 2024; Yang et al. 2024) have shown that even large language models (LLMs) struggle with domain gaps, particularly in sensitive areas such as medical and legal texts, where privacy concerns often prevent access to domain-specific data during training. For example, in medical scenarios, models trained in general corpora frequently fail to capture the nuances in medical terminology, resulting in suboptimal predictions. Additionally, PDA (Jia and Zhang 2022) attempts to minimize the distance between word and sentence representations from different domains, indicating that domain discrepancies persist in natural language. Furthermore, our experiments reveal that, in sentiment analysis tasks, identical words exhibit different distributions across domains (as Fig. 1a shows), and sentences with the same sentiment polarity also show distinct representation distributions (as Figs. 1b and 1c show). These findings highlight the persistent challenge posed by domain differences in natural language and underscore the need for robust domain generalization methods to ensure consistent performance across unseen domains without requiring additional training.

Existing text domain generalization methods can be broadly divided into two main approaches: 1) Learning domain invariant representations via adversarial training. These methods adopt adversarial training to learn domain-invariant representations by designing n domain discriminators (n is the number of source domains) (Wang et al. 2019b; Jia and Zhang 2022; Bhattacharjee et al. 2024). The discriminators aim to classify the domains of the input data as accurately as possible, while the model generates representations that are challenging for the discriminators to classify correctly, thus promoting the learning of domain-invariant features. 2) Learning intra- and inter-class distributions via contrastive learning. Contrastive learning is widely used in cross-domain learning, as it helps cluster data effectively and

*Corresponding author.



(a) 100 words representations. (b) Positive sentence representations. (c) Negative sentence representations.

Figure 1: We use RoBERTa-base to extract the last hidden states of words and sentences from four domains (book (B), dvd (D), electronics (E), kitchen (K)) in the Amazon dataset and adopt t-SNE to visualize their distributions. Different colors indicate different domains. (a) Representations of the top 100 most frequent words. (b) Representations of positive sentences. And (c) representations of negative sentences.

enables the model to learn both intra- and inter-class relationships (Tan et al. 2022; Pu et al. 2025; Wang et al. 2024b, 2025). Most of these methods rely on data augmentation techniques to generate positive and negative samples (Wang et al. 2023b; Bhattacharjee et al. 2023; Wang et al. 2022). Although these methods improve model generalization, they still face several limitations: (1) Adversarial training suffers from high computational complexity. In classification tasks with n source domains and k categories, the computational complexity of adversarial training is $O(k{\cdot}n^2)$ (Jia and Zhang 2022). (2) Data augmentation methods often fail to enhance performance in NLP tasks. Research has shown that text data augmentation does not always improve model performance and may even lead to degradation (Kobayashi 2018; Wei and Zou 2019). Furthermore, it can disrupt the syntactic structure and semantic integrity of the text (Fadaee, Bisazza, and Monz 2017).

To address the challenges outlined above, we propose a new text domain generalization framework named ***Gen****eralization via* ***Prompt****s and* ***C****ontrastive* ***L****earning (GenPromptCL)*. Specifically, GenPromptCL contains two innovative modules: Domain Misleading Prompt Learning (DMPL) and Pseudo Label-based Contrastive Learning (PCL). In contrast to adversarial training methods for learning domain-invariant features, the DMPL module employs prompt-based learning to guide the model in predicting the domain of the data, while intentionally misleading the model by assigning incorrect domain labels. **This strategy ensures that the model effectively learns domain-invariant features without requiring additional trainable parameters or increasing runtime**. Meanwhile, the PCL module facilitates the learning of both inter- and intra-class discriminative features within a mini-batch, without the need for data augmentation. It constructs a pseudo-label matrix that reduces the discriminability of intra-class data while enhancing the discriminability of inter-class data. **This approach enables the model to capture the discriminative features of the data better, ensuring improved generalization**.

The main contributions and novelties of this work are shown as follows:

- We introduce a new text domain generalization approach named GenPromptCL. This method significantly improves the domain generalization performance in text classification while reducing the learning cost.
- We present a novel Domain-Misleading Prompt Learning strategy (DMPL) for learning domain-invariant representations. This module achieves better domain-invariant feature learning with lower computational complexity compared to adversarial training approaches.
- We elaborate on a simple yet effective Pseudo Label-based Contrastive Learning (PCL) module for clustering samples, enabling the model to learn discriminative information. PCL eliminates the need for data augmentation, thus reducing computational cost.
- We conduct extensive experiments on three text classification tasks-sentiment analysis, rumor detection, and natural language inference-demonstrating that GenPromptCL achieves state-of-the-art performance in domain generalization for text classification.

Related Work

Domain Generalization. Domain generalization methods aim to train a model on multiple source domains with varying data distributions and enable the model to generalize to unseen target domains (Wang et al. 2023a). Existing approaches focus on training the model to learn both domain-invariant features and discriminative features between different categories. Some methods employ min-max adversarial training to learn domain-invariant representations, as adversarial training can encourage the model to generate domain-invariant features (Jia and Zhang 2022; Bhattacharjee et al. 2024). However, adversarial training is computationally expensive. To enable the model to learn discriminative features between categories, several methods use data augmentation techniques to generate positive and negative samples, followed by contrastive learning to enhance feature discrimination (Wang et al. 2024a). However, data augmentation in NLP often disrupts the semantic and syntactic integrity of the text (Fadaee, Bisazza, and Monz 2017) and its effectiveness is inconsistent, sometimes failing to yield promising results (Kobayashi 2018; Wei and Zou 2019). Moreover, data

