Target-Adaptive Consistency Enhanced Prompt-Tuning for Multi-Domain Stance Detection

Shaokang Wang[1,2], **Li Pan**[1,2] *

[1] Shanghai Key Laboratory of Integrated Administration Technologies for Information Security

[2] School of Electronics, Information and Electrical Engineering, Shanghai Jiao Tong University

{sklyy6.22, panli}@sjtu.edu.cn

Abstract

Stance detection is a fundamental task in Natural Language Processing (NLP). It is challenging due to diverse expressions and topics related to the targets from multiple domains. Recently, prompt-tuning has been introduced to convert the original task into a cloze-style prediction task, achieving impressive results. Many prompt-tuning-based methods focus on one or two classic scenarios with concrete external knowledge enhancement. However, when facing intricate information in multi-domain stance detection, these methods cannot be adaptive to multi-domain semantics. In this paper, we propose a novel target-adaptive consistency enhanced prompt-tuning method (TCP) for stance detection with multiple domains. TCP incorporates target knowledge and prior knowledge to construct target-adaptive verbalizers for diverse domains and employs pilot experiments distillation to enhance the consistency between verbalizers and model training. Specifically, to capture the knowledge from multiple domains, TCP uses a target-adaptive candidate mining strategy to obtain the domain-related candidates. Then, TCP refines them with prior attributes to ensure prediction consistency. The Pre-trained Language Models (PLMs) in prompt-tuning are with large-scale parameters, while only changing the verbalizer without corresponding tuning has a limited impact on the training process. Target-aware pilot experiments are conducted to enhance the consistency between the verbalizer and training by distilling the target-adaptive knowledge into prompt-tuning. Extensive experiments and ablation studies demonstrate that TCP outperforms the state-of-the-art methods on nine stance detection datasets from multiple domains.

Keywords: Stance Detection, Target-adaptive Verbalizer, Consistency Distillation, Prompt-tuning

1. Introduction

Stance detection is a text classification task that is becoming increasingly complex and diverse, as it involves identifying users' viewpoints on various topics, including support, opposition, or neutrality. This task requires two inputs, a target, and a comment, to identify the author's stance on the target. With the development of social networks, the diversity and complexity of the task have increased dramatically, making it multi-domain. In debates, the target could be topic-based or concept-related, while ideological or event-oriented in Twitter.

Multi-domain stance detection is a challenging task that requires the model to infer stances across multiple domains. Some methods are proposed to incorporate knowledge into models. Arakelyan et al. (2023) explored the basis of per-topic and inter-topic between different domains to select the balanced samples. Schiller et al. (2021) proposed that employing multiple datasets for the training is more feasible and stable. Hu et al. (2022) proposed that the knowledge between diverse targets can be inexhaustible. They incorporated the external commonsense knowledge into the task. However, introducing the auxiliary knowledge into training is inflexible for the emerging domain information. Recently, prompt-tuning has been proposed to transform downstream tasks into a cloze-style format, similar to pre-trained tasks.

However, prompt-tuning often fails to adequately incorporate domain knowledge in stance detection (Jiang et al., 2022). The prompt and verbalizer cannot make the model suitable for multi-domain information. The prompt is used to formalize the input information, and the verbalizer is employed to map the prediction to the label. The prompts are not stable for the various datasets. The original way is to employ several prompts to obtain the average results. In addition, a single mapping verbalizer maps the predictions on limited information. For instance, with stances "Favor" and "Against", we wrap the comment "Nuclear is the best way to generate electricity" and the topic "Nuclear" as {"Nuclear is the best way to generate electricity."? || "[MASK] Nuclear"}. The verbalizer {favor}→Favor indicates that only predicting [MASK] as "favor" is correct. However, the positive words "yes" and "great" and domain-related words "clean" and "effective" are also informative. Therefore, extending the verbalizer with multi-domain information is a feasible way to enhance prompt-tuning. Diverse attributes are employed to construct a verbalizer, such as external knowledge bases (KBS) (Hu et al., 2022), prototypical information (Cui et al., 2022) and relation information (Li et al., 2022). Most of the implementation of prompt-tuning for stance detection remains dealing with some concrete domains. Due to the concrete knowledge bases and attributes, the ex-

* Corresponding author

LREC-COLING 2024, pages 15585–15594
20-25 May, 2024.

tended verbalizer cannot map the domain words to labels when facing diverse domain corpora, e.g., forum debate, fact check, and Wikipedia debate. The knowledge within the verbalizer can be inconsistent with the training process.

In this study, we introduce a novel method named **T**arget-Adaptive **C**onsistency enhanced **P**rompt-tuning (**TCP**) for multi-domain stance detection. Our method aims to construct a target-adaptive verbalizer with knowledge consistency distillation to incorporate domain knowledge into prompt-tuning. The verbalizer is a key module that maps the prediction words to the labels. To enhance the coverage of the verbalizer for multiple domains, TCP employs a target-adaptive candidate selection to capture the domain-related words to the targets. Additionally, the candidates are refined by prior attributes to enhance the prediction consistency. Existing methods (Hu et al., 2022; Li et al., 2022) fail to construct a promising training process with a comprehensive verbalizer. The verbalizer is a projection function in the prompt-tuning, which does not explicitly impact the training process. Therefore, we propose a consistency distillation strategy to incorporate the target knowledge into the module. The consistency distillation employs injection modules for small-scale pilot training to obtain the coarse-grained optimization direction. Then, initializing the prompt-tuning with distilled parameters ensures consistency between the target-adaptive verbalizer and training process to learn the multi-domain knowledge. The proposed method, TCP, constructs a target-adaptive verbalizer and enhances it with consistency distillation to ensure adaptive performance on multi-domain stance detection. Extensive experimental results on nine multi-domain stance detection datasets demonstrate the superior performance of TCP.

