PROMPTDA : Label-guided Data Augmentation for Prompt-based Few Shot Learners

Canyu Chen
Illinois Institute of Technology
cchen151@hawk.iit.edu

Kai Shu
Illinois Institute of Technology
kshu@iit.edu

Abstract

Recent advances in large pre-trained language models (PLMs) lead to impressive gains on natural language understanding (NLU) tasks with task-specific fine-tuning. However, directly fine-tuning PLMs heavily relies on sufficient labeled training instances, which are usually hard to obtain. Prompt-based tuning on PLMs has shown to be powerful for various downstream few-shot tasks. Existing works studying prompt-based tuning for few-shot NLU tasks mainly focus on deriving proper label words with a verbalizer or generating prompt templates to elicit semantics from PLMs. In addition, conventional data augmentation strategies such as synonym substitution are also widely adopted in low-resource scenarios. However, the improvements they bring to prompt-based few-shot learning have been demonstrated to be marginal. Thus, an important research question arises as follows: *how to design effective data augmentation methods for prompt-based few-shot tuning?* To this end, considering the label semantics are essential in prompt-based tuning, we propose a novel *label-guided data augmentation* framework **PROMPTDA**, which exploits the enriched label semantic information for data augmentation. Extensive experiment results on few-shot text classification tasks show that our proposed framework achieves superior performances by effectively leveraging label semantics and data augmentation for natural language understanding.

1 Introduction

Pre-trained language models (PLMs) have shown promising performances in various applications such as text classification (Yang et al., 2019), document summarization (Zhang et al., 2020a), question answering (Mirzaee et al., 2021). The recent advancement of *prompt-based tuning* has shown a significant improvement over normal fine-tuning on various few-shot tasks (Brown et al., 2020). Typically, the prompt-based tuning paradigm transforms a NLU task into a masked language modeling (MLM) problem. For example, in sentiment analysis, an original sentence "`nice movie to watch.`" can be augmented with a *template* "`It is [MASK]`" as the input x. Each class (e.g., POSITIVE) is represented by a *label word* (e.g., `good`) selected by a *verbalizer* from the vocabulary (Schick and Schütze, 2021). The prediction of the class POSITIVE is based on the probability of the `[MASK]` being filled with the token `good`.

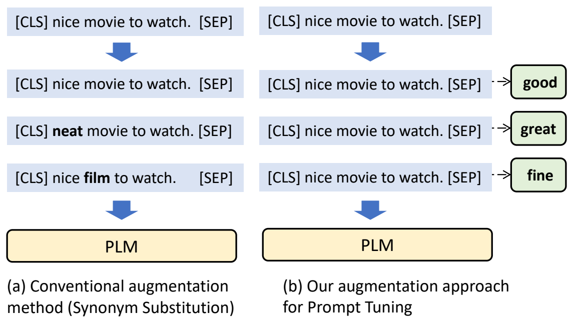


Figure 1: The basic comparison of conventional data augmentation methods and our proposed augmentation framework **PROMPTDA**. {`good, great, fine`} are the label words for prompt tuning. Conventional DA constructs ***instances*** for augmentation. But PROMPTDA conducts ***instance-label pairs*** for augmentation.

In addition, conventional data augmentation (DA) methods such as synonym substitution are also widely applied when the training data is limited (Chen et al., 2021). However, it has been shown in previous works that they can only bring marginal improvements for prompt-based few-shot learning (Zhou et al., 2021). We argue that one of the reasons could be that these DA methods mainly focus on transforming the *instances* while not incorporating the *label semantics*, which have great potential to improve the performances of few-shot tasks (Luo et al., 2021) and are essential for prompt-based few-shot learners (Liu et al., 2021). Therefore, we focus on a new problem of designing augmentation strategies for the prompt tuning paradigm and explore fusing label semantics into

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 562–574
May 2-6, 2023 ©2023 Association for Computational Linguistics

augmentation for prompt-based few-shot learners.

Specifically, different from most prompt-based tuning methods that adopt an *one-to-one verbalizer* (Schick and Schütze, 2021; Gao et al., 2021), we propose to incorporate the rich label semantic information contained in the label words derived from an *one-to-multiple verbalizer* into a new data augmentation paradigm. As shown in Figure 1, compared with previous data augmentation methods that mainly focus on constructing more ***instances***, our method PROMPTDA proposes to construct ***instance-label pairs*** for augmentation, which opens a new dimension for conducting augmentation. For example, with the one-to-multiple verbalizer mapping from the class POSITIVE to a set of label words {`good`, `great`, `fine`}, we aim to generate a set of synthetic data points $\{(x, \texttt{good}), (x, \texttt{great}), (x, \texttt{fine})\}$ based on the original instance x and leverage them to enhance the performances of the prompt-based few-shot learners. Furthermore, extensive experiment results in section § 5.5 also show that our proposed PROMPTDA can be regarded *orthogonal* to the conventional DA methods (e.g., synonym substitution) to some extent. Thus, PROMPTDA can complement with conventional augmentation approaches to further improve the performances.

To this end, we propose a new label-guided data augmentation framework for prompt-based few-shot learners named PROMPTDA, which contains three coherent modules including *Label Augmentation*, *Augmented Prompt-based Tuning*, and *Prediction Transformation*. First, we utilize a PLM to automatically search for an one-to-multiple verbalizer on a specific training set and derive a set of semantically similar tokens for each class as the label words. Second, in the training stage, we construct the instance-label pairs from the original data with regards to each label word for augmentation in prompt tuning. Third, in the inference stage, we utilize the trained language model to predict the label by aggregating the probability scores on the derived label words.

The contributions of this paper are summarized as follows: (1) we study a new problem of designing data augmentation strategies for prompt-based few-shot learners; (2) we propose a novel label-guided data augmentation framework PROMPTDA that exploits the rich label semantic information of one-to-multiple verbalizer for improving prompt tuning; (3) we conduct extensive experiments on real-world few-shot classification tasks and demonstrate the effectiveness of the proposed framework.