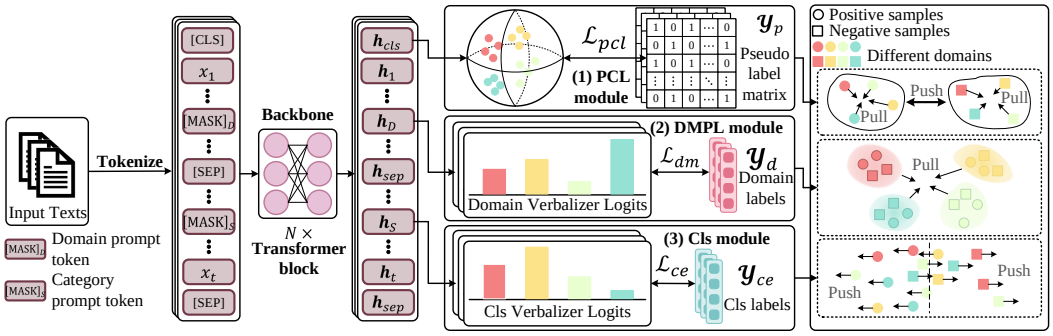


Figure 2: The model architecture of the proposed method. First, we tokenize the input texts and concatenate tokens with domain-misleading and classification prompts. Then, we feed tokens into the backbone model and output the hidden states of [CLS], $[\text{MASK}]_S$, and $[\text{MASK}]_D$. After that, we perform pseudo-label-based contrastive learning on [CLS], domain-misleading prompt learning on $[\text{MASK}]_D$, and discriminative learning on $[\text{MASK}]_S$. Finally, we adopt the losses obtained above to train our model.

augmentation-based methods increase the model's training cost (Li et al. 2025a,b). In contrast to these computationally intensive approaches, **our method seeks to adopt a low-cost strategy to learn domain-invariant features and both intra- and inter-class discriminative information efficiently**.

Prompt Learning of Masked Language Models. Traditional pre-trained language models typically fine-tune task-specific classifiers for downstream tasks, which limits the effective utilization of the pre-trained knowledge embedded in the models (Liu et al. 2023). Prompt learning aims to efficiently utilize the pre-trained parameters without adding additional trainable parameters that require fine-tuning. Prompt learning based on masked language model (such as BERT (Devlin et al. 2019), RoBERTa (Liu et al. 2019)) treats the task as a "cloze" task (Lin et al. 2020). For instance, given the sentence "I love this movie." with the prompt "Overall, it was *[MASK]*.", the model can directly predict *[MASK]* as *good* without fine-tuning any parameters. In addition to cloze tests, scholars also use learnable prompt templates to enable the model to learn better prompt templates, thereby enhancing the model's performance (Wen et al. 2025). Moreover, in domain generalization, some methods leverage prompt learning to generate continuous prompts from input samples, allowing the model to infer the domain of the input data (Zhou et al. 2022; Bose et al. 2024). However, continuous prompts are often not easily interpretable, leading to weaker interpretability. Inspired by the advantages of prompt learning, we adopt this parameter-efficient fine-tuning approach to reduce the computational cost. Additionally, we use manually designed prompts to improve the interpretability of the model.

Methodology

Problem Formulation

Problem Definition. The entire dataset $\mathcal{D}$ is divided into $N(N > 2)$ domains $\mathcal{D} = \{\boldsymbol{\mathcal{X}}^i\}_{i=1}^N$. Each domain consists texts and labels, where $\boldsymbol{\mathcal{X}}^i = \{\mathcal{T}_j^i, y_j^i\}_{j=1}^{n^i}$, with n^i represents the number of samples in the i-th domain. Here, $\mathcal{T}_j^i$ and y_j^i refer to the j-th text and its corresponding label in the i-th domain, respectively. Meanwhile, domains are characterized by different data distributions, i.e., $P(\boldsymbol{\mathcal{T}}^i) \neq P(\boldsymbol{\mathcal{T}}^j)(i \neq j)$, where $\boldsymbol{\mathcal{T}}^i = \{\mathcal{T}_j^i\}_{j=1}^{n^i}$ and $P(\boldsymbol{\mathcal{T}}^i)$ denotes the data distribution of the i-th domain. However, the label space is consistent across domains, i.e., $\boldsymbol{\mathcal{Y}}^i = \boldsymbol{\mathcal{Y}}^j$, where $\boldsymbol{\mathcal{Y}}^i = \{y_j^i\}_{j=1}^c$, and c represents the number of categories. We use the leave-one-domain-out evaluation method (Jia and Zhang 2022) to evaluate the generalization of our model. Specifically, we select $n = N - 1$ domains as source domains, denoted by $\boldsymbol{\mathcal{X}}_{src} = \{\boldsymbol{\mathcal{T}}_{src}^i, \boldsymbol{\mathcal{Y}}_{src}^i\}_{i=1}^n$, and leave one domain as the target domain $\boldsymbol{\mathcal{X}}_{tgt} = \{\boldsymbol{\mathcal{T}}_{tgt}, \boldsymbol{\mathcal{Y}}_{tgt}\}$. The model is trained on $\boldsymbol{\mathcal{X}}_{src}$ and tested on $\boldsymbol{\mathcal{X}}_{tgt}$.

Overview. The model architecture of GenPromptCL is shown in Fig. 2. Specifically, GenPromptCL adopts the Domain-Misleading Prompt Learning (DMPL) module to learn domain-invariant features. Meanwhile, GenPromptCL utilizes the Pseudo Label Contrastive Learning (PCL) module to capture inter- and intra-class discriminative information. Furthermore, GenPromptCL includes a classification module to distinguish between different categories. These three objectives are jointly optimized to enhance the performance of domain generalization in text classification.

Learning Discriminative Representations via Classification Loss

We use a prompt-based classification method to train the model, aligning text features with corresponding labels to improve its discriminative power.

Specifically, let the input texts be $\mathcal{T}_{src} = \{\mathcal{T}_{src}^i\}_{i=1}^n$. We first concatenate each text with two manually designed prompt templates $temp_S$ and $temp_D$. Both templates contain prompt tokens with a mask token. The mask token of $temp_S$, denoted as $[\text{MASK}]_S$, is used to predict the category label of the input text. Similarly, $temp_D$ has a mask

token $[\mathrm{MASK}]_D$, which is used to predict the domain label. In addition, following the tradition of masked language models, we concatenate the special tokens [CLS] and [SEP] to the input text. Thus, the final input texts are denoted as:

$$\tilde{T}_{src} = [\mathrm{CLS}]\ temp_D\ [\mathrm{SEP}]\ temp_S\ \mathcal{T}_{src}\ [\mathrm{SEP}]. \quad (1)$$

After constructing the input texts, we feed them into a pre-trained masked language model $\mathcal{M}$ to obtain the text representations,

$$\boldsymbol{h} = \mathcal{M}(\tilde{\mathcal{T}}_{src}; \theta_{\mathcal{M}}), \quad (2)$$

where $\theta_{\mathcal{M}}$ represents the trainable parameters of $\mathcal{M}$. The output $\boldsymbol{h}$ contains the hidden states of $[\mathrm{MASK}]_D$ (denoted as $\boldsymbol{h}_D$) and $[\mathrm{MASK}]_S$ (denoted as $\boldsymbol{h}_S$). Both $\boldsymbol{h}_D$ and $\boldsymbol{h}_S$ are matrices of dimension $\boldsymbol{h}_D, \boldsymbol{h}_S \in \mathbb{R}^{b \times V}$, where b is the batch size and V is the vocabulary size.