In summary, our main contributions are as follows:

- We propose a novel target-adaptive verbalizer with target-adaptive candidate selection to enhance the ability of prompt-tuning to generalize across multiple domains for multi-domain stance detection.

- We propose a consistency distillation strategy based on a target-adaptive verbalizer and pilot experiments to enhance consistency between the verbalizer and the training process in prompt-tuning.

The rest of this paper is organized as follows. In Section 2, we discuss the related works. The definitions and detailed description of the method are provided in Section 3. In Section 4 and Section 5, the results of empirical studies and ablation experiments are reported. Concluding remarks are provided in Section 6.

2. Related Work

2.1. Multi-Domain Stance Detection

With the emergence of social media platforms, stance detection has become increasingly popular. However, due to the vast amount of information on social media, the complexity of this task has increased substantially. The stance detection is with more views (Hanselowski et al., 2018; Team, FNC, 2018), topics (Sobhani et al., 2017), and expressions (Stab et al., 2018; Chen et al., 2019; Walker et al., 2012). The domains in stance detection are also becoming diverse. The source platform can be Twitter (Li et al., 2021; Mohammad et al., 2016), Website (Hanselowski et al., 2019; Team, FNC, 2018), Wikipedia (Bar-Haim et al., 2017) and Forum (Habernal et al., 2018). The form of comment can be debate (Stab et al., 2018), evidence collection (Hanselowski et al., 2019) and comment-reply (Mohammad et al., 2016). Some deep learning methods are employed for multi-domain stance detection by incorporating external knowledge. The basic implementation of external knowledge is Pretrained Language Models (PLMs). The PLMs are fine-tuned in stance detection (Devlin et al., 2019; Liu et al., 2019). In addition, some methods are proposed to use the knowledge from multiple domains (Arakelyan et al., 2023; Lai et al., 2020), datasets (Schiller et al., 2021) and knowledge bases (Hu et al., 2022). Multiple datasets are employed to obtain sufficient information for a certain specific dataset. The knowledge bases, such as ConceptNet (Speer et al., 2017) and WordNet (Pedersen et al., 2004), are employed to obtain assistance for multi-domain stance detection by sentiment or word distributions.

2.2. Prompt-tuning for Stance Detection

PLMs achieve impressive results on NLP tasks, benefiting from prior knowledge. However, PLMs need to change the architecture of models in downstream tasks, which changes the conditions of knowledge implementation. To maintain the task paradigm, prompt-tuning is proposed to transfer all downstream tasks into a cloze-style masked words prediction (Schick and Schütze, 2021; Brown et al., 2020; Jiang et al., 2020; Li and Liang, 2021), which achieves impressive results on diverse tasks. However, simply employing prompt-tuning on stance detection is not the optimal implementation. The mapping function verbalizer is unsuitable for the complex information in stance detection. Many methods try to enhance the generalization of verbalizer, including gradient search (Schick et al., 2020), external knowledge injection (Hu et al., 2022), prototype mining (Cui et al., 2022) and relation information enhancement (Li et al., 2022; Kawintiranon

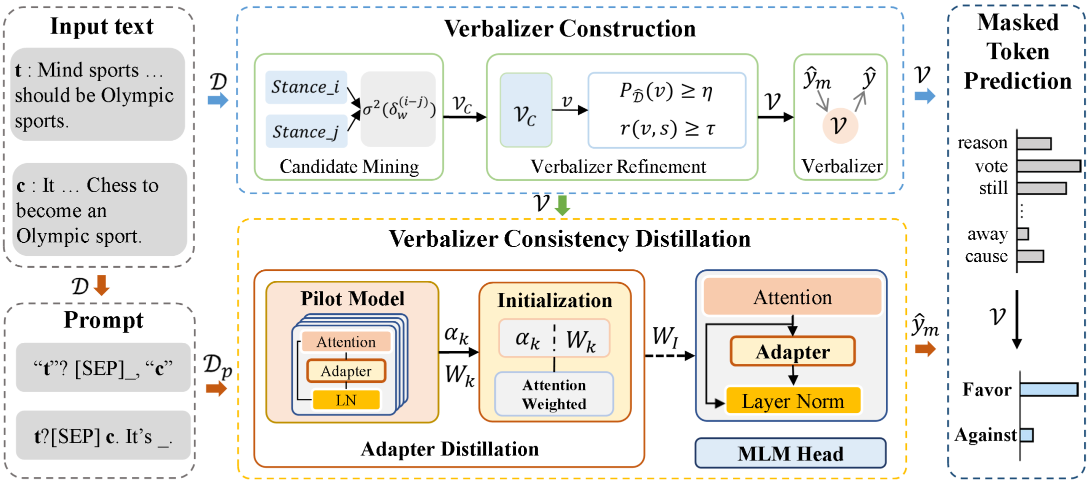


Figure 1: The illustration of TCP. The corpus $\mathcal{D}$ is wrapped by prompt p as $\mathcal{D}_p$. $\hat{y}_m$ is the prediction words of mask token in verbalizer, and $\hat{y}$ is the mapping prediction by verbalizer $\mathcal{V}$.

and Singh, 2021). Furthermore, the prompt is also important in prompt-tuning. Due to the complexity of stance detection, searching for an optimal prompt is infeasible. Therefore, employing several prompts and calculating the average performance is a common way (Jiang et al., 2022). Incorporating the related external knowledge into the verbalizer and training the model with several prompts has become an effective solution. However, when facing stance detection in multiple domains, the existing methods fail to adapt to the domain information. The adaptive knowledge incorporation of prompt-tuning remains to be challenging.

