Prompt Contrastive Transformation: An Enhanced Strategy for Efficient Prompt Transfer in Natural Language Processing

Shu Zhao Shiji Yang Shicheng Tan Zhen Yang Congyao Mei
Zhen Duan Yanping Zhang Jie Chen
School of Computer Science and Technology, Anhui University, China
zhaoshuzs2002@hotmail.com,{2337606702,1806545042}@qq.com
{uscyz094,ycduan,zhangyp2}@gmail.com
tsctan@foxmail.com,chenjie200398@163.com

Abstract

Prompt transfer is a transfer learning method based on prompt tuning, which enhances the parameter performance of prompts in target tasks by transferring source prompt embeddings. Among existing methods, weighted aggregation is effective and possesses the advantages of being lightweight and modular. However, these methods may transfer redundant or irrelevant information from the source prompts to the target prompt, leading to negative impacts. To alleviate this problem, we propose **P**rompt **C**ontrastive **T**ransformation (**PCT**), which achieves efficient prompt transfer through prompt contrastive transformation and attentional fusion. PCT transforms the source prompt into task-agnostic embedding and task-specific embeddings through singular value decomposition and contrastive learning, reducing information redundancy among source prompts. The attention module in PCT selects more effective task-specific embeddings and fuses them with task-agnostic embedding into the target prompt. Experimental results show that, despite tuning only 0.035% of task-specific parameters, PCT achieves improvements in prompt transfer for single target task adaptation across various NLP tasks.

1 Introduction

Fine-tuning pretrained language models (PLMs) has led to significant improvements across various downstream Natural Language Processing (NLP) tasks (Raffel et al., 2020). Moreover, a new tuning paradigm, named prompt tuning, has emerged (Lester et al., 2021; Li and Liang, 2021). It guides the model to output the desired results by inserting a small number of tunable prompt embeddings into the input (or into the hidden states [Li and Liang, 2021]), but this tuning method is susceptible to the influence of prompt initialization. Recent work has introduced prompt transfer (Vu et al., 2022; Jung et al., 2024; Wu et al., 2024; Belanec et al., 2024; Lan et al., 2024), which is a transfer learning method based on prompt tuning that enhances the parameter performance of prompts through transferring source prompt embeddings (Figure 1). It offers an efficient solution for parameter tuning in large language models.

To effectively transfer the source prompts to the target prompt, a common approach is direct weighted aggregation of prompts. This method determines the weight of each source prompt by calculating the relevance between the target task and each source prompt. Popular methods for weight calculation include cosine similarity (SPoT; Vu et al., 2022) and attention scores (ATTEMPT; Asai et al., 2022). Another approach is based on knowledge distillation, which regards source prompts as teachers and target prompts as students. This includes single-task knowledge distillation (Zhong et al., 2024) and multi-task knowledge distillation (Wang et al., 2023). They enrich the information of the target prompt through prompt transfer, thereby enhancing the performance of the parameters.

Among the existing methods, the weighted aggregation approach not only effectively finishes prompt transfer but also maintains the benefits of being lightweight and modular. This is because it doesn't require reasoning time and space for each source prompt, unlike knowledge distillation. However, redundant and irrelevant information in the source prompts may impact the effectiveness of weighted aggregation. On one hand, in order to facilitate prompt tuning, language models cast each task, such as SST-2 (sentiment classification), MNLI (natural language inference), and ReCoRD (reading comprehension question answering), as a text-to-text generation task. This results in the presence of mutually overlapping

Transactions of the Association for Computational Linguistics, vol. 13, pp. 848–860, 2025. https://doi.org/10.1162/tacl.a.22
Action Editor: Julia Hockenmaier. Submission batch: 5/2024; Revision batch: 10/2024; Published 8/2025.

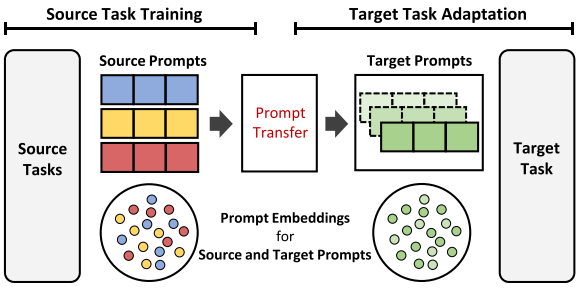

Figure 1: The source task prompts from different tasks transfer valid information to the target task through prompt transfer.

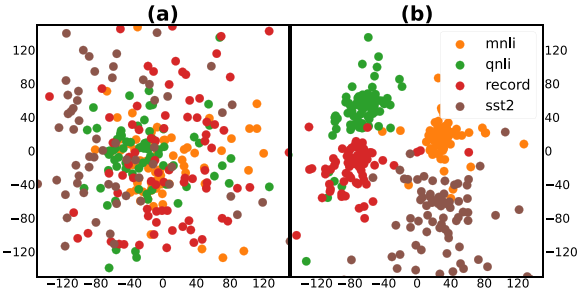

Figure 2: We extracted and visualized the principal components from four source prompts, (a) showing the distribution of the source prompt, and (b) depicting the distribution of source task-specific embeddings extracted from the source prompts by PCT.

redundant information among source prompts (Figure 2a), which in turn affects the weight distribution during the aggregation process. On the other hand, the presence of irrelevant information directly impacts the adaptation of the target task. This is primarily due to the over-sensitivity between source and target (Zhong et al., 2024), and this issue can only be avoided if every member in the source tasks is highly matched with the target task.

To alleviate the negative impact of redundant and irrelevant information from source prompts on the weighted aggregation of prompt transfer, we propose a method named **P**rompt **C**ontrastive **T**ransformation (**PCT**), which achieves effective prompt transfer through prompt transformation and fusion. Firstly, (1) PCT employs singular value decomposition to extract task-agnostic embedding from the multiple source prompts and proposes a contrastive learning approach to separate task-specific embeddings from the remaining information in each source prompt, enhancing their distinctiveness to reduce information redundancy. As shown in Figure 2b, after prompt contrastive transformation, the distribution of each task-specific embedding is more distinguishable from each other. Then, (2) PCT introduces an attention-based weighted aggregation method for selecting task-specific embeddings more relevant to the target task and assigning weights, thereby reducing the impact of irrelevant information. Finally, (3) PCT integrates the weighted task-specific embeddings with the task-agnostic embedding into the target prompt, ensuring the integrity of effective information within the prompt and achieving prompt transfer.