In the prompt learning process, the classification module retrieves the answer tokens for prompt learning from the category verbalizer $\mathcal{V}_S$. It then identifies the corresponding hidden states for these tokens in $\boldsymbol{h}_S$. Finally, the classification module adopts softmax to compute the pseudo-distributions of these logits and extracts the word with the highest probability as the predicted category label $\tilde{y}_{ce}$. The process is formulated as follows,

$$\tilde{y}_{ce} = \arg \max_{v_i^S \in \mathcal{V}_S} \frac{\exp(f(\boldsymbol{h}_S, v_i^S))}{\sum_{j=1}^{n_S} \exp(f(\boldsymbol{h}_S, v_j^S))}, \quad (3)$$

where n_S is the number of words in $\mathcal{V}_S$, v_i^S is the i-th word in $\mathcal{V}_S$, and $f(\boldsymbol{h}_S, v_i^S)$ represents the function that computes the logits for the i-th word in $\boldsymbol{h}_S$.

Finally, GenPromptCL computes the cross-entropy loss between $\tilde{y}_{ce}$ and the true category label y_{ce}, as follows:

$$\mathcal{L}_{ce} = -\sum_{i=1}^{N_{src}} y_{ce}^i \log(\tilde{y}_{ce}^i), \quad (4)$$

where N_{src} is the number of training samples in the source domains.

Learning Domain-Invariant Representations via Domain Misleading Prompt Learning

The Classification module aligns sentence representations with the category labels of the samples, enabling the model to learn category-specific information. However, this alignment also amplifies domain differences. To mitigate this, GenPromptCL incorporates the Domain-Misleading Prompt Learning (DMPL) module to learn domain-invariant features while maintaining low computational complexity.

In Equation (2), we obtain the hidden states $\boldsymbol{h}_D$ corresponding to the domain mask token $[\mathrm{MASK}]_D$. Using Equation (3), we then extract the predicted domain label $\tilde{y}_D$ from the domain verbalizer $\mathcal{V}_D$.

Traditional methods adopt cross-entropy to compute the difference between the predicted domain label $\tilde{y}_D$ and the true domain label y_D. To encourage the model to learn domain-invariant representations, the DMPL module introduces domain-misleading labels during training. Specifically, for the i-th input text belonging to the j-th source domain $\mathcal{T}_{src}^i \in \boldsymbol{D}_{src}^j$, we randomly alter the domain label. First, let the set of domain labels satisfy the uniform distribution $\boldsymbol{D}_{src} \sim \mathrm{Uniform}(0, n)$. At each step, we select a new domain label y_r from this distribution and replace the original label y_D with y_r, provided that $y_r \neq y_D$, otherwise, we re-draw the label. This process is formally defined in Equation (5).

$$y_D^i \coloneqq y_r^i \ \text{ if } \ y_r^i \neq y_D^i. \quad (5)$$

At last, we adopt cross-entropy loss to compute the domain misleading loss $\mathcal{L}_{dm}$, as follows,

$$\mathcal{L}_{dm} = -\sum_{i=1}^{N_{src}} y_D^i \log(\tilde{y}_D^i). \quad (6)$$

Since the domain labels are randomly shuffled, the model finds it challenging to accurately assign data to the correct domain during classification. This encourages the model to focus more on domain-invariant features in the text, ensuring better classification accuracy on downstream tasks.

Learning Intra- and Inter-Class Distributions via Pseudo Label-based Contrastive Learning

The DMPL module helps the model learn domain-invariant features by bringing data from different domains closer together. However, this comes at the cost of reducing the discriminability between data from different categories. Traditional methods often use data augmentation-based contrastive learning, such as synonym replacement, to enhance discriminative learning between samples (Wang et al. 2023b; Bhattacharjee et al. 2023, 2024). However, synonym replacement can disrupt the text's syntactic and semantic structures, often leading to undesirable outcomes in NLP (Kobayashi 2018; Wei and Zou 2019; Fadaee, Bisazza, and Monz 2017). For instance, "A dog bit a man." might be replaced by "A dog became a man.", resulting in an illogical statement. To address this issue, the PCL module introduces a pseudo-label matrix as supervision for contrastive learning. This matrix captures the similarity or dissimilarity between samples from different domains, enabling the model to learn discriminative features between these samples within a single batch.

Specifically, given a batch of data $\boldsymbol{\mathcal{T}}_B \in \mathbb{R}^b$ with corresponding labels $\boldsymbol{\mathcal{Y}}_B \in \mathbb{R}^b$, it contains data from all source domains. We first construct pseudo-labels $\boldsymbol{\mathcal{Y}}_p$ via $\boldsymbol{\mathcal{Y}}_B$. If the label of the i-th text is the same as the label of the j-th text, then the pseudo-label y_p^{ij} is 1, otherwise is 0:

$$y_p^{ij} = \begin{cases} 1 & \text{if } y_i = y_j \\ 0 & \text{otherwise} \end{cases}. \quad (7)$$

The pseudo-label matrix $\boldsymbol{\mathcal{Y}}_p$ is then represented as:

$$\boldsymbol{\mathcal{Y}}_p = \begin{bmatrix} 1 & y_p^{12} & \cdots & y_p^{1b} \\ y_p^{21} & 1 & \cdots & y_p^{2b} \\ \vdots & \vdots & \ddots & \vdots \\ y_p^{b1} & y_p^{b2} & \cdots & 1 \end{bmatrix}. \quad (8)$$