3. Methodology

The proposed framework TCP is shown in Figure 1. TCP consists of Verbalizer Construction and Verbalizer Consistency Distillation. The Verbalizer Construction module constructs a target-adaptive verbalizer by candidate mining and verbalizer refinement joint constraint to capture the target-adaptive knowledge within diverse domains. The Verbalizer Consistency Distillation module conducts pilot experiments with Adapter modules to inject target-adaptive knowledge into the training process. The problem formulation and detailed explanations of TCP are provided in subsequent sections.

3.1. Problem Formulation

Let $\mathcal{D}$ denote the dataset for stance detection, and let x=(t, c) denote the sentence pairs. For stance detection, the goal is to identify the stances of users for topics or discussions, formulated as $Stance(t, c) \in \mathcal{Y}$, where c is a comment, and t is a target, $\mathcal{Y}$ is the set of stances. Given the $x \in \mathcal{D}$ and its ground truth label vector $y \in \{0, 1\}^{|\mathcal{Y}|}$, the predicted probabilities is $\hat{y}$. For multi-domain stance detection, we select different datasets from diverse domains. The multi-domain dataset set is denoted by $\mathcal{D}_M = \{\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_m\}$. The final performance is the average score of all datasets in $\mathcal{D}_M$. To be concise and clear, we discuss the following process for each dataset $\mathcal{D} := \mathcal{D}_i \in \mathcal{D}_M$.

3.2. Verbalizer Construction

Target-**A**daptive **V**erbalizer (TAV) is designed to flexibly incorporate target knowledge within different domain corpus into the prompt-tuning. This section introduces the details of verbalizer candidate mining and refinement.

3.2.1. Candidate Mining

To construct the verbalizer, a reasoning candidate mining process is the key to knowledge incorporation. The main idea of candidate mining is to filter the important words for prediction. The existing measure method for word importance includes TF-IDF, PMI, and Log-Odds ratio. The TF-IDF mainly focuses on the word usage between documents, lacking the measure of importance for the stances and targets. PMI is to measure the correlation between words, which is not flexible for diverse domain information. The Log-Odds ratio is a statistical measure commonly used to assess the significance of a word between different corpora. However, the original Log-Odds ratio can not deal with fine-grained information. To capture the domain-related words of stances, the Log-Odds ratio is improved to measure the significance of words between stances within a corpus. The candidate for the verbalizer is adaptive for the diverse domain datasets. This improved Log-Odds ratio is formatted as:

$$\delta_w^{(i-j)} = log\frac{\beta_w^i + \alpha_w}{n^i + \alpha_0 - \beta_w^i - \alpha_w} \quad (1)$$

$$- log\frac{\beta_w^j + \alpha_w}{n^j + \alpha_0 - \beta_w^j - \alpha_w} \quad (2)$$

where n^i and n^j indicate the number of samples with stance i and j. β_w^i and β_w^j indicate the word count of w in stance i and j, respectively. And α_0 is the stance size in the background dataset, α_w is the word count of w in the background stance. The background dataset is the corpora used for training.

To measure the significance of words, the variance σ^2 of the Log-Odds ratio (Monroe et al., 2008) and the Z-score are as follows:

$$\sigma^2(\delta_w^{(i-j)}) = \frac{1}{\beta_w^i + \alpha_w} + \frac{1}{\beta_w^j + \alpha_w} \quad (3)$$

$$Z = \frac{\delta_w^{(i-j)}}{\sqrt{\sigma^2(\delta_w^{(i-j)})}} \quad (4)$$

A higher Z-score indicates more significance of word w within stance i than j. According to the Z-score, words are chosen for each stance i as the candidate set $\mathcal{V}_C^i \in \mathcal{V}_C$, noting that $\mathcal{V}_C = \{\mathcal{V}_C^1, ..., \mathcal{V}_C^{|\mathcal{Y}|}\}$.

3.2.2. Verbalizer Refinement

The set of verbalizer candidates was initially constructed based on significant words of stances, which is adaptive for diverse domain datasets. These words may contain noise due to an excessive focus on word significance in stance knowledge, neglecting the target information. To obtain a high-quality word set, refining target information from the candidates is crucial. A promising way is aligning with target information by the prior knowledge of PLMs (Hu et al., 2022). The candidates are refined by contextualization and relevant attributes of PLMs for each target in the corpus.

The contextualized priors of words in PLMs reveal target-identifiable knowledge. Given the stance detection task, for each sentence pair in target split corpus $\hat{\mathcal{D}}$, x_p is the wrapped sentence pair. We define $P_{\mathcal{M}}([MASK] = v|x_p)$ as the probability of each word v in the masked position. The expectation of the prediction probability over all sentence pairs in $\hat{\mathcal{D}}$ is formalized as:

$$P_{\hat{\mathcal{D}}}(v) = \mathbb{E}_{x \sim \hat{\mathcal{D}}} P_{\mathcal{M}}([MASK] = v|x_p) \quad (5)$$

Using small-size unlabelled support sets $\tilde{\mathcal{D}}$ can yield a satisfying estimate of the above expectation. Thus, the expectation of $P_{\hat{\mathcal{D}}}(v)$, assuming that the samples $x \in \tilde{\mathcal{D}}$ have a uniform distribution, can be approximated by:

$$P_{\hat{\mathcal{D}}(v)} \approx \frac{1}{|\tilde{\mathcal{D}}|} \sum_{x \in \tilde{\mathcal{D}}} P_{\mathcal{M}}([MASK] = v|x_p) \quad (6)$$

Considering the prior knowledge in PLMs, a threshold η is established to filter out the words from candidates that have probabilities lower than it. Regarding the contextualization attribute, it is important that the qualifying word v must have a high probability of prediction for the masked word.