2 Related Work

Prompt-based Tuning has attracted increasing attention recently for various natural language processing tasks including text classification (Gao et al., 2021), question answering (Jiang et al., 2020), language generation (Li and Liang, 2021), etc. The prompt-based learning framework has shown promising performances especially in zero shot or few shot classification tasks when limited or no labels are available (Liu et al., 2021). For example, Gao *et al.* propose a prompt-based fine-tuning framework that automatically generates prompt templates and incorporates demonstrations to improve few-shot classification performances (Gao et al., 2021). Shin *et al.* proposes the AutoPrompt method to automatically generate prompts and verbalizers for eliciting the knowledge from language models (Shin et al., 2020). Other works on improving prompt-based model performances also mainly focus on constructing various types of prompt templates and verbalizers (Liu et al., 2021).

Few-shot Text Classification aims to build text classification model when few labeled data is available. Existing works mainly follow the following categories. First, semi-supervised learning where unlabeled data, alongside usually a small amount of labeled data, is used for learning (Mukherjee and Awadallah, 2020; Lee et al., 2021). For example, Subhabrata *et al.* propose to jointly learn from a small set of labeled data and a large amount of unlabeled data with uncertainty using self-training (Mukherjee and Awadallah, 2020). Second, meta-learning frameworks such as metric-based (Sui et al., 2020) and optimization-based approaches (Bansal et al., 2019). Third, weakly supervised learning to derive weak labels (Shu et al., 2020; Meng et al., 2020) in addition to the limited clean labels to improve text classification. Other approaches include transfer learning via learning to adapt transferable information from the source domain to the target domain (Gupta et al., 2020), or leveraging auxiliary tasks to improve the target tasks (Xia et al., 2021; Yin, 2020).

Data Augmentation is to construct synthetic data from an available dataset to enlarge the data size, which can help supervised training with enriched training data (Chen et al., 2021; Guo, 2020;

Shu et al., 2018), or self-supervised learning for constructing samples in pretext tasks (Zhang et al., 2017; Yoon et al., 2020), etc. Data augmentation techniques for natural language generally fall into data space and feature space (Bayer et al., 2021). In the data space, augmentation methods transform the data in character-level, word-level, phrase-level or document-level. In the feature space, representations in the latent space are manipulated by adding noise or interpolation (Schwartz et al., 2018; Verma et al., 2019). However, conventional data augmentation methods bring marginal improvements under prompt tuning paradigm (Zhou et al., 2021). It is under exploring about how to design effective data augmentation methods for prompt-based few-shot scenarios. Therefore, we propose a novel label-guided data augmentation mechanism in prompt-based tuning for few shot tasks.

3 Problem Definition

The goal of few-shot classification task is to learn a classifier to predict the label of unseen instances with limited labeled samples during the training. Following the widely-used few-shot setting (Gao et al., 2021; Liu et al., 2021), we assume that a large pre-trained language model (e.g., BERT) $\mathcal{M}$ can be utilized to fine-tune on a downstream task with the dataset $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\}$, where $\mathcal{X}$ denotes the instances and $\mathcal{Y}$ indicates the corresponding labels. For each task, the number of training instances for each class is K, which is usually small (e.g., 8 or 16). The goal is to design a prompt learning strategy that generalizes well on unseen samples in the test set $\mathcal{D}_{\text{test}}$ with few labeled training data in $\mathcal{D}_{\text{train}}$. To ensure a fair parameter setting, we assume that a validation set $\mathcal{D}_{val}$ is available, and $|\mathcal{D}_{\text{val}}| = |\mathcal{D}_{\text{train}}|$. The test set $\mathcal{D}_{\text{test}}$ is the same as the full-data training setting.

4 Label-guided Data Augmentation for Prompt-based Tuning

In this section, we detail the proposed framework PromptDA, which is illustrated in Figure 2. It mainly consists of three modules: (1) a Label Augmentation module to derive multiple label words for each class to enrich the label space; (2) an Augmented Prompt-based Tuning module for augmenting the training data guided by label words; (3) a Prediction Transformation module to transform the prediction from the label words to original classes.

4.1 Label Augmentation

Due to the limited available labels in few-shot learning, recent works are generating label words to help prediction (Schick and Schütze, 2021; Gao et al., 2021). The goal is to extend the label space by incorporating the rich semantics in the vocabulary. While existing works mainly focus on selecting one label word for each class manually or automatically in prompt-tuning, the resultant label words often have a large variance and the semantics in other candidate label words are ignored. Therefore, we explore automatically searching for multiple label words for each class to better enrich the label space. Let $\mathcal{F} : \mathcal{Y} \rightarrow \mathcal{V}_{\mathcal{Y}}$ denote the one-to-multiple verbalizer that maps each label category $y \in \mathcal{Y}$ to a set of label words $\mathcal{V}_y = \{v_y^1, v_y^2, ..., v_y^{k_y}\} \subset \mathcal{V}$, where $k_y = |\mathcal{V}_y|$ denotes the number of selected label words for each class.

Firstly, we aim to search for a candidate set of label word $\tilde{\mathcal{V}}_y \subset \mathcal{V}$ that is semantically similar to each class $y \in \mathcal{Y}$. Let $\mathcal{D}_{\text{train}}^y$ denote the subset of training data with the class y. $\mathcal{T}(x)$ denotes the input x with a fixed template $\mathcal{T}$. `Po([mask])` denotes the position of `[mask]` in the input x. We propose to select the Top- m label words from vocabulary as $\tilde{\mathcal{V}}_y$ based on the conditional likelihood over $\mathcal{D}_{\text{train}}^y$ for each class y:

$$\tilde{\mathcal{V}}_y = \underset{v \in \mathcal{V}}{\text{Top-}\,m} \left\{ \sum_{(x,y) \in \mathcal{D}_{\text{train}}^y} \Pr(v, \mathcal{T}(x)) \right\} \tag{1}$$

where $\Pr(v, \mathcal{T}(x))$ denotes the corresponding probability score of each token in the vocabulary filling in `Po([mask])` in PLM inference as:

$$\Pr(v, \mathcal{T}(x)) = \Pr(\texttt{Po([mask])} = v \mid \mathcal{T}(x)) \tag{2}$$