Model	DEK→B	BEK→D	BDK→E	BDE→K	Avg.
SimCSE	91.17 ± 0.16	90.10 ± 0.39	$\mathbf{92.27 \pm 0.22}$	93.87 ± 0.05	91.85 ± 0.21
SwAV	90.83 ± 0.10	89.90 ± 0.41	92.00 ± 0.29	93.65 ± 0.32	91.60 ± 0.28
IRM	55.34 ± 5.74	51.19 ± 5.44	60.56 ± 2.62	64.40 ± 1.07	57.87 ± 2.88
DeepCORAL	77.23 ± 3.23	81.55 ± 0.80	83.98 ± 0.71	80.56 ± 2.93	80.83 ± 1.92
MSCL	91.10 ± 0.33	89.78 ± 0.28	92.10 ± 0.16	93.50 ± 0.18	91.60 ± 0.05
PDA	91.28 ± 0.31	90.17 ± 0.38	91.70 ± 0.15	93.47 ± 0.09	91.65 ± 0.17
EAGLE	91.20 ± 0.26	90.00 ± 0.23	92.15 ± 0.22	93.78 ± 0.09	91.78 ± 0.15
TACIT	89.75 ± 0.53	89.05 ± 0.17	91.17 ± 0.48	93.60 ± 0.29	90.89 ± 0.37
DomCLP	90.88 ± 0.23	89.20 ± 0.27	92.03 ± 0.30	93.20 ± 0.16	91.33 ± 0.24
GenPromptCL (ours)	$\mathbf{91.67 \pm 0.15}$	$\mathbf{90.18 \pm 0.02}$	$\mathbf{92.27 \pm 0.09}$	$\mathbf{94.03 \pm 0.13}$	$\mathbf{92.04 \pm 0.10}$

Table 1: The comparison experimental results on the Amazon dataset (%). We bold the best results.

Model	CH	FG	GW	OS	SS	Avg.
SimCSE	80.35 ± 0.63	68.25 ± 1.61	71.20 ± 1.77	76.92 ± 0.92	76.16 ± 0.97	74.57 ± 1.18
SwAV	79.59 ± 0.11	66.22 ± 1.50	77.54 ± 1.56	77.92 ± 0.28	76.09 ± 1.32	75.47 ± 0.95
IRM	24.97 ± 8.49	34.81 ± 6.97	41.44 ± 19.62	19.98 ± 27.76	51.89 ± 5.31	34.62 ± 13.63
DeepCORAL	78.71 ± 0.92	65.80 ± 1.07	68.39 ± 0.46	75.06 ± 2.08	73.36 ± 0.87	72.26 ± 1.08
MSCL	78.74 ± 0.37	64.03 ± 1.69	76.50 ± 0.45	74.30 ± 0.78	73.84 ± 1.73	73.48 ± 1.00
PDA	43.81 ± 0.00	51.35 ± 9.20	39.87 ± 9.72	32.06 ± 0.00	36.76 ± 0.50	40.77 ± 3.88
EAGLE	$\mathbf{80.39 \pm 0.66}$	66.17 ± 0.60	72.53 ± 1.65	78.88 ± 0.21	76.16 ± 0.64	74.83 ± 0.75
TACIT	43.81 ± 0.00	42.91 ± 0.00	33.00 ± 0.00	32.06 ± 0.00	36.41 ± 0.00	37.64 ± 0.00
DomCLP	79.74 ± 0.75	68.98 ± 1.62	72.54 ± 0.43	75.20 ± 1.84	73.96 ± 0.43	74.08 ± 1.01
GenPromptCL (ours)	80.01 ± 0.04	$\mathbf{73.02 \pm 0.25}$	$\mathbf{79.34 \pm 0.06}$	$\mathbf{79.56 \pm 0.21}$	$\mathbf{78.72 \pm 0.14}$	$\mathbf{78.18 \pm 0.14}$

Table 2: The comparison experimental results on the PHEME dataset (%). We bold the best results. CH means the target domain is CH, while FG, GW, OS, SS are the source domains (FG, GW, OS, SS→CH), and the other abbreviations follow this rule.

Note that: 1) the main diagonal of $\boldsymbol{\mathcal{Y}}_p$ consists of ones; and 2) $\boldsymbol{\mathcal{Y}}_p$ is a symmetric matrix, meaning $y_p^{ij} = y_p^{ji}$. Since each batch contains data from all source domains, we argue that data with the same label but from different domains should have similar hidden states, and vice versa. To achieve this, we first feed the sentence representation (the hidden state of [CLS]) of the i-th data sample, denoted as $\boldsymbol{h}_{cls}^i$, into a fully connected layer:

$$\boldsymbol{O}_i = \sigma(\boldsymbol{W}_f^{\mathrm{T}} \boldsymbol{h}_{cls}^i + \boldsymbol{b}_f), \tag{9}$$

where $\sigma(\cdot)$ is the activation function; $\boldsymbol{W}_f$ and $\boldsymbol{b}_f$ are the weights and biases of the fully connected layer, respectively. Then, we compute the similarity between the i-th data and the j-th data using the inner product (Lewis et al. 2020):

$$s_{ij} = \mathrm{Sigmoid}(\boldsymbol{O}_i^{\mathrm{T}} \boldsymbol{O}_j), \tag{10}$$

where $\mathrm{Sigmoid}(\cdot)$ maps the values to the range $(0, 1)$, with a value close to 1 indicating high similarity between the i-th and j-th data samples. Finally, we apply the cross-entropy loss to keep similar data as close as possible and dissimilar data as far away as possible:

$$\mathcal{L}_{pcl} = -\frac{1}{b^2} \sum_{i=1}^{b} \sum_{j=1}^{b} \left[y_p^{ij} \log(s_{ij}) + (1 - y_p^{ij}) \log(1 - s_{ij}) \right]. \tag{11}$$

Objective Function

After the data passes through the three modules, we obtain three separate losses $\mathcal{L}_{ce}$, $\mathcal{L}_{dm}$, and $\mathcal{L}_{pcl}$. They are combined into a total loss for joint learning, defined as:

$$\mathcal{L} = \lambda_{ce} \mathcal{L}_{ce} + \lambda_{dm} \mathcal{L}_{dm} + \lambda_{pcl} \mathcal{L}_{pcl},$$

where λ_{ce}, λ_{dm}, and λ_{pcl} are hyperparameters that balance the contributions of each loss. However, selecting these hyperparameters can be costly and time-consuming.