The Log-Odds ratio measures the significance of words concerning stances, lacking the target relevance of labels in PLMs. Considering different targets, certain words may be more relevant to some labels than others. Therefore, to incorporate the relevance knowledge in PLMs, we calculate the prediction probability of each word in the candidate set on the target support set $\tilde{\mathcal{D}}$, which is denoted as $\mathbf{q}^v$, and $\mathbf{q}^v$'s i-th element is:

$$\mathbf{q}_i^v = P_{\mathcal{M}}([MASK] = v|x_{ip}), x_i \in \tilde{\mathcal{D}} \quad (7)$$

where x_{ip} denotes the sentence pair x_i wrapped with the template p. Similarly, the representation of the stance label s_i of the target is estimated. And the cosine similarity between v and s_i is calculated as:

$$r(v, s_i) = cos(\mathbf{q}_v, \mathbf{q}_{s_i}) \quad (8)$$

A threshold τ is set for $r(v, s_i)$ to construct the verbalizer from $\mathcal{V}_C$. For stance i and target k, verbalizer $\mathcal{V}_i^k \in \mathcal{V}$, k_i is the number of verbalizer words from candidates for stance i and target k:

$$\mathcal{V}_i^k = \{v_1, ..., v_{k_i}\} \quad (9)$$

$$s.t. P_{\hat{\mathcal{D}}}(v_{k_i}) \geq \eta; r(v_{k_i}, s_i) \geq \tau \quad (10)$$

A verbalizer $\mathcal{V}_i^k \in \mathcal{V}$ is constructed with target-adaptive knowledge within multi-domain corpora to map the prediction words.

3.3. Verbalizer Consistency Distillation

Due to the diverse targets and topics, a basic verbalizer may not be sufficient to map the predictions. Instead, a target-adaptive verbalizer is more effective, as it possesses prior knowledge and a strong preference for different targets in a corpus. However, the verbalizer does not affect the training process explicitly, which is not promising that the target-adaptive verbalizer is fully used for prompt-tuning. To enhance the consistency of the verbalizer and training process, we proposed a Verbalizer **C**onsistency **D**istillation (CD) strategy to solidify the knowledge into prompt-tuning. Algorithm 1 provides a pseudo-code outlining the process of verbalizer consistency distillation.

Algorithm 1 Verbalizer Consistency Distillation

```
Input: The pilot experiments models $\mathcal{M} = (\mathcal{M}_1, \mathcal{M}_2, \cdots, \mathcal{M}_K)$; The dataset $\mathcal{D} = (x_1, x_2, \cdots, x_n)$; The target-adaptive verbalizer $\mathcal{V}$.
Output: The initialization parameters $W_I$
for $k$ = 1 to $K$ do
    Select target-related verbalizer $\mathcal{V}^k$ from $\mathcal{V}$
    Sample dataset $\mathcal{D}_k \in Sample(\mathcal{D})$
    Construct samples $X_p^k$ for $\mathcal{D}_k$
    Calculate prediction $\hat{Y}_{dk}$ in Eq.15
    Calculate cross-entropy loss $\mathcal{L}_{ce}$
    Optimize model $\mathcal{M}_k$ with loss $\mathcal{L}_{ce}$
    Obtain parameter $W_k$ and F1-score $s_k$
end for
Constructing $W_I = \sum_k softmax(s_k) W_k$
return Parameter Initialization $W_I$
```

The target-adaptive verbalizer is constructed for the target and stance in the corpus to capture the fine-grained domain knowledge. Training the prompt-tuning for each verbalizer is time-consuming and easily overfitted. To address the problem, we adopt Adapter (Houlsby et al., 2019) for the distillation. The Adapter is a lightweight injected module of prompt-tuning for learning the task-specific information without training the default parameters. Small-scale pilot experiments with Adapter tuning for target-adaptive verbalizer are conducted to obtain the parameters denoted as $W = \{W_1, W_2, \cdots, W_K\}$ and results $\hat{Y}_d = \{\hat{Y}_{d1}, \hat{Y}_{d2}, \cdots, \hat{Y}_{dK}\}$ in Figure 2, K is the number of targets. Subsequently, we employed an F1-score-weighted strategy to incorporate the parameters into W_I for initialization in the prompt-tuning.

Concretely, a PLM encoder contains L layers, and each layer ℓ has M self-attention heads, where a self-attention head (m, ℓ) contains $query$, key and $value$ elements, which can be denoted as: $\mathbf{Q}^{m,\ell}(x^{\ell-1})$, $\mathbf{K}^{m,\ell}(x^{\ell-1})$, and $\mathbf{V}^{m,\ell}(x^{\ell-1})$. $x^{\ell-1}$ is the output of the last layer (the first layer $x^{\ell-1}$ is the output of the embedding layer). After the self-attention layer, aggregating multi-head attention can be formatted as:

$$h_1^\ell = att(\mathbf{Q}^{1,\ell}, \mathbf{K}^{1,\ell}, \mathbf{V}^{1,\ell}, ..., \mathbf{Q}^{m,\ell}, \mathbf{K}^{m,\ell}, \mathbf{V}^{m,\ell}) \tag{11}$$

Then, $Adapter(\cdot)$ is inserted into Transformer module, where $Adapter(\cdot)$ is:

$$h_{a_1}^\ell = W_{i,m_1}^\ell h_1 + b_{i,m_1}^\ell \tag{12}$$

$$h_{a_2}^\ell = RELU(W_{i,m_2}^\ell h_{a_1}^\ell + b_{i,m_2}^\ell) \tag{13}$$

$$h_{a_3}^\ell = W_{i,m_3}^\ell h_{a_2}^\ell + b_{i,m_3}^\ell \tag{14}$$

where h_1^ℓ is the ℓ-th hidden layer with first calculation, a_i is the i-th calculation in one layer. And W_{i,m_1}^ℓ and b_{i,m_1}^ℓ are the parameters of Adapter and bias terms of i-th sub-dataset in ℓ-th layer with m_1-th calculation.