The main contributions of this work are summarized as follows in three points:

1. We propose PCT, which transforms source prompts into task-agnostic embedding and task-specific embeddings through singular value decomposition and contrastive learning, effectively reducing the information redundancy among source prompts.
2. PCT introduces an effective method to select and integrate task-specific embeddings that are more relevant to the target task, reducing the impact of irrelevant information on the target task, and achieving efficient prompt transfer.
3. Experiments show that, despite tuning only 0.035% of task-specific parameters, PCT achieves improvements in prompt transfer for single target task adaptation across 21 NLP tasks, and it is highly competitive when compared to full fine-tuning.[1]

2 Related Work

2.1 Prompt Tuning

Since the parameter scale of language models (LMs), such as T5 (Raffel et al., 2020), has reached the billion level, parameter-efficient tuning has gained increasing attention (Ding et al., 2023). Among these methods, prompt tuning has particularly interested researchers, as it effectively reduces the parameter scale of model tuning when LMs are adapted to downstream tasks. It involves inserting a small, trainable prompt vector into the input of the language model (or into each layer of the model), which guides the model towards generating the intended outputs (Lester et al., 2021; Li and Liang, 2021). Moreover, research has shown

[1] Our code is available at https://github.com/AHU-YangSJ/PCT.

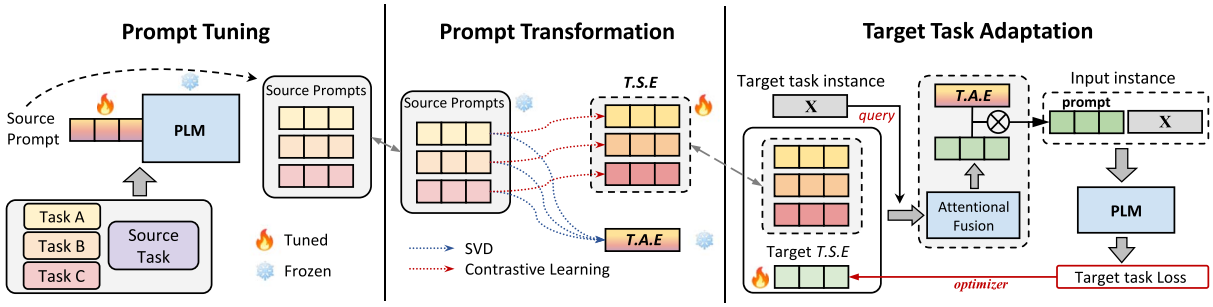

Figure 3: Our proposed PCT consists of two steps: Prompt Transformation, and Target Task Adaptation. In the Prompt Transformation step, PCT transforms the source prompt into **task-agnostic embedding (T.A.E)** and **task-specific embeddings (T.S.E)** through singular value decomposition and contrastive learning(Section 3.2). During the Target Task Adaptation step, attention scores are calculated between each task-specific embeddings and the input instances X of the target task to determine their respective weights, where $\otimes$ denotes the hadamard product, used to integrate task-agnostic embedding and task-specific embeddings.

that an appropriate initialization significantly enhances the parameter performance of prompts. A straightforward initialization approach is to use existing discrete prompt search methods to create or discover discrete prompts for initializing continuous prompts (Qin and Eisner, 2021), but designing discrete prompts requires considerable manual effort. Recent research has shown that continuous prompts can be reused or initialized across different tasks, a process referred to as prompt transfer.

2.2 Prompt Transfer

Prompt transfer is a transfer learning method based on prompt tuning, which enhances the parameter performance of the prompt on the target task by transferring source prompts (Su et al., 2022).

A common approach is to directly retrieve and aggregate multiple source prompts and transfer them to the target prompt (Vu et al., 2022; Li et al., 2022). In these methods, SPoT (Vu et al., 2022) calculates the weight of each source prompt based on the cosine similarity between source and target, PTG (Li et al., 2022) performs spectral clustering on multiple source prompts and adapts to the target task through attention queries, while ATTEMPT (Asai et al., 2022) uses an attention mechanism to fuse multiple source prompts for adaptation to the target task. Another method involves using knowledge distillation (Zhong et al., 2024; Wang et al., 2023), which includes single-task and multi-task knowledge distillation. These methods regard source prompts as teachers and target prompts as students, achieving prompt transfer through distillation. Additionally, there are methods that train meta-learners to capture cross-task knowledge from source tasks (Wang et al., 2021), and methods that have achieved cross-model prompt transfer (Su et al., 2022).

Among the above prompt transfer methods, the weighted aggregation methods also have the advantages of being lightweight and modular (Asai et al., 2022), while multi-task knowledge distillation (Wang et al., 2023) has achieved the better comprehensive performance.

2.3 Contrastive Learning

In the field of NLP, contrastive learning is commonly employed to learn text representations. By comparing the similarities and differences between pairs of texts, better sentence representations can be acquired (Jiang et al., 2022). These representations can be utilized to enhance model performance across various NLP tasks, such as sentiment analysis, text classification, and question-answering systems. Moreover, contrastive learning is also frequently used for feature separation. In the work on adaptive contrastive knowledge distillation (Guo et al., 2023), contrastive learning was employed to achieve feature separation and yielded promising results. In our proposed PCT, contrastive learning, in conjunction with singular value decomposition, is used to achieve prompt contrastive transformation, reducing the impact of redundant and irrelevant information in source prompts on prompt transfer.

3 Methodology

In the following sections, we will demonstrate how PCT achieves efficient prompt transfer (Figure 3).

3.1 Prompt Tuning

Prompt tuning can achieve model adaptation through a small segment of embedding parameters. Formally, the input sequence X is concatenated with a soft prompt embedding $P_i \in \mathbb{R}^{m \times d}$ to form $[P_i; X]$, where m denotes the length of the prompt, and d denotes the dimension of the language model, which is then input into the frozen parameter language model θ for output (Lester et al., 2021). Finally, prompt tuning is achieved by optimizing the following loss function and updating the prompt parameters P_i.

$$L_{PLM} = -\sum_i \log P\left(y_i | x_i; \theta, P_i\right) \quad (1)$$

Additionally, prompt can be acquired directly from the open-source market without the need for training, hence this is not a necessary step for PCT.