To mitigate this, we adopt the Cov-Weighting loss (Groenendijk et al. 2021), which dynamically adjusts the weights based on the loss reduction rate. This method is parameter-free, enabling the model to autonomously adjust the importance of each loss term at different training stages, leading to more efficient and adaptive optimization. The final objective function is:

$$\mathcal{L} = \mathrm{COV}\left([\mathcal{L}_{ce}, \mathcal{L}_{dm}, \mathcal{L}_{pcl}]\right), \tag{12}$$

where $\mathrm{COV}(\cdot)$ is the cov-weighting loss function.

Experiments

Datasets and Baseline Methods

We evaluate our model on three text classification tasks: sentiment analysis (SA), rumor detection (RD), and natural language inference (NLI). **Sentiment Analysis**. We use the Amazon reviews (Blitzer, Dredze, and Pereira 2007), IMDB (Maas et al. 2011), and SST-2 (Socher et al. 2013) datasets. The Amazon Reviews contains four domains:

Model	GSTT'→F	FSTT'→G	FGTT'→S	FGST'→T	FGST→T'	Avg.
SimCSE	80.05 ± 0.51	84.26 ± 0.40	78.65 ± 0.14	80.05 ± 0.21	81.42 ± 0.48	80.89 ± 0.35
SwAV	80.67 ± 0.12	84.17 ± 0.21	78.49 ± 0.11	78.52 ± 1.11	81.20 ± 0.34	80.61 ± 0.38
IRM	28.82 ± 1.38	31.52 ± 0.51	31.49 ± 0.44	30.81 ± 0.90	31.34 ± 0.78	30.80 ± 0.80
DeepCORAL	20.26 ± 4.09	17.83 ± 0.00	20.88 ± 5.12	17.39 ± 0.00	17.38 ± 0.00	18.75 ± 1.84
MSCL	78.34 ± 0.04	82.28 ± 1.37	76.98 ± 0.33	78.58 ± 0.22	79.82 ± 0.57	79.20 ± 0.51
PDA	16.12 ± 0.00	29.96 ± 18.82	36.52 ± 29.42	36.41 ± 29.92	16.00 ± 0.00	27.00 ± 15.43
EAGLE	80.17 ± 0.20	84.61 ± 0.48	78.29 ± 0.39	79.50 ± 0.19	80.19 ± 0.37	80.55 ± 0.32
TACIT	78.57 ± 0.45	84.40 ± 0.60	76.62 ± 0.54	79.23 ± 0.25	79.66 ± 0.24	79.70 ± 0.42
DomCLP	78.72 ± 0.17	84.16 ± 0.46	78.27 ± 0.17	78.92 ± 0.35	80.74 ± 0.06	80.16 ± 0.24
GenPromptCL (ours)	$\mathbf{81.15 \pm 0.10}$	$\mathbf{84.84 \pm 0.02}$	$\mathbf{79.75 \pm 0.17}$	$\mathbf{80.88 \pm 0.07}$	$\mathbf{82.07 \pm 0.07}$	$\mathbf{81.74 \pm 0.09}$

Table 3: The comparison experimental results on the MNLI dataset (%). We bold the best results.

Model	Amazon→IMDB	Amazon→SST-2	MNLI→SICK	MNLI→SNLI	Avg.
SimCSE	$\mathbf{90.96 \pm 0.07}$	90.95 ± 0.56	64.13 ± 0.91	61.38 ± 0.33	76.86 ± 14.57
SwAV	90.76 ± 0.16	89.90 ± 0.52	63.56 ± 1.61	60.90 ± 0.50	76.28 ± 14.09
IRM	58.26 ± 6.31	65.88 ± 0.44	28.66 ± 1.14	19.18 ± 3.68	42.99 ± 19.55
DeepCORAL	85.82 ± 0.27	85.30 ± 2.10	24.17 ± 0.00	31.16 ± 0.00	56.61 ± 29.05
MSCL	90.82 ± 0.07	90.65 ± 0.71	65.40 ± 0.58	58.40 ± 0.86	76.32 ± 14.63
PDA	90.91 ± 0.02	90.78 ± 0.37	48.68 ± 17.66	13.54 ± 9.92	60.98 ± 32.35
EAGLE	90.51 ± 0.11	91.43 ± 0.64	$\mathbf{66.86 \pm 1.60}$	60.25 ± 0.03	77.26 ± 13.91
TACIT	90.45 ± 0.10	91.80 ± 0.14	60.05 ± 0.75	60.47 ± 0.96	75.69 ± 15.44
DomCLP	90.22 ± 0.14	91.85 ± 0.21	63.82 ± 2.96	60.19 ± 1.63	76.52 ± 14.58
GenPromptCL (ours)	90.77 ± 0.09	$\mathbf{91.96 \pm 0.09}$	66.56 ± 0.43	$\mathbf{62.04 \pm 0.26}$	$\mathbf{77.83 \pm 13.63}$

Table 4: The comparison experimental results on the IMDB, SST-2, SICK, and SNLI datasets. We bold the best values (%). RL represents the representation learning methods; DG represents the domain generalization methods.

book (B), dvd (D), electronics (E), and kitchen (K). **Rumor Detection**. We conduct experiments on the PHEME dataset (Kochkina, Liakata, and Zubiaga 2018a).[1] Following MSCL (Tan et al. 2022), we use five domains: CharlieHebdo (CH), Ferguson (FG), GermanWings (GW), OttawaShooting (OS), and SydneySiege (SS). **Nature Language Inference**. We adopt the MNLI (Wang et al. 2019a), SICK (Marelli et al. 2014), and SNLI (Kochkina, Liakata, and Zubiaga 2018b) datasets. For MNLI, following PDA (Jia and Zhang 2022), we use five domains: fiction (F), government (G), state (S), telephone (T), and travel (T'). We use the shorthand "BDK→E" to denote using B, D, K as source domains and E as the target, and apply this convention throughout.