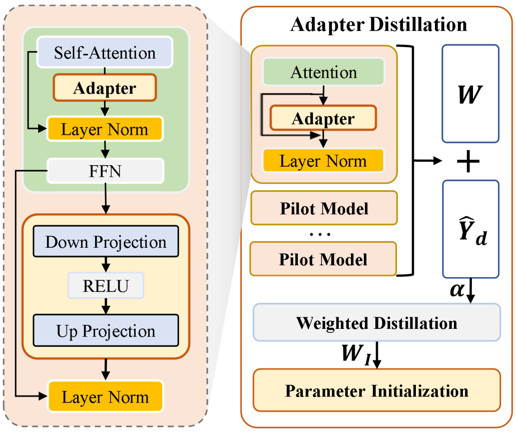


Figure 2: The architecture of the Verbalizer Consistency Distillation. The Up and $Down$ projections are the injected layers in the Adapter. The pilot model is run K times to obtain parameters set W and tuning results set $\hat{Y}_d$.

In the context of stance detection, $\mathcal{D}$ is randomly separated into K parts, where K is the number of targets, i.e., $Sample(\mathcal{D})_p = \{D_1, D_2, ..., D_K\}$. For $D_k \in Sample(\mathcal{D})_p$, pilot experiments are conducted on model $\mathcal{M}_k$, injecting the typical bottleneck Adapter. For prediction $P_{\mathcal{M}_k}([MASK] = v|X_p^k)$, X_p^k denotes the texts wrapped by pattern p of dataset $\mathcal{D}_k$. The results $\hat{Y}_{dk} \in \hat{Y}_d$ for D_k is:

$$\hat{Y}_{dk} = \underset{y \in \mathcal{Y}}{argmax} \sum_{v \in \mathcal{V}_i^k} \tilde{P}_{\mathcal{M}_k}([MASK] = v|X_p^k) \tag{15}$$

Meanwhile, the parameters W_k of the k-th pilot model are recorded for distillation. However, directly incorporating the mean of $W_k \in W$ leads to considerable bias error. This is because every small dataset $D_k \in Sample(\mathcal{D})_p$ only provides a local view of the corpus. Therefore, the $W_k \in W$ is weighted by α, according to the F1-score $s(\hat{Y}_{dk}|X_p^k)$:

$$\alpha_k = \frac{exp(s(\hat{Y}_{dk}|X_p^k))}{\sum_{Sample(\mathcal{D})_p} exp(s(\hat{Y}_d|X_p))} \tag{16}$$

The initialization of Adapter for prompt-tuning can be formatted as follows:

$$W_I = \sum_k \alpha_k W_k \tag{17}$$

The overall process is trained sequentially. The W_I is employed to initialize the Adapter in prompt-tuning. During the inference, we feed the wrapped text with prompt into $P_{\mathcal{M}}^*$ and predict the probability of words by target-adaptive verbalizer $\mathcal{V}$.

4. Experiments

This section introduces the details of datasets, baseline methods, and experiment settings. The main results of the experiments are analyzed in effectiveness analysis.

Datasets	Train	Validation	Test	Total	Stances	Domain
arc	12,382	1,852	3,559	17,792	4	Debate Forum
ibmcs	935	104	1,355	2,394	2	Debate Wikipedia
iac1	4,227	454	924	5,605	3	Debate Politics
argmin	6,845	1,568	2,726	11,139	2	Debate Website
perspectrum	6,978	2,071	2,773	11,822	2	Debate Website
snopes	14,416	1,868	3,154	19,438	2	Fact-Check Website
fnc1	42,476	7,496	25,413	75,385	4	Fake News Website
2016t6	2,497	417	1,249	4,163	3	Hot Topics Twitter
pstance	19,228	2,462	2,374	24,065	2	Politics Twitter

Table 1: The statistics details of datasets.

4.1. Datasets and Evaluation

The nine datasets are from different domains to simulate the complex space of semantics of stance detection. The details of datasets are shown in Table 1. The datasets are: **arc** The Argument Reasoning Corpus (Hanselowski et al., 2018); **ibmcs** The IBM Debater® - Claim Stance Dataset (Bar-Haim et al., 2017); **iac1** The Internet Argument Corpus V1 (Walker et al., 2012); **argmin** The UKP Sentential Argument Mining Corpus(Stab et al., 2018); **perspectrum** The PERSPECTRUM dataset (Chen et al., 2019); **fnc1** The Fake News Challenge dataset (Team, FNC, 2018); **snopes** The Snopes dataset (Hanselowski et al., 2019); **2016t6** The SemEval-2016 Task 6 dataset (Mohammad et al., 2016); **pstance** The Pstance dataset (Li et al., 2021). The macro F1-score aggregates performance across targets by calculating the F1 score independently and then averaging them. The final average results evaluate the model's ability to generalize across all domains, which is useful for assessing overall model performance.

4.2. Baselines

Methods based on Multiple Knowledge Enhancement: MT-DNN (Schiller et al., 2021) transfers knowledge from the GLUE benchmark by multi-dataset learning; **TESTED** (Arakelyan et al., 2023) consists of a topic-guided diversity sampling technique for multi-domain learning.