3.2 Prompt Transformation

Through prompt tuning in Section 3.1 or the open-source market, we can obtain n source prompts. In this section, we will perform prompt transformation, decomposing them into task-agnostic embeddings and task-specific embeddings. It is important to note that this is a pre-training process that requires minimal computational resources.

3.2.1 Task-agnostic Embedding Extraction

Before prompt transfer, it is necessary to extract the task-agnostic embedding $P^* \in \mathbb{R}^{m \times d}$ from each source prompt and save it. This embedding contains most of the key information from multiple source prompts, ensuring the integrity of the information during the embedding fusion process.

Formally, we first stack these n source prompts into a full matrix $P = \left[P_1^\top, P_2^\top, \ldots, P_n^\top\right]^\top \in \mathbb{R}^{(n \times m) \times d}$. After decomposing $P^\top$ through Equation 2, we can obtain the left singular vectors u, right singular vectors v, and singular values s, then extract $u_{[0 \sim k]} \in \mathbb{R}^{d \times k}$, $s_{[0 \sim k]} \in \mathbb{R}^{1 \times k}$, and $v_{[0 \sim k]} \in \mathbb{R}^{k \times (n \times m)}$, and obtain the task-agnostic embedding P^* through Equation 3.

$$u, s, v = SVD\left(\left[P_1^\top, P_2^\top, \ldots, P_n^\top\right]\right) \quad (2)$$

$$P^* = \left(u_{[0 \sim k]} \cdot diag(s_{[0 \sim k]}) \cdot v_{[0 \sim k]}\right)^\top_{[0 \sim m]} \quad (3)$$

where $SVD(\cdot)$ denotes singular value decomposition, and $diag(\cdot)$ is a diagonalization function that converts a vector into a diagonal matrix. $(\cdot)^\top_{[0 \sim m]}$ represents selecting the first m column vectors from the embedding matrix and then transposing it to align with the source prompt. The final extracted task-agnostic embedding is $P^* \in \mathbb{R}^{m \times d}$.

3.2.2 Task-specific Embeddings Separation

Next, we separate the task-specific embeddings from the remaining information of each source prompt, which will be used for attention-weighted aggregation. Specifically, we initialize a trainable parameter matrix of the same shape as P for the task-specific embedding $M = \left[M_1^\top, M_2^\top, \ldots, M_n^\top\right]^\top \in \mathbb{R}^{(n \times m) \times d}$, where each $M_i \in \mathbb{R}^{m \times d}$ can be used to reconstruct the source prompt by combining it with the task-agnostic embedding P^* through $Q_i = M_i \otimes P^*$, and by stacking them we obtain restored prompt matrix $Q = [Q_1^\top, Q_2^\top, \ldots, Q_n^\top]^\top \in \mathbb{R}^{(n \times m) \times d}$. To facilitate the formalization of separating task-specific embeddings from source prompts, we define the cosine similarity between the two matrices as $sim(\cdot, \cdot)$, as shown in the following equation.

$$sim(U, V) = \frac{1}{m} \sum_{i=1}^{m} \cos(U_i, V_i) \quad (4)$$

In this equation, $\cos(\cdot, \cdot)$ denotes the function for calculating the cosine similarity between row vectors, U_i and V_i are row vectors in matrix U and V, respectively.

We propose a contrastive learning method to train the task-specific embeddings (Figure 4). Specifically:

1. The positive sample pairs consist of the submatrix P_i of the source prompt P and the submatrix Q_i of the restored prompt Q, to ensure that the separated task-specific embedding M_i can fully incorporate the remaining information in the source prompt P_i.

2. Negative sample pairs are formed by pairing the n submatrices of the trainable parameter matrix for task-specific embedding M, namely $\{M_1, M_2, \ldots, M_n\}$, with each other. This approach increases the distinctiveness between them, which is beneficial for the allocation of attention weights in the weighted aggregation process.

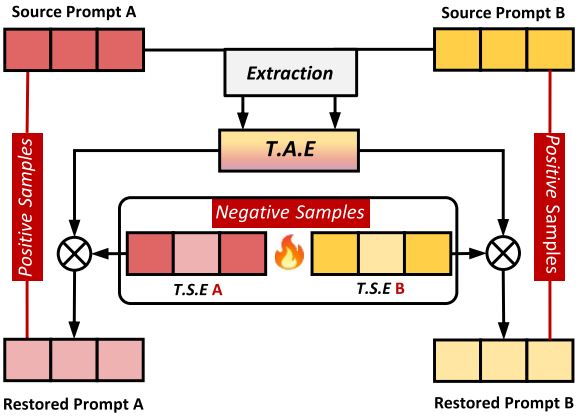

Figure 4: We derive task-agnostic embedding (T.A.E) from source prompts, combined with task-specific embeddings (T.S.E) to construct the restored prompt. Source and restored prompts are positive pairs to ensure embedding integrity, while n task-specific embeddings pair with each other to form negative pairs to minimize information redundancy.

The loss function for the separation of task-specific embeddings is designed as follows:

$$POS = \exp(sim(P_i, Q_i)/\tau) \tag{5}$$

$$NEG = \sum_{k=1}^{n} exp[\mathbf{1}_{k \neq i} sim(M_i, M_k)/\tau] \tag{6}$$

$$L_c = -\sum_{i=1}^{n} \log \frac{POS}{POS + NEG} \tag{7}$$

$$L = L_c + MSE(P, Q) \tag{8}$$

where τ is the temperature, which is used to control the strength of task-specific embedding separation, $Q_i = M_i \otimes P^*$, and $M = \left[M_1^\top, M_2^\top, \ldots, M_n^\top\right]^\top$ is the only optimizable parameter in this stage. This step does not require access to any training data.

3.3 Target Task Adaptation

In this section, we will re-integrate the task-agnostic embedding and task-specific embeddings transformed from the source prompts into the target prompt and adapt it to the target task.