We compare GenPromptCL with SimCSE (Gao, Yao, and Chen 2021), SwAV (Caron et al. 2020), MSCL (Tan et al. 2022), PDA (Jia and Zhang 2022), EAGLE (Bhattacharjee et al. 2024), TACIT (Song et al. 2024), IRM (Arjovsky et al. 2020), DeepCORAL (Sun and Saenko 2016), and DomCLP (Lee, Kim, and Lee 2025).

Implementation Details

We adopt RoBERTa base (Liu et al. 2019)[2] as the backbone for all models except IRM, which utilizes an MLP backbone. We set the maximum sequence length for the Amazon reviews and SST-2 to 128, IMDB to 196, PHEME to 64, and the maximum single sentence length in MNLI, SICK, and SNLI to 48. We train the models on the source domains for 30 epochs with a batch size of 16. We adopt AdamW as the optimizer with a learning rate of 1e-5 and weight decay of 1e-2. We set $\boldsymbol{O}_i \in \mathbb{R}^{256}$, and all hyperparameters of the other models strictly obey the values given in their papers. Meanwhile, we use macro-F1 as an evaluation metric. In addition, each model undergoes 3 independent runs, and we report the mean and standard deviation of the results[3]. All experiments are implemented on a single NVIDIA V100 GPU.

Individual Dataset Results

Table 1, 2, and 3 show the experimental results for individual datasets. These results demonstrate that GenPromptCL achieves the best performance across all three tasks. Specifically, the proposed model outperforms the second-best methods by: 0.19% on the Amazon dataset, 2.71% on the PHEME dataset, and 0.85% on the MNLI dataset. From these findings, we draw the following conclusions: 1) Under the same settings, PDA experiences a significant performance drop in the PHEME and the MNLI datasets, due to the instability introduced by adversarial training. In contrast, GenPromptCL consistently performs well across all datasets, demonstrating both stability and robustness. 2) Methods such as DeepCORAL excel in single-sentence classification but struggle with multi-sentence tasks. This high-

[1]https://github.com/kochkinaelena/Multitask4Veracity

[2]https://huggingface.co/FacebookAI/roberta-base

[3]We reproduce all baseline models and report the results.

Model	Runtime (↓)			Param. (↓)		
	Amazon	IMDB	SST-2	Amazon	IMDB	SST-2
MSCL	**51.74** (12.32 ↓)	190.03 (37.47 ↑)	**68.33** (16.53 ↓)	124.84 (99.84 ↑)	124.84 (99.84 ↑)	124.84 (99.84 ↑)
PDA	65.26 (10.95 ↑)	129.70 (8.38 ↑)	91.13 (12.62 ↑)	1.77 (88.90 ↑)	3.55 (94.45 ↑)	3.55 (94.45 ↑)
EAGLE	83.87 (30.71 ↑)	171.98 (30.91 ↑)	116.23 (31.49 ↑)	2.00 (90.18 ↑)	3.78 (94.79 ↑)	3.78 (94.79 ↑)
TACIT	69.57 (16.47 ↑)	141.54 (16.05 ↑)	142.89 (44.28 ↑)	82.88 (99.76 ↑)	82.88 (99.76 ↑)	82.88 (99.76 ↑)
GenPromptCL	58.11	**118.82**	79.62	**0.20**	**0.20**	**0.20**

Table 5: The runtime within one epoch (s) and the number of additional parameters of domain generalization models (M). The parentheses indicate how much the model improves or reduces compared to GemPromptCL (%) ($(T_m - T_{ours})/T_m$, T_m is the runtime of the comparison model, T_{ours} is the runtime of our model, and the same formula applies to the calculation of the number of additional parameters). Param. indicates the number of additional parameters.

Module	DEK→B	BEK→D	BDK→E	BDE→K
GenPromptCL	**91.67**	**90.18**	**92.27**	**94.03**
w/o DMPL	91.13	89.53	91.68	93.48
w/o PCL	91.08	89.72	91.82	93.42
w/o both	91.08	89.45	91.83	93.47

Table 6: The ablation study on the Amazon dataset (%). We bold the best values. w/o DMPL indicates ablation of the DMPL module; w/o PCL represents ablation of the PCL module; w/o both stands for ablation of the DMPL and PCL components.

lights the ability of GenPromptCL to handle both single-sentence and multi-sentence classification tasks effectively. And 3) GenPromptCL exhibits low standard deviations across all domains, with a maximum standard deviation of only 0.25%, further underscoring its high stability and reliability.

Cross Datasets Results

Table 4 shows the experimental results on the IMDB, SST-2, SICK, and SNLI datasets. These results reveal the following key observations: 1) GenPromptCL achieves the best results (77.83%) on the IMDB and SST-2 datasets. 2) Although GenPromptCL does not achieve the top results on the IMDB and SICK datasets, it ranks as the second-best model. And 3) GenPromptCL achieves the highest average macro-F1 score across the four datasets, surpassing the second-best model by 0.57%. From these findings, we can conclude: 1) Consistent with the results on individual datasets, GenPromptCL exhibits remarkable stability across all datasets, with a maximum standard deviation of only 0.43%. And 2) The proposed method demonstrates superior stability and robustness, delivering consistent and competitive results in both single-sentence and multi-sentence classification tasks, as well as across individual and multiple datasets.

Model Operation Efficiency Comparison

In this subsection, we evaluate the efficiency of domain generalization models by examining the runtime per epoch during training and the number of additional parameters required beyond the backbone model. A detailed analysis of the time and space complexity of GenPromptCL is provided in Supplementary. To ensure fairness, only one model is run on the machine at a time during the experiments. Since the data volume is consistent across domains in the Amazon dataset, we measure runtime and additional parameters on a single domain from the Amazon dataset.

Table 5 presents the results of the efficiency experiments. From these results, we can conclude: 1) Although GenPromptCL is not always the fastest in terms of runtime (12.32% and 16.53% slower than MSCL on Amazon and SST-2, respectively), it significantly outperforms MSCL in parameter efficiency, requiring only 0.2M additional parameters, compared to MSCL, which uses 99.84% more additional parameters than our model. 2) Unlike adversarial training-based models (e.g., PDA and EAGLE), GenPromptCL does not incur a parameter increase as the number of domains grows. And 3) while models like MSCL and TACIT maintain constant parameter counts regardless of domain variations, their additional parameter counts are 99.84% and 99.76% higher than those of GenPromptCL, respectively.