Secondly, we construct a verbalizer candidate set F for the whole dataset. It is a combinatorial problem to select k_y label words from $\tilde{\mathcal{V}}_y$ to construct $\mathcal{V}_y$ for each class y. The number of possible candidates of $\mathcal{V}_y$ is $\binom{|\tilde{\mathcal{V}}_y|}{k_y}$. Then the element number of the verbalizer candidate set F is $|F| = \binom{|\tilde{\mathcal{V}}_y|}{k_y}^{|\mathcal{Y}|}$. We utilize each one-to-multiple verbalizer candidate in F to infer and calculate the prediction accuracy on $\mathcal{D}_{\text{train}}$ via the same *prediction transformation* method in section § 4.3. Then we select the Top- n candidates from F based on the prediction accuracy. If there exist multiple candidates with the same accuracy score, we randomly select one as the

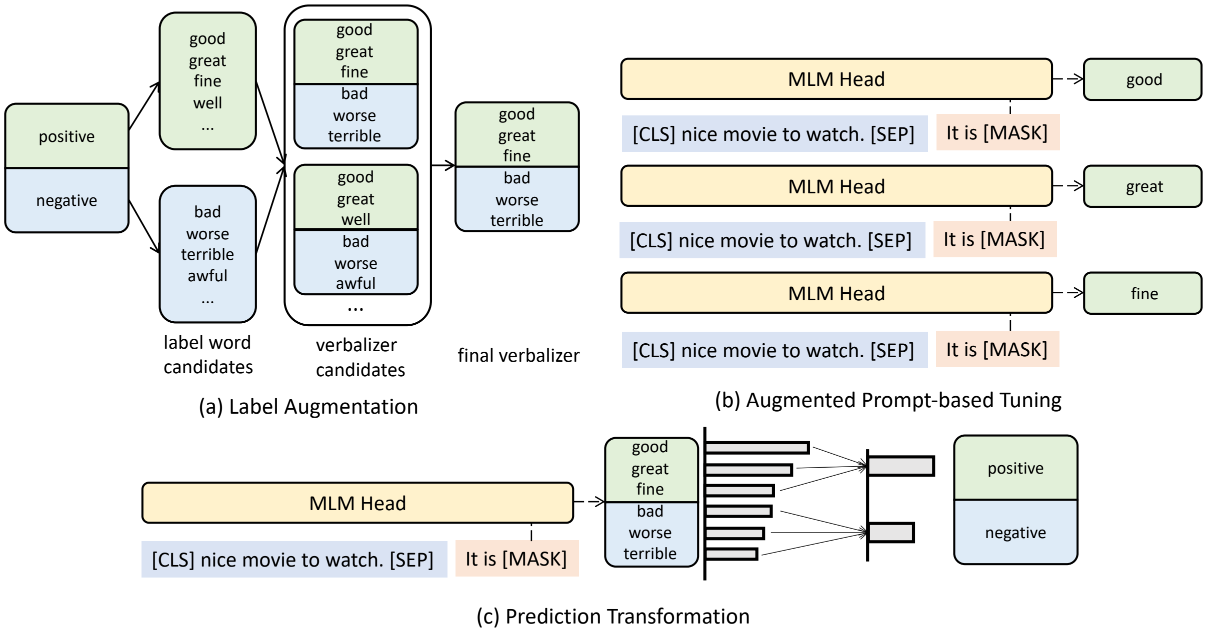


Figure 2: The proposed **PROMPTDA** for few-shot learning (with sentiment classification task as an example): (a) **Label Augmentation**: deriving multiple label words for each class to enrich the label semantic space; (b) **Augmented Prompt-based Tuning**: training with the augmented instance-label pairs via masked language modeling; (c) **Prediction Transformation**: aggregating the probability scores on the derived label words for the final prediction.

final one-to-multiple verbalizer. Otherwise, we select the verbalizer candidate with highest accuracy score. Note that m and n are both hyperparameters picked by pilot study on the specific datasets.

4.2 Augmented Prompt-based Tuning

To enrich the training data for the few-shot text classification task, it is natural to utilize data augmentation methods such as token-level or sentence-level augmentation for fine-tuning (Chen et al., 2021). Most of the existing data augmentation methods focus on enlarging training data conditioned on the original label space. Orthogonal to previous augmentation methods, our method incorporates label semantic information into prompt-tuning via augmenting sample-label pairs rather than only augmenting samples. For $(x, y) \in \mathcal{D}_{\text{train}}$, we have obtained the corresponding label word set $\mathcal{V}_y = \{v_y^1, v_y^2, ..., v_y^{k_y}\}$. Then we can include $\{(x, v_y^1), (x, v_y^2), ..., (x, v_y^{k_y})\}$ for augmentation. Let $\tilde{\mathcal{D}}_{\text{train}}$ denote the augmented dataset. The resultant dataset can be denoted as follows:

$$\tilde{\mathcal{D}}_{\text{train}} = \cup_{(x,y)\in\mathcal{D}_{\text{train}}}\{(x, v_y^1), (x, v_y^2), ..., (x, v_y^{k_y})\} \tag{3}$$

In the training process, we follow the MLM training paradigm and minimize the negative log-likelihood on the whole augmented training set $\tilde{\mathcal{D}}_{\text{train}}$. The optimization objective is:

$$\mathcal{L} = \sum_{(x,v)\in\tilde{\mathcal{D}}_{\text{train}}} - \log \Pr(v \mid x) \tag{4}$$

For $(x, v) \in \tilde{\mathcal{D}}_{\text{train}}$, the conditional probability of filling the position of `[mask]` with v is:

$$\begin{aligned}\Pr(v \mid x) &= \Pr(\texttt{Po([mask])} = v \mid x) \\ &= \frac{\exp\left(\mathbf{w}_v \cdot \mathbf{h}_{\texttt{[MASK]}}\right)}{\sum_{v'\in\mathcal{V}} \exp\left(\mathbf{w}_{v'} \cdot \mathbf{h}_{\texttt{[MASK]}}\right)}\end{aligned} \tag{5}$$

where $\mathbf{w}_v$ denotes the pre-softmax output vector for each token v in the vocabulary, and $\mathbf{h}_{\texttt{[MASK]}}$ denotes the corresponding hidden state of the `[MASK]` position. Note that we completely reuse the PLM and do not introduce new parameters in the training process, which is important for prompt-based tuning to be effective in few-shot scenarios.