3.3.1 Task-specific Embeddings Attention

PCT stacks a small trainable target task-specific embedding $M_{target} = M_{n+1} \in \mathbb{R}^{m \times d}$ behind the source task-specific embeddings $M = [M_1^\top, M_2^\top, \ldots, M_n^\top; M_{n+1}^\top]^\top$, with each embedding M_i determining its contribution to the target prompt by calculating attention scores with the target task input instances. This is achieved through a simple attention module that computes attention weights $\alpha_1, \alpha_2, \ldots, \alpha_{n+1}$, which represent the relevance between the input instances and each task-specific embedding.

Since the input embedding $X \in \mathbb{R}^{l \times d}$ and task-specific embedding $M_i \in \mathbb{R}^{m \times d}$ have different sequence lengths, we first perform max pooling operations on them, obtaining $\hat{X} \in \mathbb{R}^d$ and $\hat{M}_i \in \mathbb{R}^d$. Finally, we represent the attention score α_i by calculating the inner product between $\hat{X}$ and $\hat{M}_i$, using the softmax function and temperature T for post-processing to avoid over-confidence (this calculation method following ATTEMPT (Asai et al., 2022). This is shown as follows:

$$\alpha_i = \frac{\exp\left(\hat{M}_i \hat{X}\right)/T}{\sum_{k=1}^{n+1} \exp\left(\hat{M}_k \hat{X}\right)/T} \tag{9}$$

3.3.2 Determine the Target Prompt

Finally, we need to weight the task-specific embeddings and aggregate it with the task-agnostic embedding into the target prompt. PCT uses $\alpha_1, \alpha_2, \ldots, \alpha_{n+1}$ to perform weighted aggregation on the task-specific embeddings, and updates the target task-specific embedding M_{target} through Equation 10.

$$M_{instance} = M_{target} + \sum_{i=1}^{n+1} \alpha_i M_i \tag{10}$$

Subsequently, we fuse the target task-specific embedding M_{target} of this instance with the task-agnostic embedding P^* of the source prompts, to obtain the final prompt $P_{instance} = M_{instance} \otimes P^*$ for this instance.

We achieve prompt tuning by optimizing the loss in Equation 1, thereby completing the target task adaptation, and M_{target} is the only parameter that is updated during the target task adaptation.

3.4 Parameter Efficiency

For each task, PCT introduces a new trainable target task-specific embedding $M_{target} \in \mathbb{R}^{m \times d}$, where m and d are the two dimensions of the matrix. Since the prompt contrastive transformation of the source prompts is a pre-training process that is completed independently and requires only minimal computational resources, the amount of

parameters trained for each target task is $m \times d$. The parameter amount of Adapter (Houlsby et al., 2019) and fine tuning increases rapidly with the scale of the backbone model, but the parameter amount of PCT is only related to the prompt length and the dimension of the language model (LM). For example, with a prompt length of 100, when t5-base (220 Million) is the base model, only 0.035% of the parameters are tuned, and when t5-xxl (11 billion) is the base model, only 0.0037% of the parameters are tuned, which is consistent with vanilla prompt tuning (Lester et al., 2021).

4 Experiment

4.1 Datasets and Tasks

We utilize six large-scale datasets as source tasks and evaluate on 21 different target tasks, including entailment, paraphrase detection, sentiment analysis, question answering (QA), and commonsense reasoning.

4.1.1 Source Task

We use the following datasets with more than 100k annotations in total from GLUE (Wang et al., 2019b), SuperGLUE (Wang et al., 2019b), and MRQA (Fisch et al., 2019) for source prompts: MNLI, QNLI, QQP, SST2, ReCoRD, and SQuAD.

4.1.2 GLUE and SuperGLUE

We use 8 GLUE tasks (Wang et al., 2019b) and 5 SuperGLUE (Wang et al., 2019a) tasks as target datasets to test the model's natural language understanding abilities: BoolQ, CB, MultiRC, WiC and WSC for SuperGLUE benchmark tests; and RTE, CoLA, STSB, MRPC, MNLI, QQP, QNLI, and SST-2 for GLUE benchmark tests.

4.1.3 Question Answering

We use the MRQA 2019 (Fisch et al., 2019) shared task data to test on four large-scale QA datasets: Natural Questions (NQ), HotpotQA (HP), NewsQA (News), and SearchQA (SQA).

4.1.4 Others

We include other experiments on four different datasets, whose tasks are related to the source tasks but domains differ. SciTail (Khot et al., 2018) is a scientific entailment dataset. Yelp-2 (Zhang et al., 2015) is a sentiment analysis dataset on Yelp reviews. WinoGrande (Sakaguchi et al., 2021) is commonsense reasoning task in multiple choice format. PAWS-Wiki (Zhang et al., 2019) is a Wikipedia-based paraphrase detection dataset.

4.2 Baselines

We compare PCT with the following baselines: (1) Full finetuning (FT) and Vanilla prompt tuning (PT) (Lester et al., 2021). (2) Existing prompt transfer methods include retrieval and aggregation approaches such as SPoT (Vu et al., 2022) and ATTEMPT (Asai et al., 2022), as well as knowledge distillation methods like PANDA (Zhong et al., 2024) and MPT (Wang et al., 2023). (3) Popular parameter-efficient tuning methods, including Adapters (Houlsby et al., 2019), BitFit (Zaken et al., 2022), LoRA (Hu et al., 2022), and QLoRA (Dettmers et al., 2023).

Among these methods, FT, PT, single prompt transfer (SPoT and PANDA), multi-prompt transfer (ATTEMPT, MPT, and PCT), and popular parameter-efficient tuning methods can only access target task data during model adaptation.

4.3 Metrics

For the 21 target task datasets involved in the experiments, we use pearson correlation for STS-B, matthews correlation for CoLA, F1 for MultiRC (Multi) and MRQA, and accuracy for other tasks as the evaluation metric.

4.4 Models and Implementation Details

We directly cite the data reported in published papers, such as ATTEMPT (Asai et al., 2022) and MPT (Wang et al., 2023), and when adapting to the target task, we follow the publicly available code to set the same experimental environment for each baseline method to ensure fairness.