Ablation Study

The proposed method includes two components: DMPL and PCL. To investigate the individual contributions of these modules, we conduct ablation studies on the Amazon dataset. Table 6 reports the experimental results. From these results, we observe that both modules significantly contribute to the overall performance improvement of the model. This suggests that the DMPL module enables the model to learn domain-invariant representations, while the PCL module helps the model learn discriminative features. These findings highlight the complementary roles of DMPL and SCL in enhancing the model's effectiveness.

Conclusion

In this paper, we propose a novel text domain generalization method, GenPromptCL, designed to address the high complexity of adversarial training and the challenges associated with data augmentation in NLP tasks. We conduct comprehensive experiments on three text classification tasks, demonstrating that GenPromptCL outperforms existing methods in domain generalization. Additionally, efficiency evaluations reveal that the proposed method effectively reduces both time and space complexities, offering a more efficient solution for domain generalization in NLP.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (62372315, 62306197), Sichuan Science and Technology Planning Project (2025ZNSFSC1507, 2024YFG0007, 2024ZDZX0004, 2024NSFTD0049), Central Government's Guide to Local Science and Technology Development Fund (2025ZYDF101), China Postdoctoral Science Foundation (2021TQ0223, 2022M712236), Chengdu Science and Technology Project (2023-XT00-00004-GX), Postdoctoral Joint Training Program of Sichuan University (SCDXLHPY2307).

References

Arjovsky, M.; Bottou, L.; Gulrajani, I.; and Lopez-Paz, D. 2020. Invariant Risk Minimization. arXiv:1907.02893.

Bhattacharjee, A.; Kumarage, T.; Moraffah, R.; and Liu, H. 2023. ConDA: Contrastive Domain Adaptation for AI-generated Text Detection. In *IJCNLP 2023*, 598–610. Association for Computational Linguistics.

Bhattacharjee, A.; Moraffah, R.; Garland, J.; and Liu, H. 2024. EAGLE: A Domain Generalization Framework for AI-generated Text Detection. arXiv:2403.15690.

Blitzer, J.; Dredze, M.; and Pereira, F. 2007. Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In *ACL 2007*. The Association for Computational Linguistics.

Bose, S.; Jha, A.; Fini, E.; Singha, M.; Ricci, E.; and Banerjee, B. 2024. StyLIP: Multi-Scale Style-Conditioned Prompt Learning for CLIP-based Domain Generalization. In *WACV 2024*, 5530–5540. IEEE.

Caron, M.; Misra, I.; Mairal, J.; Goyal, P.; Bojanowski, P.; and Joulin, A. 2020. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In *NeurIPS 2020*.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 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*, 4171–4186. Association for Computational Linguistics.

Ding, N.; Qin, Y.; Yang, G.; Wei, F.; Yang, Z.; Su, Y.; Hu, S.; Chen, Y.; Chan, C.; Chen, W.; Yi, J.; Zhao, W.; Wang, X.; Liu, Z.; Zheng, H.; Chen, J.; Liu, Y.; Tang, J.; Li, J.; and Sun, M. 2022. Delta Tuning: A Comprehensive Study of Parameter Efficient Methods for Pre-trained Language Models. *CoRR*, abs/2203.06904.

Fadaee, M.; Bisazza, A.; and Monz, C. 2017. Data Augmentation for Low-Resource Neural Machine Translation. In *ACL 2017*, 567–573. Association for Computational Linguistics.

Gao, T.; Yao, X.; and Chen, D. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *EMNLP 2021*, 6894–6910. Association for Computational Linguistics.

Groenendijk, R.; Karaoglu, S.; Gevers, T.; and Mensink, T. 2021. Multi-Loss Weighting with Coefficient of Variations. In *WACV 2021*, 1468–1477. IEEE.

Guo, X.; and Yu, H. 2022. On the Domain Adaptation and Generalization of Pretrained Language Models: A Survey. *CoRR*, abs/2211.03154.

Jia, C.; and Zhang, Y. 2022. Prompt-based Distribution Alignment for Domain Generalization in Text Classification. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022*, 10147–10157. Association for Computational Linguistics.

Kobayashi, S. 2018. Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations. In *NAACL-HLT 2018*, 452–457. Association for Computational Linguistics.

Kochkina, E.; Liakata, M.; and Zubiaga, A. 2018a. All-in-one: Multi-task Learning for Rumour Verification. In *COLING 2018*, 3402–3413. Association for Computational Linguistics.

Kochkina, E.; Liakata, M.; and Zubiaga, A. 2018b. All-in-one: Multi-task Learning for Rumour Verification. In *COLING 2018*, 3402–3413. Association for Computational Linguistics.

Lee, J.; Kim, N.; and Lee, J. 2025. DomCLP: Domain-wise Contrastive Learning with Prototype Mixup for Unsupervised Domain Generalization. In *AAAI 2025*, 18119–18127. AAAI Press.

Lei, T.; Hu, H.; Luo, Q.; Peng, D.; and Wang, X. 2022. Adaptive Meta-learner via Gradient Similarity for Few-shot Text Classification. In *COLING 2022*, 4873–4882. International Committee on Computational Linguistics.

Lewis, P. S. H.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.; Rocktäschel, T.; Riedel, S.; and Kiela, D. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*.

Li, Q.; Li, X.; Chang, Z.; Zhang, Y.; Ji, C.; and Wang, S. 2025a. Multimodal Knowledge Retrieval-Augmented Iterative Alignment for Satellite Commonsense Conversation. In *IJCAI 2025*, 8168–8176. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Li, Q.; Liang, S.; Zhang, Y.; Ji, C.; Chang, Z.; and Wang, S. 2025b. Meta-Knowledge Path Augmentation for Multi-Hop Reasoning on Satellite Commonsense Multi-Modal Knowledge Graphs. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 7568–7577.

Lin, B. Y.; Lee, S.; Khanna, R.; and Ren, X. 2020. Birds have four legs?! NumerSense: Probing Numerical Commonsense Knowledge of Pre-Trained Language Models. In *EMNLP 2020*, 6862–6868. Association for Computational Linguistics.