4.3 Prediction Transformation

We have demonstrated the process of training the MLM classifier head with the augmented data in the prompt-based tuning paradigm. Next, we describe how to perform the inference for the target class. Let h denote the function that transforms the probability scores on the label word set $\mathcal{V}_y = \{v_y^1, v_y^2, ..., v_y^{k_y}\}$ into the probability score of each class y. Since the label word with the highest probability score in set $\mathcal{V}_y$ can represent the class y, we use $h = max()$ to calculate the final probability score of each class. Then the probability score of each class y can be calculated as:

$$\Pr(y \mid x) = h(\mathrm{P}(v_y^1, x), \mathrm{P}(v_y^2, x), ..., \mathrm{P}(v_y^{k_y}, x)) \quad (6)$$

where for (x, v_y^i) that satisfies $(x, v_y^i) \in \tilde{\mathcal{D}}_{\text{train}}$ and $v_y^i \in \mathcal{V}_y, (i = 1, 2, ..., k_y)$, $\mathrm{P}(v_y^i, x)$ is denoted as the conditional probability of filling the position of `[mask]` with v_y^i:

$$\mathrm{P}(v_y^i, x) = \Pr(\texttt{Po([mask])} = v_y^i \mid x) \quad (7)$$

After we obtain the probability score over each class, the final predicted class $\hat{y}$ is calculated as:

$$\hat{y} = \operatorname{argmax}_{y \in \mathcal{Y}} \Pr(y \mid x) \quad (8)$$

5 Experiments

In this section, we present the experiments to evaluate the effectiveness of the proposed PROMPTDA. Specifically, we aim to answer the following research questions:

- **RQ1** Can PROMPTDA improve the performance of few-shot prompt-based tuning?
- **RQ2** Can the proposed Label Augmentation strategy help the target label prediction?
- **RQ3** Can the PROMPTDA make the prompt-based tuning method more stable?

5.1 Experimental Settings

Datasets. We evaluate the proposed framework on few shot text classification datasets from the widely-used NLU benchmark GLUE (Wang et al., 2018) including SST-2 (Socher et al., 2013), CoLA (Warstadt et al., 2019) and other common datasets including MR (Pang and Lee, 2005), CR (Hu and Liu, 2004), Subj (Pang and Lee, 2004), MPQA (Wiebe et al., 2005), SST-5 (Socher et al., 2013). These datasets covers different tasks such as sentiment analysis, topic classification and opinion classification from various domains including movie reviews, news pieces, etc. The statistics of the datasets are shown in Table 4 in Appendix.

Baselines. We compare the proposed approach with various representative methods including **Majority**, **Fine-Tuning**, **GPT-3** (Brown et al., 2020), **EFL** (Wang et al., 2021), **LM-BFF** (Gao et al., 2021) and **Prompt Tuning**. More details are described in the Appendix A.3.

Evaluation setting. Evaluation is critical in few-shot scenarios because small changes of the training set can result in a large variance in the performance of the test set. Following the few-shot setting in (Perez et al., 2021), (Zhang et al., 2020b), (Gu et al., 2021) and (Gao et al., 2021), we randomly select K-shot samples from original dataset for each class to construct the training set $\mathcal{D}_{\text{train}}$ and select another K-shot samples to construct the development set $\mathcal{D}_{\text{val}}$. For enhancing the stability of evaluation, we utilize the whole test set of original dataset as out test set $\mathcal{D}_{\text{test}}$ and change the random seed of sampling $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{val}}$ for five times. We select RoBERTa-large as our backbone model to make fair comparison with baseline LM-BFF.

5.2 Experimental Results

In this section, we present our main results, and address the aforementioned research questions pertaining to our PROMPTDA approach.

In addition to comparing with baselines such as Majority, normal fine tuning and prompt-based method GPT-3, EFL, LM-BFF, we conduct more experiments to verify the effectiveness of our proposed method PROMPTDA as a plug-in module. Because different template choices can result in a large variance of performance (Gao et al., 2021), we design two groups of experiments, namely template-free and template-augmented, to investigate whether or not our method can improve over standard prompt-based tuning method regardless of template design. For the template-augmented group of experiments, we manually choose "`It is [MASK]`" as the template, following (Wang et al., 2021). For the template-free group of experiments, we only append "`[MASK]`" in the input. We report the results of PROMPTDA in Table 1 when the size of data augmentation is $\times 3$ (i.e., $k_y = 3$). We also consider two scenarios where the label words are derived manually or with our automatic label augmentation mechanism. We choose 8 samples ($K = 8$) per class as the few-shot setting of our main experiments. For fair comparison, we choose the same random seed of training set sampling as LM-BFF. We train for 10 epochs on each dataset following (Wang et al., 2021). We report the average performance and standard variance of our results over five runs of sampling on each dataset. The main results can be seen in Table 1.

Performance analysis We analyze the performance from three perspectives to answer the afore-