4.4.1 Models

Following the standard method of prompting adjustment (Lester et al., 2021; Asai et al., 2022), we primarily use the publicly pre-trained T5-Base model with 220M parameters for our experiments. In addition to analytical experiments, we introduce a larger-scale model (11B) to test the performance of PCT on models with different parameter sizes.

4.4.2 Prompt Tuning

Source Prompt is obtained through prompt tuning (Lester et al., 2021) and we set the prompt length to 100. We tune the source prompts for six datasets from five epochs. We use the checkpoint with the best development score as our source prompt, and

Datasets		GLUE									SuperGLUE					
Method	param / task	MNLI (363K)	QQP (364K)	QNLI (105K)	SST-2 (67K)	RTE (2.5K)	CoLA (8.5K)	STS-B (7K)	MRPC (3.7K)	Avg.	Multi (5.1K)	BoolQ (9.4K)	WiC (6K)	WSC (554)	CB (250)	Avg.
Fine Tuning	220M	**86.8**	**91.6**	93.0	**94.6**	71.9	61.8	89.7	**90.2**	84.9	72.8	81.1	**70.2**	59.6	85.7	73.9
Adapter	1.9M	86.5	90.2	93.2	93.8	71.9	**64.0**	90.7	85.3	84.5	75.9	**82.5**	67.1	**67.3**	85.7	75.7
BitFit	280K	85.3	90.1	93.0	94.2	67.6	58.2	**90.9**	86.8	83.3	74.5	79.6	70.0	59.6	78.6	72.5
LoRA	880K	83.7	86.9	92.0	94.0	79.1	61.3	90.5	87.3	84.4	75.7	80.0	68.7	55.8	**94.6**	75.0
QLoRA	880K	85.8	89.7	93.5	94.3	81.6	61.3	90.6	87.2	85.5	75.6	81.8	67.1	63.5	**94.6**	76.5
Prompt Tuning	77K	81.3	89.7	92.8	90.9	54.7	10.6	89.5	68.1	72.2	58.7	61.7	48.9	51.9	67.9	57.8
SPoT	77K	85.4	90.1	93.0	93.4	69.8	57.1	90.0	79.7	82.3	74.0	77.2	67.0	50.0	46.4	62.9
ATTEMPT	232K	84.3	90.3	93.0	93.2	73.4	57.4	89.7	85.7	83.4	74.4	78.8	66.8	53.8	78.6	70.5
MPT	78K	85.9	90.3	93.1	93.8	79.4	62.4	90.4	89.1	85.6	74.8	79.6	69.0	**67.3**	79.8	74.1
PANDA	77K	82.2	90.1	93.3	94.2	79.8	45.8	89.8	87.9	82.9	73.4	77.8	64.6	63.5	82.4	72.3
Ours	77K	86.1	90.6	**93.8**	**94.6**	**82.3**	63.4	90.5	88.3	**86.2**	**76.0**	**82.1**	**70.2**	64.5	**94.6**	**77.5**

Table 1: Results on GLUE and SuperGLUE. param/task represents the number of parameters trained for each target task. The bold text represents the best results on this task, while the underlined text denotes the best results among the prompt transfer methods. Most of the experimental results are derived from ATTEMPT (Asai et al., 2022) and MPT (Wang et al., 2023), and the same experimental setup was followed as theirs.

each source prompt is initialized by randomly sampled tokens.

4.4.3 Prompt Transformation

PCT performs singular value decomposition on the large matrix composed of 6 (4 in GLUE experiments) source prompts and selects the top 100 singular values and singular vectors to form the task-agnostic embedding through matrix multiplication. Additionally, when executing task-specific embeddings extraction and separation, we set the temperature τ in the contrastive learning (Equations 5 and 6) to 0.2, the learning rate to 0.3, and update for 24,000 steps, with a total time of about 30 minutes. In the submodule analysis section (Figure 6), we will set different numbers of singular values and singular vectors, as well as the temperature τ for contrastive learning, to analyze their impact on PCT. This is an independent pre-training phase that requires minimal computational resources and time.

4.4.4 Target Task Adaptation

This work focuses on model adaptation in single-task scenarios; therefore, all of our experiments were conducted on a single target task. In terms of the target task dataset parameters, we follow ATTEMPT (Asai et al., 2022) and set the maximum token length of the MRQA dataset to 512, the maximum token length of MultiRC to 348, and the maximum token length of all other datasets to 256. We set $T = d \times \exp(1)$, where d is the dimension of the language model (LM), to control the soft maximum temperature in Equation 9. The prompt length m is 100. All of our experiments were conducted on a single target task.

For training, we use Adam (Kingma and Ba, 2015) to optimize the objective function, using a learning rate of 0.1 for the SuperGLUE, Yelp, Winogrande, SciTail, and PAWS, and a learning rate of 0.3 for other experiments. Weight decay is set to 1×10^{-5}. All our experiments are conducted on a single GPU with 24GB memory, except for T5-XL (3B) which runs on a single 40GB A100 GPU. The batch size per GPU is 32; for MRQA, we set the batch size per GPU to 16 and the gradient accumulation steps to 2. In addition, for T5-XXL (11B), due to the model's large size, we choose to quantize it to 8-bit and collect the corresponding experimental results on some of the datasets. For the T5-XL and T5-XXL models, we set the batch size per GPU to 4 and the gradient accumulation steps to 4, with the learning rate set to 0.1.