Methods based on Prompt-tuning: Adapter (Houlsby et al., 2019), neural network layers that are inserted into the Transformer architecture; **PET** (Schick and Schütze, 2021), the regular prompt-tuning method using a single mapping verbalizer.

Methods based on Knowledge Enhanced Verbalizer: KPT (Hu et al., 2022) expands the label word space of the verbalizer using external knowledge bases (KBS); **TAPD** (Jiang et al., 2022) distills the target-aware knowledge of multi prompts to learn the representations for stance detection.

4.3. Experiment Settings

The Hugging-Face PyTorch interface (Wolf et al., 2020) is employed to run experiments on one NVIDIA TITAN RTX with 24GB of memory. Optimizers AdamW (Loshchilov and Hutter, 2019) and Adam (Kingma and Ba, 2015) are chosen to train the models with a batch size of 16 and a maximum sequence length of 256. The training process is run five times, and record the mean results. For PLMs fine-tuning, optimizer Adam is employed with a learning rate 1×10^{-5} to train the models. Optimizer AdamW is chosen for prompt-tuning with learning rates 5×10^{-5} and 7×10^{-4}. The parameters of $Adapter$ are set as the bottleneck one in the paper (Houlsby et al., 2019). The base model of all methods is the roberta-large (Liu et al., 2019) version. We choose three prompts for each dataset and record the average results of all prompts, which are:

$$< t >?||[MASK], < c > \tag{18}$$

$$< t >?|| < c > .I\ stay\ [MASK]\ for\ it. \tag{19}$$

$$< t >?|| < c > .It's\ about\ [MASK]. \tag{20}$$

In verbalizer construction, η and τ vary with the size of the candidate set. η and τ are set to remove three-quarters of the candidates.

4.4. Effectiveness Analysis

The results of all methods are shown in Table 2. The macro F1-score evaluates the performance. In Table 2, TCP achieves the best F1-score results. The comprehensive analysis of TCP is shown in subsequent sections.

Comparison with PLMs enhanced by multiple knowledge: The PLMs-based methods for multi-domain stance detection enhance external knowledge learning through multi-dataset and multi-domain learning. We run the MT-DNN and TESTED for the datasets. TCP performs better than the two methods by 8.23% and 3.44% on average performance, respectively. Except on iac1, TCP achieves the best results. The max margin is up to 15.06% on argmin. On iac1, TESTED outperforms TCP by 7.71%, which is impressive. We analyzed the information in the iac1 and observed that the longest length is 1479, which significantly exceeds the max length. The topic information used in TESTED supplements the truncated information. From other baseline results, the over-length causes the knowledge inconsistency of knowledge injection. Compared to TAV+Adapter, the version of TCP without distillation, TCP obtains an improvement of 4.05% by consistency distillation (CD). The result indicates

Models	argmin	pstance	iac1	ibmcs	2016t6	snopes	arc	fnc1	perspectrum	Average
Train size	6.8k	19.2k	4.2k	0.9k	2.5k	14.4k	12.3k	42.4k	6.9k	
MT-DNN	62.74	77.45	39.97	78.98	67.44	71.05	66.24	78.38	81.65	69.32
TESTED	62.79*	79.55	**56.97***	88.06*	69.91*	78.61*	64.82*	83.17*	83.11*	74.11
Adapter	70.18	77.46	38.29	80.47	64.82	71.08	66.18	79.22	85.78	70.39
PET	72.94	78.23	42.45	85.87	68.70	74.56	68.69	80.53	87.86	73.31
KPT	70.89	78.15	40.45	86.59	69.91	75.79	66.29	81.44	88.25	73.08
TAPD	72.21	80.34	46.15	88.47	70.49	77.62	65.74	82.81	87.29	74.57
TAV+PET	75.94	79.39	45.21	87.28	70.49	76.14	68.66	81.49	89.38	74.89
TAV+Adapter	74.81	78.59	39.88	85.73	68.73	75.71	66.75	80.86	88.77	73.31
TCP	**77.85**	**81.73**	49.26	**90.40**	**73.32**	**79.73**	**70.09**	**84.34**	**91.24**	**77.55**

Table 2: Average macro F1-score for models on nine datasets. The best scores are **bold**, and the second best scores are underlined. The results with * are from (Arakelyan et al., 2023).

Methods			arc	argmin	snopes	2016t6
CM	CR	RR	F1-score			
			TCP-TAV			
×	×	×	67.81(–)	75.49(–)	77.56(–)	71.70(–)
			TCP			
×			67.22(↓)	74.67(↓)	75.81(↓)	70.85(↓)
	×		68.84(↑)	76.78(↑)	77.83(↑)	72.71(↑)
		×	69.30(↑)	76.22(↑)	78.53(↑)	72.58(↑)
			70.09(↑)	**77.85**(↑)	**79.73**(↑)	**73.33**(↑)

Table 3: The results of ablation experiments of target-adaptive verbalizer. "CM," "CR" and "RR" are "Candidate Mining," "Contextualization Refinement," and "Relevance Refinement". The "×" denotes the model is trained without the corresponding operation.

that the consistency distillation effectively injects the knowledge from the verbalizer.

Comparison with prompt-tuning based methods: The comparing methods include vanilla prompt-tuning, PET and Adapter, and external knowledge-enhanced prompt-tuning, KPT and TAPD. From Table 2, TCP achieves an overall surpass on each dataset, indicating that TCP is more adaptive for multiple domain datasets. Compared to KPT and TAPD, the average improvement of TCP is 4.47% and 2.98%, respectively. We observed that KPT performs worse than PET on several datasets, and TAV+Adapter underperforms TAPD. First, KPT extends the verbalizer with knowledge bases by considering the neighbors of labels in the knowledge bases, such as ConceptNet. The labels in multiple domain datasets are similar. Using neighborhood information leads to similar verbalizers and a lack of leveraging the domain knowledge. Second, TADP extends the verbalizer and distills the diverse prompts into training, while TAV+Adapter is only equipped with TAV. The results show that the knowledge implementation of verbalizer is essential for the training. Furthermore, TCP outperforms TAPD on all datasets, utilizing TAV and verbalizer consistency distillation. The target-adaptive verbalizer and consistency distillation enhance prompt-tuning for multiple domain knowledge manipulations.