Method	SST-2 (Acc)	MR (Acc)	CR (Acc)	Subj (Acc)	CoLA (Acc)	MPQA (Acc)	SST-5 (Acc)
Majority (full)	50.9	50.0	50.0	50.0	69.1	50.0	23.1
Fine-Tuning (full)	95.0	90.8	89.4	97.0	86.2	89.4	58.7
Fine-Tuning	60.5 (3.1)	60.3 (7.5)	61.9 (5.1)	78.3 (8.2)	51.1 (8.4)	59.0 (3.4)	31.5 (7.5)
GPT-3 (Brown et al., 2020)	82.9 (3.4)	81.2 (2.5)	86.8 (1.5)	53.2 (1.5)	52.1 (6.2)	62.9 (3.5)	31.5 (4.3)
EFL (Wang et al., 2021)	67.5 (8.5)	69.8 (7.5)	75.3 (4.8)	78.9 (7.8)	54.3 (8.9)	68.4 (5.7)	35.2 (6.3)
LM-BFF (Gao et al., 2021)	89.1 (4.1)	83.6 (3.4)	87.8 (4.3)	81.6 (6.1)	53.5 (4.5)	73.9 (8.9)	41.2 (3.1)
Prompt Tuning‡	85.5 (5.2)	83.0 (3.7)	86.5 (3.0)	81.8 (5.6)	50.5 (10.3)	71.5 (9.8)	37.5 (5.5)
PT + PROMPTDA(m.)‡	87.3 (4.4)	82.5 (1.4)	88.1 (2.7)	81.3 (4.9)	51.2 (7.5)	72.9 (9.1)	39.4 (4.3)
PT + PROMPTDA(au.)‡	87.6 (4.1)	83.1 (3.1)	87.8 (1.2)	83.4 (2.5)	52.8 (8.1)	74.5 (7.8)	41.8 (3.9)
Prompt Tuning†	85.8 (5.8)	79.3 (8.2)	86.1 (8.0)	81.2 (5.7)	52.7 (6.6)	75.1 (13.7)	38.4 (4.7)
PT + PROMPTDA(m.)†	88.9 (3.9)	**83.8 (1.9)**	84.9 (5.7)	82.4 (9.9)	51.3 (15.5)	78.1 (8.9)	42.7 (7.1)
PT + PROMPTDA(au.)†	**89.5 (2.9)**	83.7 (2.6)	**88.3 (4.1)**	**86.8 (3.1)**	**55.9 (7.1)**	**78.4 (9.2)**	**43.3 (1.6)**

Table 1: The main results using RoBERTa-large on representative NLU tasks. All the results are evaluated on full test sets and averaged over 5 runs. $K = 8$: 8 samples per class for all the experiments; †: template augmented; ‡: template-free; (m.): manual label augmentation; (au.): automatic label augmentation; PT: Prompt Tuning.

mentioned research questions.

To answer **RQ1**, we compare the proposed method with existing baselines. First, in general, we can observe that the standard prompt-based tuning method with PROMPTDA consistently perform better than or is comparable with baselines such as GPT-3, EFL, LM-BFF and normal fine tuning (results of "PT + PROMPTDA(au.)†" in Table 1). Compared with LM-BFF, standard prompt-based tuning with PROMPTDA performs better on all the datasets. For example, our method achieves a 6% gain over LM-BFF on Subj and MPQA datasets. Compared with normal fine tuning, our method achieves superior performance by a large margin. For example, our method obtains a 47.9% improvement over normal fine tuning on SST-2 dataset.

Second, we can see that PROMPTDA can improve over standard prompt-based tuning method regardless of template design (results of "PT + PROMPTDA(au.)†" and "PT + PROMPTDA(au.)‡" in Table 1). Compared with standard prompt tuning, PROMPTDA can achieve a better performance over the seven datasets regardless of being template-free or template-augmented, which suggests that PROMPTDA has no relation with template design and can be used as a plug-in module for improving performance of prompt tuning.

Third, PROMPTDA generally improves over standard prompt-based tuning method regardless of automatic label word selection or manual label word selection (results of "PT + PROMPTDA(au.)†" and "PT + PROMPTDA(m.)†" in Table 1). Compared with standard prompt tuning, prompt tuning with automatic label word selection achieves improvements over all the datasets. For prompt tuning with manual label word selection, it also has performance gains over SST-2, MR, Subj, MPQA and SST-5 datasets.

To answer **RQ2**, we perform an ablation study of PROMPTDA, and compare the results of "PT + PROMPTDA(au.)†" and "PT + PROMPTDA(m.)†" in Table 1. We can see that regardless of template design, our proposed automatically searched label words generally perform better than manually searched label words. For example, "PT + PROMPTDA(au.)†" achieves a 5.3% improvement over "PT + PROMPTDA(m.)†" on Subj dataset.

We analyze the reason from three perspectives. First, we hypothesize that human bias may hinder selecting optimal label words and our proposed automatic method relies on language model itself and can minimize human bias. Second, it may be easier for human to select similar words as label words for sentiment-related datasets with the label name "`positive, negative`", but it is hard to select semantically similar words as label words for tasks in other domains. For example, it is hard to manually identify semantically similar words as label words for Subj dataset with the label name "`subjective, objective`", which illustrates the necessity of our proposed automatic method for searching label words. Third, our proposed Label Augmentation method can search for different label words on different training data, but it is hard for the manual label word selection method to adapt to different specific datasets.

To answer **RQ3**, we analyze the stability of performances of PROMPTDA. In general, we observe that PROMPTDA reduces the variance of prompt-tuning. (Standard variance of "PT +

SST-2	label name	`positive	negative`
	label words (m.)	`positive, great, good	negative terrible bad`
	label words (au.)	`wonderful brilliant fantastic	terrible done disappointing`
Subj	label name	`objective	subjective`
	label words (m.)	`good neutral fair	bad emotional personal`
	label words (au.)	`disturbing terrifying key	bad not nonsense`
SST-5	label name	`very positive	positive	neutral	negative	very negative`
	label words (m.)	`great perfect excellent	good, pretty, wonderful	neutral normal fine	bad worse not	terrible awful ridiculous`
	label words (au.)	`great brilliant fantastic	extraordinary remarkable fascinating	enough terrible funny	awful bad worse	boring done unnecessary`

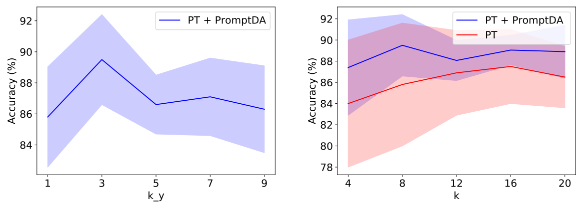
Table 2: An illustration of the label words searched automatically or manually on SST-2, Subj and SST-5 datasets.

PROMPTDA(au.)†" in Table 1). The uncertainty of prompt-based tuning methods mainly comes from different distribution of small training set, different designs of the template and various selections of label words for each class. Compared with LM-BFF and normal fine tuning methods, our method generally reduces the variance of prediction. For example, the standard variance of prediction over five runs for "PT + PROMPTDA(au.)†" has decreased around 48.4% on Subj compared to LM-BFF and has decreased 78.7% on SST-5 compared with normal fine tuning. Compared with standard prompt-based tuning method, PROMPTDA can improve the stability of tuning on most of the datasets.