5 Main Results

5.1 Full-dataset Adaptation

Tables 1 and 2 present the experimental results of various baseline methods across the four benchmarks. On GLUE and SuperGLUE, PCT achieves new state-of-the-art performance (Figure 5). Compared to the basic Prompt Tuning (Lester et al., 2021), PCT offers a relative improvement of

Datasets		MRQA					Others				
Method	**param / task**	**NQ (100K)**	**HP (72K)**	**SQA (117K)**	**News (74K)**	**Avg.**	**WG (40K)**	**Yelp (100K)**	**SciTail (27K)**	**PAWS (49K)**	**Avg.**
Fine Tuning	220M	**75.1**	77.5	81.1	65.2	**74.7**	**61.9**	96.7	**95.8**	94.1	**87.1**
Adapter	1.9M	74.2	77.6	**81.4**	**65.6**	**74.7**	59.2	96.9	94.5	**94.3**	86.2
BitFit	280K	70.7	75.5	77.7	64.1	72.0	57.2	94.7	94.7	92.0	84.7
LoRA	880K	73.8	76.1	80.2	65.2	73.8	59.2	**97.0**	93.8	93.4	85.9
QLoRA	880K	73.9	**77.8**	79.6	64.5	73.9	58.5	**97.0**	94.7	93.7	85.9
Prompt Tuning	77K	67.9	72.9	75.7	61.1	69.4	49.6	95.1	95.1	55.8	73.9
SPoT	77K	68.2	74.8	75.3	58.2	69.1	50.4	95.4	91.2	91.1	82.0
ATTEMPT	232K	70.4	75.2	77.3	62.8	71.4	57.6	96.7	93.1	92.1	84.9
MPT	78K	72.0	75.8	77.2	63.7	72.2	56.5	96.4	95.5	93.5	85.5
PANDA	77K	70.7	74.3	75.2	62.6	70.7	55.2	96.3	91.7	93.2	84.1
Ours	77K	72.5	76.8	78.7	63.8	73.0	58.5	**97.0**	95.5	94.1	86.3

Table 2: Results on MRQA datasets, WinoGrande (WG), Yelp, Scitail, and PAWS. The bold text represents the best results on this task, while the underlined text denotes the best results among the prompt transfer methods.

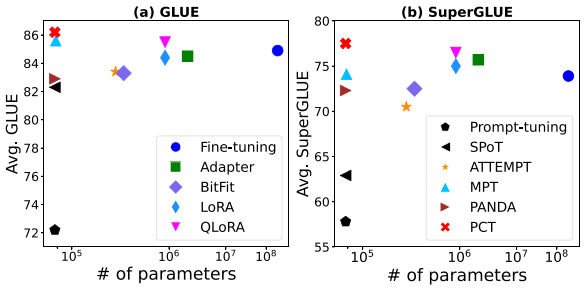

Figure 5: This figure illustrates the parameter scale and performance of existing prompt transfer methods and other parameter-efficient tuning methods.

14% on GLUE and 19.7% on SuperGLUE, using the same number of task-specific parameters. This emphasizes the advantages of multi-prompt transfer.

The parameter performance of the PCT method also surpasses other prompt transfer methods. Compared to weighted aggregation methods (like ATTEMPT), our method achieved relative improvements of 2.8%, 7%, 1.6%, and 1.4% on the GLUE, SuperGLUE, MRQA, and Other datasets, respectively. Compared to knowledge distillation methods (like MPT), PCT also demonstrated better parameter performance. Although it was slightly lower than MPT on the MRPC and WSC datasets, PCT still enhanced the parameter performance of weighted aggregation methods on such tasks. This series of experiments shows that PCT's prompt contrastive transformation and fusion can effectively alleviate the negative impact of redundant and irrelevant information on prompt transfer methods based on weighted aggregation. (Further validation is provided in Section 6.2.)

Moreover, compared to parameter-efficient adapters, our PCT still exhibits the best overall performance in GLUE and SuperGLUE. In MRQA and other datasets, since the Q&A task-related information in the source prompts is derived only from SQuAD, prompt tuning and prompt transfer generally perform worse on Q&A tasks compared to full fine-tuning and parameter-efficient adapters. However, our PCT still outperforms other prompt transfer methods comprehensively, further reducing this performance gap, and its parameter efficiency is more than ten times that of parameter-efficient tuning adapters. This indicates that prompt tuning still has research value and opportunities for work in the context of large language models.

5.2 Few-shot Adaptation

In most cases, there is always a significant gap between the parameter performance of prompt-tuning and full fine-tuning, especially in few-shot scenarios. Following previous work (Asai et al., 2022), we conducted multiple few-shot experiments on the RTE, BoolQ, CB, and SciTail tasks to measure the performance of PCT with k training samples (k = 4, 16, 32). Table 3 shows the results of PCT and other baseline methods, including full fine-tuning (FT), prompt tuning (PT), SPoT (ST), ATTEMPT (ATP), and MPT.

From Table 3, it can be observed that the PT has difficulty adapting in few-shot settings, with a large gap between it and full fine-tuning. Weighted aggregation methods (such as ATP) effectively

k-shot		FT	PT	ST	ATP	MPT	PCT
RTE	4	52.3	56.3	56.6	62.3	62.6	**63.3**
	16	61.2	54.7	59.6	68.6	64.8	**68.9**
	32	63.5	54.7	62.1	65.7	59.7	**68.2**
BoolQ	4	50.5	61.6	50.5	61.8	62.2	**69.8**
	16	56.5	61.9	50.6	60.0	63.3	**70.9**
	32	58.4	61.7	61.2	65.3	68.9	**72.4**
CB	4	57.7	53.5	71.4	82.1	73.6	**82.1**
	16	77.0	63.5	64.3	78.5	78.6	**79.2**
	32	80.0	67.8	64.3	85.7	82.1	**87.9**
SciTail	4	79.6	57.7	69.6	80.2	80.2	**80.2**
	16	80.0	60.8	71.9	79.5	**87.3**	80.3
	32	81.9	60.2	71.9	80.2	**86.3**	84.2

Table 3: Few-shot learning results with $k = \{4, 16, 32\}$ on BoolQ, CB, and SciTail. FT: Fine tuning, PT: Prompt tuning, ST: SPoT, ATP: ATTEMPT.

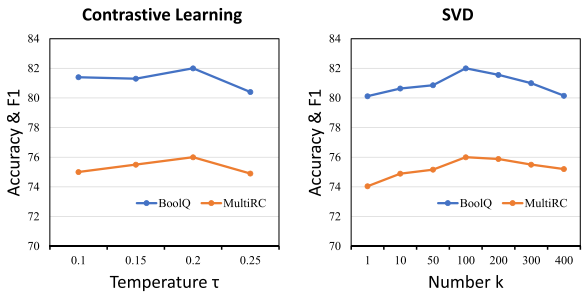

Figure 6: The impact of different contrastive learning temperatures (τ) and numbers of singular values (k) on the performance of PCT.

alleviate this issue, and PCT brings a comprehensive improvement to the weighted aggregation approaches Furthermore, we can also observe that PCT comprehensively surpasses full fine-tuning and outperforms other prompt transfer methods. Although it is lower than MPT on some cases of SciTail, PCT still brings stable improvement to weighted aggregation methods, such as ATP, which further demonstrates the effectiveness of prompt contrastive transformation and fusion.