Ling, C.; Zhao, X.; Lu, J.; Deng, C.; Zheng, C.; Wang, J.; Chowdhury, T.; Li, Y.; Cui, H.; Zhang, X.; Zhao, T.; Panalkar, A.; Mehta, D.; Pasquali, S.; Cheng, W.; Wang, H.; Liu, Y.; Chen, Z.; Chen, H.; White, C.; Gu, Q.; Pei, J.; Yang, C.; and Zhao, L. 2024. Domain Specialization as the Key to Make Large Language Models Disruptive: A Comprehensive Survey. arXiv:2305.18703.

Liu, H.; Ma, Y.; Yan, M.; Chen, Y.; Peng, D.; and Wang, X. 2024. DiDA: Disambiguated Domain Alignment for Cross-Domain Retrieval with Partial Labels. In *AAAI 2024*, 3612–3620. AAAI Press.

Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2023. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys*, 55(9): 195:1–195:35.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. abs/1907.11692.

Maas, A. L.; Daly, R. E.; Pham, P. T.; Huang, D.; Ng, A. Y.; and Potts, C. 2011. Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 142–150. Association for Computational Linguistics.

Marelli, M.; Menini, S.; Baroni, M.; Bentivogli, L.; Bernardi, R.; and Zamparelli, R. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *LREC 2014*, 216–223. European Language Resources Association (ELRA).

Pu, R.; Sun, Y.; Qin, Y.; Ren, Z.; Song, X.; Zheng, H.; and Peng, D. 2025. Robust Self-Paced Hashing for Cross-Modal Retrieval with Noisy Labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 19969–19977.

Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Manning, C. D.; Ng, A. Y.; and Potts, C. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *EMNLP 2013*, 1631–1642. ACL.

Song, R.; Giunchiglia, F.; Li, Y.; Tian, M.; and Xu, H. 2024. TACIT: A Target-Agnostic Feature Disentanglement Framework for Cross-Domain Text Classification. In *AAAI 2024*, 18999–19007. AAAI Press.

Sun, B.; and Saenko, K. 2016. Deep CORAL: Correlation Alignment for Deep Domain Adaptation. In *ECCV 2016*, volume 9915 of *Lecture Notes in Computer Science*, 443–450.

Tan, Q.; He, R.; Bing, L.; and Ng, H. T. 2022. Domain Generalization for Text Classification with Memory-Based Supervised Contrastive Learning. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022*, 6916–6926. International Committee on Computational Linguistics.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2019a. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *7th International Conference on Learning Representations, ICLR 2019*. OpenReview.net.

Wang, J.; Lan, C.; Liu, C.; Ouyang, Y.; Qin, T.; Lu, W.; Chen, Y.; Zeng, W.; and Yu, P. S. 2023a. Generalizing to Unseen Domains: A Survey on Domain Generalization. *IEEE Transactions on Knowledge and Data Engineering*, 35(8): 8052–8072.

Wang, M.; Liu, Y.; Yuan, J.; Wang, S.; Wang, Z.; and Wang, W. 2024a. Inter-Class and Inter-Domain Semantic Augmentation for Domain Generalization. *IEEE Transactions on Image Processing*, 33: 1338–1347.

Wang, S.; Liu, X.; Liao, Q.; Wen, Y.; Zhu, E.; and He, K. 2025. Scalable Multi-View Graph Clustering With Cross-View Corresponding Anchor Alignment. *IEEE Transactions on Knowledge and Data Engineering*.

Wang, S.; Liu, X.; Liu, S.; Tu, W.; and Zhu, E. 2024b. Scalable and structural multi-view graph clustering with adaptive anchor fusion. *IEEE Transactions on Image Processing*.

Wang, X.; Hu, P.; Liu, P.; and Peng, D. 2022. Deep Semisupervised Class- and Correlation-Collapsed Cross-View Learning. *IEEE Transactions on Cybernetics*, 52(3): 1588–1601.

Wang, X.; Peng, D.; Hu, P.; Gong, Y.; and Chen, Y. 2023b. Cross-Domain Alignment for Zero-Shot Sketch-Based Image Retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(11): 7024–7035.

Wang, X.; Peng, D.; Hu, P.; and Sang, Y. 2019b. Adversarial correlated autoencoder for unsupervised multi-view representation learning. *Knowledge-Based Systems*, 168: 109–120.

Wang, X.; Peng, D.; Yan, M.; and Hu, P. 2023c. Correspondence-Free Domain Alignment for Unsupervised Cross-Domain Image Retrieval. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023*, 10200–10208. AAAI Press.

Wei, J. W.; and Zou, K. 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. In *EMNLP-IJCNLP 2019*, 6381–6387. Association for Computational Linguistics.

Wen, J.; Liu, Y.; Huang, C.; Liu, C.; Xu, Y.; and Cao, X. 2025. Causal Interventional Prompt Tuning for Few-Shot Out-of-Distribution Generalization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–15.

Wen, J.; Xu, G.; Tang, Z.; Wang, W.; Fei, L.; and Xu, Y. 2024. Graph Regularized and Feature Aware Matrix Factorization for Robust Incomplete Multi-View Clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(5): 3728–3741.

Yang, S.; Du, Y.; Liu, J.; Li, X.; Chen, X.; Gao, H.; Xie, C.; and Lee, Y. 2024. Few-shot multi-domain text intent classification with Dynamic Balance Domain Adaptation Meta-learning. *Expert Systems with Applications*, 255: 124429.

Yao, H.; Yang, X.; Pan, X.; Liu, S.; Koh, P. W.; and Finn, C. 2024. Improving Domain Generalization with Domain Relations. In *The Twelfth International Conference on Learning Representations*.

Yin, Z.; Feng, Y.; Yan, M.; Song, X.; Peng, D.; and Wang, X. 2025. RoDA: Robust Domain Alignment for Cross-Domain Retrieval Against Label Noise. In *AAAI 2025*, 9535–9543. AAAI Press.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022. Conditional Prompt Learning for Vision-Language Models. In *CVPR 2022*, 16795–16804. IEEE.