5.3 Analysis of Label Word Selection

Without loss of generosity, we take the dataset SST-2, Subj and SST-5 for example to analyze the quality of Label Augmentation (the label word results are shown in Table 2 and the complete label word results over five runs on SST-2, CR, MR, Subj, CoLA, MPQA, SST-5 datasets are shown in Appendix Table 6). *The goal of label augmentation is to find semantically similar words to enrich the label space.* With regards to the manual way, we find the synonyms of label name from dictionary as the label words and ensure these words are in the vocabulary. And we select the same label words for different seeds. With regards to our proposed automatic method, we only rely on the training set and language model (e.g., RoBERTa-large) to find the semantically similar words from vocabulary and do not rely on label name itself.

The Table 2 shows the label words automatically or manually searched on datasets SST-2, Subj and SST-5 respectively. For sentiment related datasets such as SST-2 with the label name {`positive/negative`}, the label words automatically searched are literally similar to the manually selected label words, which probably means



(a) # size of augmentation (b) # samples per class

Figure 3: The impact analysis of the size of label words and training samples per class on SST-2 dataset.

the way language models (e.g., RoBERTa-large) reasons about what are similar words is close to the human way in sentiment domain. Nonetheless, for other datasets such as Subj with the label name {`objective/subjective`}, it is interesting to observe that the label words automatically searched are not literally similar to label name or manually selected label words, which may infer that the way language models (e.g., RoBERTa-large) reason about what are similar words is different from the human way in other domains. We argue that *how to define word similarity in label semantic space* needs more research in the future. For dataset such as SST-5, we can see that it is much harder to select appropriate label words when the number of classes is larger, which also verifies the importance of automatic label word selection.

5.4 Assessment of Data Augmentation

We analyze data augmentation from two perspectives including the size of data augmentation and the size of training set.

The size of data augmentation We choose to study the effect of the size of PROMPTDA on template-augmented prompt-based tuning on SST-2 dataset. The results over five runs for 10 epochs are presented in Figure 2 (a). We can observe that PROMPTDA can generally improve over prompt-based tuning regardless of the size of augmentation. However, larger augmentation may result in more

Method	SST-2 (Acc)	MR (Acc)	CR (Acc)	Subj (Acc)	SST-5 (Acc)
PT	85.8 (5.8)	79.3 (8.2)	86.1 (8.0)	81.2 (5.7)	38.4 (4.7)
PT with Conventional DA	89.2 (1.3)	80.3 (3.1)	86.5 (4.5)	82.3 (8.0)	39.1 (4.5)
PT with PROMPTDA	89.5 (2.9)	83.7 (2.6)	88.3 (4.1)	86.8 (3.1)	43.3 (1.6)
PT with PROMPTDA & Conventional DA	**89.7 (1.6)**	**84.8 (1.5)**	**89.2 (1.3)**	**87.0 (3.1)**	**44.7 (1.1)**

Table 3: The main results of evaluating Prompt Tuning (PT) with PROMPTDA and conventional DA method on NLU tasks. All the results are evaluated on full dev sets and averaged across 5 different training sets. $K = 8$: 8 samples per class for the experiments. Conventional DA refers to *synonym substitution*.

unstable final prediction. We analyze the reason from two perspectives. First, larger data augmentation may contain more label noise. Since we utilize an one-to-multiple verbalizer to guide data augmentation, the size of data augmentation is equal to the number of label words per class, which may cause more noisy label words. Unsuitable label word selections may worsen the performance and increase the variance of final prediction. Second, more label words per class may cause the model harder to converge on small training sets. When training for the same epochs, prompt tuning with more label words per class may perform more unstable.

The size of the training set We study the effect of the size of training set on template-augmented prompt-based tuning with and without PROMPTDA. The size of data augmentation is $\times 3$. The results over five runs for 10 epochs are presented in Figure 2 (b). We have several observations from the results. First, our method PROMPTDA consistently improves over standard prompt-based tuning regardless of the size of training sets. Second, our proposed method generally decreases the variance of prompt-based tuning. Third, the improvement space of PROMPTDA over prompt-based tuning decreases as the number of samples per class increases.

5.5 Combination with Conventional DA

Although conventional data augmentation methods are still effective when training data is limited (Chen et al., 2021), previous works verified that they can bring marginal improvement for the prompt tuning paradigm (Zhou et al., 2021). It is worth exploring whether or not PROMPTDA can complement with conventional DA for further enhancing the performance of prompt tuning.

We follow the same setting as the main experiments and test conventional DA, PROMPTDA and the combination on standard prompt-based tuning paradigm with template. With regards to conventional DA, we select *synonym substitution* method from nlpaug toolkit (Ma, 2019) and enlarge the training set by $\times 2$. With regards to our proposed PROMPTDA, we enlarge the training set by $\times 3$. The experiment results over five different sampling seeds for 10 epochs are shown in Table 3.

We can observe that the combination of PROMPTDA and Conventional DA method consistently outperforms only using PROMPTDA or Conventional DA method. Conventional DA methods mostly focus on exploiting the semantic information of the instance itself. Our method proposes to utilize label semantic information to guide data augmentation and does not change instances. PROMPTDA conducts the augmentation from a different perspective compared with conventional augmentation methods. Therefore, our proposed method PROMPTDA can be regarded *orthogonal* to conventional DA methods to some extent and complement with each other.

6 Conclusion and Future Work

In this paper, we study a new problem of data augmentation in prompt-based tuning for few shot learners. To leverage the label semantic information, we propose a novel label-guided data augmentation approach PROMPTDA, which can derive multiple label words and exploit the rich semantic information of the label words. We conduct extensive experiments on various datasets and demonstrate the effectiveness of PROMPTDA for few-shot learning. We also conduct detailed analysis on the effects of manual/automatic label augmentation, the size of augmentation, the size of label words, and combination with conventional DA.

There are several interesting directions for future work. First, we will extend PROMPTDA to multi-label few shot tasks and leverage multi-aspect label space. Second, we will explore prompt-based data augmentation for token-level tasks such as few-shot name entity recognition (NER). Third, we will explore prompt-based tuning to enhance the interpretability capacity for various NLP tasks.