5.3 Model Parameter Scale

We followed the work of Asai et al. (2022) and conducted extended experiments to analyze the performance changes of PCT when the scale of pretrained model parameters increases across three SuperGLUE tasks. As shown in Figure 7, FT and PT may see a decline in performance in certain scenarios. However, as the scale of parameters increases, the performance of PT gradually approaches that of FT, which aligns with the perspective of Lester et al. (2021). Our proposed PCT exhibits moderate performance on small-scale models, which is related to the original performance limitations of source prompts. However, as the scale of parameters increases (from 220M to 11B), the PCT method consistently brings improvements to the parameter performance of prompt tuning (PT) and the prompt weighted aggregation method (ATP), and outperforms other prompt transfer methods.

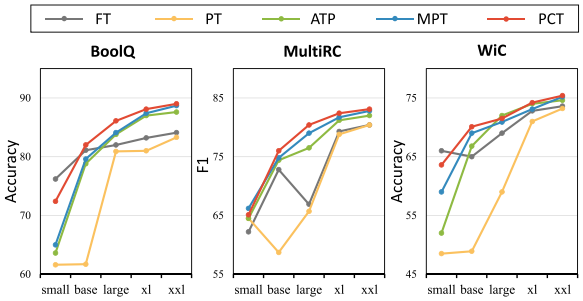

Figure 7: Performance of different baseline methods (FT, PT, and multi-source prompts transfer) on models of various parameter scales.

6 Extended Analysis

6.1 Submodule

To clearly understand the impact of these two sub-modules on PCT, we conducted two sets of experiments. Under the condition that all other factors remained unchanged, we adjusted the key parameters in each sub-module, namely, the values of k and τ, and observed the changes in Accuracy (BoolQ) and F1 (MultiRC).

The experimental results show that (1) as the contrastive learning temperature τ gradually increases (Figure 6 left), the performance of PCT on the two tasks generally first rises and then falls. We analyze this to be due to τ representing the strength of task-specific embeddings separation. The lower the τ, the greater the separation strength. An excessively low τ value will cause the distribution of the restored prompts matrix Q_i to deviate from the source prompt P_i (Equation 5), while an excessively high τ value will result in incomplete separation of task-specific embeddings. (2) As the number of singular values and vectors gradually increases from 1, the overall performance of PCT shows fluctuations (Figure 6 right). This is because an excessively high k value leads to the transmission of some redundant or irrelevant information to the target tasks through the task-agnostic embedding, while an excessively low k value results in the task-agnostic embedding not being fully capable of accommodating key information,

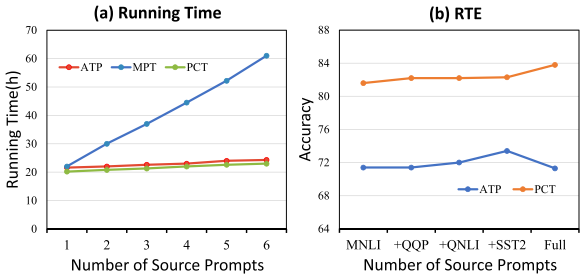

Figure 8: In multi-source prompt transfer, the impact of varying numbers of source prompts on runtime and parameter performance. (a) represents the runtime for approximately 20,000 steps of gradient descent.

ultimately causing a lack of information in target prompt.

6.2 Lightweight and Modularization

Similar to weighted aggregation prompt transfer methods like ATTEMPT, our PCT also maintains the characteristics of being lightweight and modular. We estimated the runtime of several prompt transfer methods, as shown in Figure 8(a). MPT, on the other hand, requires longer runtime as it infers multiple teacher prompts, and its runtime increases with the number of source prompts. Our PCT method is slightly higher than ATTEMPT (ATP), and both only increase at a very slow rate.

Moreover, the modular nature of PCT and ATTEMPT (ATP) allows for the instant assembly of various source prompts. Therefore, we analyzed the impact of different source tasks on PCT and ATP in the RTE task of the GLUE dataset. As shown in Figure 8(b), both methods' performance slowly increases up to SST2. However, after adding SQuAD and ReCoRD, ATP experiences a decline in performance, while our PCT continues to improve. This is because its singular value decomposition has already extracted most of the key information, and the attention mechanism further reduces the impact of irrelevant task-specific embeddings and selects more useful parts.

7 Ablation Study

To verify the effectiveness of PCT, we conducted ablation experiments on three tasks where comprehensive data experiments showed performance improvements, and we re-ran the corresponding experiments. To test the effect of attention-weighted aggregation, we replaced the original attention module with average score weighting. Compared to PCT (as shown in Table 4, row 4), the performance decreased by 1.5%, 3.2%, and 1.2% respectively, which demonstrates the contribution of attention-weighted aggregation to the performance enhancement of PCT.

Transformation	Attention	BoolQ	RTE	HP
×	×	73.7	71.3	74.8
×	√	77.2	73.6	75.2
√	×	80.5	79.1	75.6
√	√	82.0	82.3	76.8

Table 4: Results of ablation studies. HP: HotpotQA.

In order to test the effectiveness of prompt contrastive transformation, we will remove the prompt transformation module and replace the transformed task-agnostic embedding and task-specific embeddings with the source prompt. This ablation version (Table 4, row 2) is similar to the ATTEMPT method. Compared with PCT (Table 4, row 4), the performance drops by 4.8%, 8.7%, and 1.6% respectively, which reflects the contribution of prompt contrastive transformation to PCT. In addition, we can also observe that this ablation version loses more performance compared to the previous ablation version (Table 4, row 3), which further demonstrates the necessity of prompt contrastive transformation in PCT.

We removed the prompt contrastive transformation and attention modules, using simple average scoring for prompt aggregation. This version (Table 4, row 1) showed a significant performance decline, indicating the significance of integrating these modules in PCT.