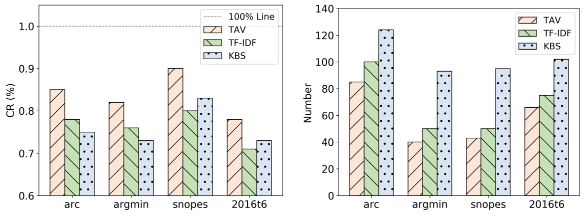
The Effectiveness of TAV and CD: To further elaborate on the effectiveness of TAV and CD, PET and Adapter are wrapped by TAV. The results show that TAV+PET achieves the best second average performance, and TAV+Adapter achieves a competitive average performance with improvements up to 1.58% and 2.92%, respectively. The target-adaptive verbalizer is effective for prompt-tuning. TAV+Adapter is as good as PET. Considering the scale of parameters, the verbalizer distillation is not feasible for PET. The performance of TCP shows the effectiveness of CD, with improvement up to 2.24%, surpassing the results of TAV+PET.

5. Ablation Study

Ablation experiments are conducted to explore the effect of target-adaptive verbalizer and consistency distillation. We chose four datasets for each domain: arc, argmin, snopes, and 2016t6. The performance is evaluated by macro F1-score.

5.1. The Effect of Candidate Mining and Refinement

Table 3 presents the performance of *TCP* under different *TAV* element compositions on four datasets. The method without CM operation employs TF-IDF as the mining method and selects the same size words of datasets as *TCP*. Notably, *TCP-TAV* performs better than *TAV+Adapter* in Table 2, which supports the conclusion that the knowledge distillation of the verbalizer is essential for the training. *TCP-CM* performs worse than *TCP-TAV* indicating that the refinement operation performs worse than the top word selection. We conjecture that top words are more related to domain knowledge, while the tail-distributed words in TF-IDF are suitable for refinement but not for domain information. *TCP-CM* has a margin to *TCP*, indicating that CM is crucial for constructing the verbalizer, and the



(a) Coverage Ratio (b) Number of Words

Figure 3: (a) The coverage ratio (CR) of predicted words of different verbalizers. (b) The number of words of different verbalizers.

Methods	arc	argmin	snopes	2016t6
TF-IDF	65.35	70.91	73.33	68.76
TF-IDF+CD	67.81(↑2.46)	75.49(↑4.58)	77.56(↑4.23)	71.70(↑2.94)
KPT	66.92	70.89	75.79	69.91
KPT+CD	68.81 (↑1.89)	75.46(↑4.57)	77.35 (↑1.56)	70.48(↑0.57)
TAV	66.75	75.94	76.14	70.49
TAV+CD	70.09 (↑3.34)	77.85(↑1.91)	79.73 (↑3.59)	73.32 (↑2.83)

Table 4: The results of different verbalizers with or w/o consistency distillation (CD).

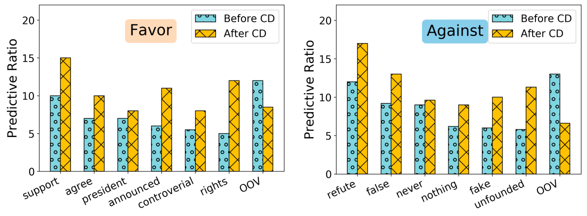
lack of proper candidate selection can result in reverse optimization. The Log-Odds ratio is natural for the domain words filtering due to the information constraints. Compared to the CM, operations CR and RR lead to similar improvements. From the results, we conjecture that CM determines the potential domain adaptive ability of the verbalizer and improves its adaptation by refinement.

5.2. The Candidate Coverage of Target-Adaptive Verbalizer

In Figure 3, we utilize the ratio of coverage of predicted words and the number of words to measure the effectiveness of the verbalizer. TF-IDF and KBS indicate that the verbalizer is constructed by TF-IDF and external knowledge bases. For instance, we record the predicted words in each dataset and calculate the words in the verbalizer. The ratio $CR(\%)$ indicates the model's capacity to identify the domain words and guide the prediction. The TAV achieves the highest CR ratio, and the number of words in TAV is the least. The results indicate that TAV covers more predicted words with a smaller candidate set. TAV, a more concise and precise verbalizer, endows the model with enhanced adaptability to multi-domain datasets.

5.3. Consistency Distillation Analysis

Consistency distillation is employed to enhance the implementation of knowledge within the verbalizer. The single verbalizer extension is not enough for the multi-domain stance detection. The TF-IDF, KPT, and TAV indicate the methods of constructing the verbalizer. The tuning module is the Adapter.



(a) Favor related words (b) Against related words

Figure 4: The times of words prediction in TAV of snopes. The predictive ratio is calculated by *predicted times*/*all words predicted times*.

From the results in Table 4, consistency distillation improves the performance of models. The average improvements of datasets are 3.55%, 2.15%, and 2.92%, respectively. With the help of CD, TF-IDF can perform as well as KPT in multi-domain stance detection. The concrete external knowledge base and similar labels limit the performance of KPT, which supports the assumption in Section 5.1. The results of the three models indicate that consistency distillation is effective for implementing the knowledge of verbalizer into training.