Limitations

Our work is the first step for designing data augmentation strategies for prompt tuning paradigm. In this work, we focus on the natural language understanding (NLU) tasks. The prompt tuning paradigm is applied in various tasks including language generation, question answering, dialog system, etc. Designing augmentation strategies for prompt-based few-shot learners in more applications is under exploration.

Ethics Statement

This paper focuses on the task of few-shot natural language understanding and conducts experiments on open datasets. The implementation details are described in Appendix for reproduction.

Acknowledgements

This work is supported in part by NSF SaTC-2241068, and a Cisco Research Award

References

Trapit Bansal, Rishikesh Jha, and Andrew McCallum. 2019. Learning to few-shot learn across diverse natural language classification tasks. *arXiv preprint arXiv:1911.03863*.

Markus Bayer, Marc-André Kaufhold, and Christian Reuter. 2021. A survey on data augmentation for text classification. *arXiv preprint arXiv:2107.03158*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Jiaao Chen, Derek Tam, Colin Raffel, Mohit Bansal, and Diyi Yang. 2021. An empirical survey of data augmentation for limited data learning in nlp. *arXiv preprint arXiv:2106.07499*.

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

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2021. Ppt: Pre-trained prompt tuning for few-shot learning. *arXiv preprint arXiv:2109.04332*.

Hongyu Guo. 2020. Nonlinear mixup: Out-of-manifold data augmentation for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4044–4051.

Aakriti Gupta, Kapil Thadani, and Neil O'Hare. 2020. Effective few-shot classification with transfer learning. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1061–1066.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Ju-Hyoung Lee, Sang-Ki Ko, and Yo-Sub Han. 2021. Salnet: Semi-supervised few-shot text classification with attention-based lexicon construction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13189–13197.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Qiaoyang Luo, Lingqiao Liu, Yuhao Lin, and Wei Zhang. 2021. Don't miss the labels: Label-semantic augmented meta-learner for few-shot text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2773–2782.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang, and Jiawei Han. 2020. Text classification using label names only: A language model self-training approach. *arXiv preprint arXiv:2010.07245*.

Roshanak Mirzaee, Hossein Rajaby Faghihi, Qiang Ning, and Parisa Kordjmashidi. 2021. Spartqa:: A textual question answering benchmark for spatial reasoning. *arXiv preprint arXiv:2104.05832*.

Subhabrata Mukherjee and Ahmed Awadallah. 2020. Uncertainty-aware self-training for few-shot text classification. *Advances in Neural Information Processing Systems*, 33.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *arXiv preprint cs/0409058*.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *arXiv preprint cs/0506075*.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *arXiv preprint arXiv:2105.11447*.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269.

Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Rogerio Feris, Abhishek Kumar, Raja Giryes, and Alex M Bronstein. 2018. Delta-encoder: an effective sample synthesis method for few-shot object recognition. *arXiv preprint arXiv:1806.04734*.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Eliciting knowledge from language models using automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235.

Kai Shu, Subhabrata Mukherjee, Guoqing Zheng, Ahmed Hassan Awadallah, Milad Shokouhi, and Susan Dumais. 2020. Learning with weak supervision for email intent detection. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1051–1060.

Kai Shu, Suhang Wang, Thai Le, Dongwon Lee, and Huan Liu. 2018. Deep headline generation for clickbait detection. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 467–476. IEEE.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Dianbo Sui, Yubo Chen, Binjie Mao, Delai Qiu, Kang Liu, and Jun Zhao. 2020. Knowledge guided metric learning for few-shot text classification. *arXiv preprint arXiv:2004.01907*.

Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. 2019. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447. PMLR.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Sinong Wang, Han Fang, Madian Khabsa, Hanzi Mao, and Hao Ma. 2021. Entailment as few-shot learner. *arXiv preprint arXiv:2104.14690*.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2):165–210.

Mengzhou Xia, Guoqing Zheng, Subhabrata Mukherjee, Milad Shokouhi, Graham Neubig, and Ahmed Hassan Awadallah. 2021. Metaxl: Meta representation transformation for low-resource cross-lingual learning. *arXiv preprint arXiv:2104.07908*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Wenpeng Yin. 2020. Meta-learning for few-shot natural language processing: A survey. *arXiv preprint arXiv:2007.09604*.

Jinsung Yoon, Yao Zhang, James Jordon, and Mihaela van der Schaar. 2020. Vime: Extending the success of self-and semi-supervised learning to tabular domain. *Advances in Neural Information Processing Systems*, 33.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q Weinberger, and Yoav Artzi. 2020b. Revisiting few-sample bert fine-tuning. *arXiv preprint arXiv:2006.05987*.

Jing Zhou, Yanan Zheng, Jie Tang, Jian Li, and Zhilin Yang. 2021. Flipda: Effective and robust data augmentation for few-shot learning. *arXiv preprint arXiv:2108.06332*.

A Appendix

A.1 Implementation Details

We implemented our model and all baselines with PyTorch and run each experiment on a single NVIDIA GeForce RTX 3090 GPU. The hyperparameters are the same for all methods based on RoBERTa-large (the learning rate is 3e-6, the batch size is 4, the number of training epochs is 10). Following (Gao et al., 2021), we select the random seeds for sampling the training set and validation set as $\{13, 21, 42, 87, 100\}$.

A.2 Dataset Details

In general, we follow the experiment setting of (Gao et al., 2021). For datasets from GLUE (Wang et al., 2018) including SST-2 (Socher et al., 2013) and CoLA (Warstadt et al., 2019), we use the original development sets for testing. For datasets requiring cross-validation evaluation like MR (Pang and Lee, 2005), CR (Hu and Liu, 2004), MPQA (Wiebe et al., 2005) and Subj (Pang and Lee, 2004), we randomly sample 2,000 instances as the testing set and remove them from the training set. For the dataset SST-5 (Socher et al., 2013), we use the official test sets. The dataset statistics are shown in Table 4.