8 Conclusion

In this paper, we propose an effective prompt contrastive transformation method that expands the differentiation between source tasks, reducing the information redundancy among source prompts. Through a fusion module that integrates transformed task-agnostic embedding and task-specific embeddings, we achieve efficient prompt transfer. Extensive experimental validation demonstrates that this method further improves the prompt parameter performance of the prompt transfer method. Additionally, we analyze the two core sub-modules of PCT: singular value decomposition and contrastive learning. Lastly, we also test the performance of PCT under various scenarios.

Acknowledgments

Our work is supported by the National Natural Science Foundation of China (62476003), Anhui Province Excellent Scientific Research and Innovation Team (2024AH010004), Anhui Provincial Natural Science Foundation - Water Science Joint Fund (2408055US006), the University Synergy Innovation Program of Anhui Province (GXXT-2023-050), and SMP-Zhipu.AI Large Model Cross-Disciplinary Fund (SMP-Zhipu20240210). We also acknowledge the support from Zhipu AI-Anhui University Joint Research Center, and the High-Performance Computing Platform of Anhui University. Additionally, we would like to extend our gratitude to the TACL editors and the anonymous reviewers for their valuable feedback, which has significantly improved the quality of the paper.

References

Akari Asai, Mohammadreza Salehi, Matthew E. Peters, and Hannaneh Hajishirzi. 2022. ATTEMPT: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7–11, 2022*, pages 6655–6672. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.emnlp-main.446

Róbert Belanec, Simon Ostermann, Ivan Srba, and Mária Bieliková. 2024. Task prompt vectors: Effective initialization through multi-task soft-prompt transfer. *CoRR*, abs/2408.01119. https://doi.org/10.48550/ARXIV.2408.01119

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10–16, 2023*.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, Jing Yi, Weilin Zhao, Xiaozhi Wang, Zhiyuan Liu, Hai-Tao Zheng, Jianfei Chen, Yang Liu, Jie Tang, Juanzi Li, and Maosong Sun. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235. https://doi.org/10.1038/S42256-023-00626-4

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering, MRQA@EMNLP 2019, Hong Kong, China, November 4, 2019*, pages 1–13. Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-5801

Jinyang Guo, Jiaheng Liu, Zining Wang, Yuqing Ma, Ruihao Gong, Ke Xu, and Xianglong Liu. 2023. Adaptive contrastive knowledge distillation for BERT compression. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9–14, 2023*, pages 8941–8953. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.findings-acl.569

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 the 36th International Conference on Machine Learning, ICML 2019, 9–15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25–29, 2022*. OpenReview.net.

Ting Jiang, Jian Jiao, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Denvy Deng, and Qi Zhang. 2022. Promptbert: Improving BERT sentence embeddings with prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7–11, 2022*,

pages 8826–8837. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.emnlp-main.603

Minji Jung, Soyeon Park, Jeewoo Sul, and Yong Suk Choi. 2024. Is prompt transfer always effective? An empirical study of prompt transfer for question answering. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Short Papers, NAACL 2024, Mexico City, Mexico, June 16–21, 2024*, pages 528–539. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.naacl-short.44

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2–7, 2018*, pages 5189–5197. AAAI Press. https://doi.org/10.1609/aaai.v32i1.12022

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings*.

Pengxiang Lan, Enneng Yang, Yuting Liu, Guibing Guo, Linying Jiang, Jianzhe Zhao, and Xingwei Wang. 2024. Efficient prompt tuning by multi-space projection and prompt fusion. *CoRR*, abs/2405.11464. https://doi.org/10.48550/arxiv.2405.11464

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7–11 November, 2021*, pages 3045–3059. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.243

Junyi Li, Tianyi Tang, Jian-Yun Nie, Ji-Rong Wen, and Xin Zhao. 2022. Learning to transfer prompts for text generation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10–15, 2022*, pages 3506–3518. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.257

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1–6, 2021*, pages 4582–4597. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.353

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying lms with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6–11, 2021*, pages 5203–5212. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.410

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:140:1–140:67.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106. https://doi.org/10.1145/3474381

Yusheng Su, Xiaozhi Wang, Yujia Qin, Chi-Min Chan, Yankai Lin, Huadong Wang, Kaiyue Wen, Zhiyuan Liu, Peng Li, Juanzi Li, Lei Hou, Maosong Sun, and Jie Zhou. 2022. On transferability of prompt tuning for natural language processing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational*

Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10–15, 2022, pages 3949–3969. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.290

Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou', and Daniel Cer. 2022. Spot: Better frozen model adaptation through soft prompt transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22–27, 2022*, pages 5039–5059. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.346

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019a. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8–14, 2019, Vancouver, BC, Canada*, pages 3261–3275.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019b. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6–9, 2019*. OpenReview.net.

Chengyu Wang, Jianing Wang, Minghui Qiu, Jun Huang, and Ming Gao. 2021. Transprompt: Towards an automatic transferable prompting framework for few-shot text classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7–11 November, 2021*, pages 2792–2802. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.221

Zhen Wang, Rameswar Panda, Leonid Karlinsky, Rogério Feris, Huan Sun, and Yoon Kim. 2023. Multitask prompt tuning enables parameter-efficient transfer learning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1–5, 2023*. OpenReview.net.

Xinglong Wu, C. L. Philip Chen, Shuzhen Li, and Tony Zhang. 2024. Snapshot prompt ensemble for parameter-efficient soft prompt transfer. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2024, Seoul, Republic of Korea, April 14–19, 2024*, pages 11946–11950. IEEE. https://doi.org/10.1109/ICASSP48485.2024.10448070

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. 2022. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2022, Dublin, Ireland, May 22–27, 2022*, pages 1–9. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-short.1

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7–12, 2015, Montreal, Quebec, Canada*, pages 649–657.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2–7, 2019, Volume 1 (Long and Short Papers)*, pages 1298–1308. Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1131

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2024. Panda: Prompt transfer meets knowledge distillation for efficient model adaptation. *IEEE Transactions on Knowledge and Data Engineering*, 36(9):4835–4848. https://doi.org/10.1109/TKDE.2024.3376453