5.4. Visualization of Prompt-tuning Consistency

In Figure 4, we record the times of words when they are predicted correctly. The predictive ratio is used to measure the prediction consistency. If a word has a high ratio, the model predicts it as the label instead of another word. We compared the top six words' predictive ratios in verbalizer with or w/o consistency distillation. "OOV" indicates the predicted words are out-of-verbalizer. The predictions become more focused on the verbalizer words after the process of distillation, which helps to reduce the occurrence of OOV words. The total increase of top words is larger than the reduction, indicating a discernible tendency for predicting the top words instead of tailed words in the verbalizer.

6. Conclusion

TCP is proposed to construct a target-adaptive verbalizer with knowledge consistency distillation to incorporate domain knowledge into prompt-tuning. The existing methods mainly focus on introducing external knowledge without considering multiple domain knowledge. Furthermore, stance detection is multi-domain, and the involved knowledge is too complex to adapt. To overcome these limitations, a target-adaptive verbalizer is proposed to capture the adaptive words toward the domain information. Moreover, we utilize consistency distillation to enhance the implementation of TAV in the training. We conducted comprehensive experiments on nine multi-domain datasets to verify the effectiveness of our model. The target-adaptive verbalizer effectively captures multi-domain information, and consistency distillation is a valuable strategy to inject knowledge into the training process.

7. Acknowledgment

This work is supported by National Natural Science Foundation of China (62172278).

8. Bibliographical References

Erik Arakelyan, Arnav Arora, and Isabelle Augenstein. 2023. Topic-guided sampling for data-efficient multi-domain stance detection. In *Proceedings of ACL*, pages 13448–13464.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of NeurIPS*.

Ganqu Cui, Shengding Hu, Ning Ding, Longtao Huang, and Zhiyuan Liu. 2022. Prototypical verbalizer for prompt-based few-shot tuning. In *Proceedings of ACL*, pages 7014–7024.

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

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. In *Proceedings of NAACL-HLT*, pages 1930–1940.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of ICML*, volume 97, pages 2790–2799.

Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Jingang Wang, Juanzi Li, Wei Wu, and Maosong Sun. 2022. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. In *Proceedings of ACL*, pages 2225–2240.

Yan Jiang, Jinhua Gao, Huawei Shen, and Xueqi Cheng. 2022. Few-shot stance detection via target-aware prompt distillation. In *Proceedings of SIGIR*, pages 837–847.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know. *Trans. Assoc. Comput. Linguistics*, 8:423–438.

Kornraphop Kawintiranon and Lisa Singh. 2021. Knowledge enhanced masked language model for stance detection. In *Proceedings of NAACL-HLT*, pages 4725–4735.

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

Mirko Lai, Alessandra Teresa Cignarella, Delia Irazú Hernández Farías, Cristina Bosco, Viviana Patti, and Paolo Rosso. 2020. Multilingual stance detection in social media political debates. *Computer Speech & Language*, 63:101075.

Qing Li, Yichen Wang, Tao You, and Yantao Lu. 2022. Bioknowprompt: Incorporating imprecise knowledge into prompt-tuning verbalizer with biomedical text for relation extraction. *Inf. Sci.*, 617:346–358.

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

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. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of ICLR*.

Burt L Monroe, Michael P Colaresi, and Kevin M Quinn. 2008. Fightin'words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. Wordnet: : Similarity - measuring the relatedness of concepts. In *Proceedings of AAAI*, pages 1024–1025.

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

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

Benjamin Schiller, Johannes Daxenberger, and Iryna Gurevych. 2021. Stance detection benchmark: How robust is your stance detection? *Künstliche Intell.*, 35(3):329–341.

Parinaz Sobhani, Diana Inkpen, and Xiaodan Zhu. 2017. A dataset for multi-target stance detection. In *Proceedings of EACL*, pages 551–557.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of AAAI*, pages 4444–4451.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of EMNLP*, pages 38–45.

9. Language Resource References

Roy Bar-Haim and Indrajit Bhattacharya and Francesco Dinuzzo and Amrita Saha and Noam Slonim. 2017. *Stance Classification of Context-Dependent Claims*.

Sihao Chen and Daniel Khashabi and Wenpeng Yin and Chris Callison-Burch and Dan Roth. 2019. *Seeing Things from a Different Angle: Discovering Diverse Perspectives about Claims*.

Andreas Hanselowski and Avinesh P. V. S. and Benjamin Schiller and Felix Caspelherr and Debanjan Chaudhuri and Christian M. Meyer and Iryna Gurevych. 2018. *A Retrospective Analysis of the Fake News Challenge Stance-Detection Task*.

Andreas Hanselowski and Christian Stab and Claudia Schulz and Zile Li and Iryna Gurevych. 2019. *A Richly Annotated Corpus for Different Tasks in Automated Fact-Checking*.

Yingjie Li and Tiberiu Sosea and Aditya Sawant and Ajith Jayaraman Nair and Diana Inkpen and Cornelia Caragea. 2021. *P-Stance: A Large Dataset for Stance Detection in Political Domain*.

Saif M. Mohammad and Svetlana Kiritchenko and Parinaz Sobhani and Xiao-Dan Zhu and Colin Cherry. 2016. *SemEval-2016 Task 6: Detecting Stance in Tweets*.

Stab, Christian and Miller, Tristan and Gurevych, Iryna. 2018. *Cross-topic argument mining from heterogeneous sources using attention-based neural networks*.

Team, FNC. 2018. *Exploring how artificial intelligence technologies could be leveraged to combat fake news*.

Marilyn A. Walker and Jean E. Fox Tree and Pranav Anand and Rob Abbott and Joseph King. 2012. *A Corpus for Research on Deliberation and Debate*.