A.3 Baseline Details

The details of the baselines are as follows:

- **Majority**: The label is predicted by taking the majority class in the training set. We run this baseline on the full-data setting.
- **Fine-Tuning**: The prediction is based on the pre-trained language model that is fine-tuned with the specific training data. We run this baseline in the full-data and few-shot setting.
- **GPT-3** (Brown et al., 2020): GPT-3 in-context tuning in the zero-shot setting. We pack the training samples into the input together and directly conduct inference.
- **EFL** (Wang et al., 2021): An entailment-based prompt tuning framework. For fair comparison, we do not pretrain the language model on MNLI task but directly tune the language model as the prompt tuning paradigm.
- **LM-BFF** (Gao et al., 2021): A prompt tuning model that automatically searches for demonstrations, templates and label words. Note that LM-BFF utilizes one-to-one verbalizer for label word selection.
- **Prompt Tuning**: The standard Prompt-based Tuning augmented by a simple template or template-free.

A.4 Comparison of RoBERTa vs BERT

We conduct experiments to investigate the impact of the backbone model. Table 5 shows the results of using BERT-large(uncased) and RoBERTa-large. The experiment setting is the same as the main experiments. We can observe that our proposed PROMPTDA improves the performance of prompt tuning regardless of the backbone model.

A.5 The Verbalizer and Template Design

For each random seed of $\{13, 21, 42, 87, 100\}$, we can construct different training sets and validation sets. Thus, the verbalizers searched automatically for each run are different, which are shown as “label words (au.)” in Table 6. The verbalizers manually designed for each run are the same, which are shown as “label words (m.)” in Table 6. We follow previous works (Gao et al., 2021; Wang et al., 2021) and design a simple template “`It is [MASK]`” for each input.

Dataset	# Classes	# Length	# Train	# Test	Type	Labels
SST-2	2	19	6,920	872	sentiment	positive, negative
MR	2	20	8,662	2,000	sentiment	positive, negative
CR	2	19	1,775	2,000	sentiment	positive, negative
Subj	2	23	8,000	2,000	subjectivity	subjective, objective
CoLA	2	8	8,551	1,042	acceptability	grammatical, not_grammatical
MPQA	2	3	8,606	2,000	opinion polarity	positive, negative
SST-5	5	18	8,544	2,210	sentiment	v. pos., positive, neutral, negative, v. neg.

Table 4: Statistics of the datasets.

BERT-large	**SST-2**	**Subj**	**SST-5**
PT	82.3 (4.6)	80.3 (6.2)	34.5 (3.8)
PT + PROMPTDA	87.1 (3.1)	82.9 (3.3)	37.5 (2.8)
RoBERTa-large	**SST-2**	**Subj**	**SST-5**
PT	85.8 (5.8)	81.2 (5.7)	38.4 (4.7)
PT + PROMPTDA	**89.5 (2.9)**	**86.8 (3.1)**	**43.3 (1.6)**

Table 5: A comparison of RoBERTa-large vs BERT-large on template-augmented prompt tuning.

SST-2	label name	positive | negative
	label word (s.)	positive | negative
	label words (m.)	good perfect fantastic | terrible awful hilarious
	label words (au.)	brilliant amazing wonderful | not awful terrible
		great perfect brilliant | terrible disappointing bad
		beautiful perfect fantastic | terrible awful hilarious
		fantastic excellent beautiful | terrible awful worse
		wonderful, brilliant, fantastic | terrible done disappointing
MR	label name	positive | negative
	label word (s.)	positive | negative
	label words (m.)	positive, great, good | negative, terrible, bad
	label words (au.)	refreshing good beautiful | not terrible disappointing
		beautiful perfect fantastic | awful disappointing horrible
		fantastic wonderful beautiful | terrible awful funny
		fantastic incredible unforgettable | terrible funny bad
		excellent refreshing amazing | terrible wrong bad
CR	label name	positive | negative
	label word (s.)	positive | negative
	label words (m.)	good perfect fantastic | terrible awful hilarious
	label words (au.)	amazing fun cool | disappointing frustrating bad
		excellent fun cheap | awful horrible terrible
		free fun cool | bad painful useless
		fantastic brilliant incredible | terrible inevitable useless
		amazing great awesome | terrible awful horrible
Subj	label name	objective | subjective
	label word (s.)	actual | individual
	label words (m.)	good neutral fair | bad emotional personal
	label words (au.)	epic life America | madness not wrong
		life history significant | right that great
		what real interesting | me good great
		fiction interesting America | wonderful great brilliant
		disturbing terrifying key | bad not nonsense
CoLA	label name	grammatical | not_grammatical
	label word (s.)	good | bad
	label words (m.)	positive correct good | negative wrong bad
	label words (au.)	it wrong correct | ridiculous not good
		different sad interesting | complicated hilarious scary
		wrong interesting important | insane sad crazy
		all good important | bad new impossible
		how amazing normal | true him me
MPQA	label name	positive | negative
	label word (s.)	good | bad
	label words (m.)	good perfect fantastic | terrible awful hilarious
	label words (au.)	possible necessary adopted | wrong bad dark
		obvious awesome fun | then difficult gone
		right fun decided | reported unfair rejected
		accepted good great | unavoidable awful bad
		different good amazing | wrong bad funny
SST-5	label name	very positive | positive | neutral | negative | very negative
	label word (s.)	extraordinary | great | enough | boring | awful
	label words (m.)	great perfect excellent | good pretty wonderful | neutral normal fine | bad worse not | terrible awful ridiculous
	label words (au.)	good excellent unforgettable | hilarious inevitable funny | different time interesting | predictable bad over | dreadful boring horrible
		magnificent unforgettable fantastic | refreshing remarkable sublime | disappointing bad hilarious | neither predictable inevitable | depressing pathetic unnecessary
		wonderful fantastic incredible | terrifying refreshing interesting | hilarious done easy | better disappointing predictable | disgusting ridiculous horrible
		magnificent excellent too | stunning unexpected refreshing | simple done interesting | boring there worse | ridiculous sad weird
		great brilliant fantastic | extraordinary remarkable fascinating | enough terrible funny | awful bad worse | boring done unnecessary

Table 6: The verbalizer design (single label word (s.) for normal prompt tuning, label words manually designed (m.) and automatically searched (au.) for prompt tuning with PROMPTDA) over five runs on SST-2, CR, MR, Subj, CoLA, MPQA, SST-5 datasets.