Unsupervised Few-shot Adaptation of Entailment Classifiers for Robust Natural Language Understanding

Anonymous ACL submission

Abstract

Although large-scale pretrained language models (LMs) have achieved significant improvements on different natural language tasks, their fine-tuning still heavily relies on task-specific data annotation and is sensitive to adversarial evaluation examples. In this work, we propose an entailment self-training framework for improving the accuracy and robustness of unsupervised few-shot task adaptations for language understanding without using any labeled data on the target tasks. We pretrain language models on the natural language inference (NLI) task, and adapt the model to new tasks with coordinated prompts and pseudo-labels. We find that the coordinated prompts, which jointly describe the task, serve as an equivalent dimension of the training data as human labels that enables learning. The proposed method enables task-specific fine-tuning without human-generated label. Experiments on the GLUE and AdvGLUE show that the coordinated prompts constantly outperform the no-prompt and single-prompt models under the unsupervised few-shot task adaptation setting. With preliminary logic pretraining on the entailment task and self-training, an unsupervised few-shot adapted medium LM can outperform existing few-shot, large-scale LMs.

1 Introduction

Although achieving state-of-the-art performance in different natural language understanding (NLU) tasks (Devlin et al., 2018; Liu et al., 2019; Yang et al., 2019; Clark et al., 2020; He et al., 2020; Joshi et al., 2020), large-scale pretrained language models still highly depend on human-annotated, task-specific training corpora for fine-tuning because the self-supervised pretraining objective does not incorporate explicit task-related knowledge. As a result, state-of-the-art language models are still challenged by lack of adequate fine-tuning data and difficult evaluation examples crafted by adversarial attacks or model-in-loop adversarial data annotations (Wang et al., 2021; Jin et al., 2020; Bartolo et al., 2020; Zang et al., 2019; Garg and Ramakrishnan, 2020; Li et al., 2020).

While humans can fully understand the nature of a machine learning task, the only way for a pretrained language model to understand a given task is fitting the training data. However, a given training corpus does not necessarily represent the ground-truth description of a task. For example, a sentiment analysis task for understanding audience attitudes toward movies (Socher et al., 2013) might be interpreted as trivially as "matching the keywords *good* and *bad*" by an overfit or unconverged model. Recent studies have been analyzing the difficulty by measuring the data distribution gap between different tasks, domains, and difficulties (Taori et al., 2020; Carlini et al., 2019; Miller et al., 2020; Koh et al., 2021). The different distribution within regular training data and adversarial evaluation data leads to the performance decrease under adversarial attack. Similarly, a training corpus consisting of a limited number of training examples is not enough for fully representing the distribution of the target task under the few-shot learning setting.

In this work, we show that preliminary logic pretraining and self-training based on coordinated task descriptions, namely coordinated prompts, can achieve better unsupervised NLU adaptation performance and robustness with medium-sized LMs than existing few-shot medium-sized and large-scale LMs (Gao et al., 2020; Schick and Schütze, 2021; Gu et al., 2021; Thoppilan et al., 2022; Wei et al., 2021). We first train an entailment classifier model on a natural language inference (NLI) (Williams et al., 2018) corpus, which allows the model to roughly learn if an input discourse is logically *true* or *false*. For any target tasks where a data distribution gap exists between the training and evaluation splits, we formulate each example in both splits as a logically *true* / *false* classification task to mitigate the distribution gap among the training, evaluation, and NLI corpora. This

formulation enables unsupervised task adaptation without any additional processing. Furthermore, the entailment model can be further improved with a limited number of task-specific unlabeled data examples.

We found that self-training coordinated task prompts with pseudo-labels generated by the entailment classifier itself can significantly outperform direct task adaptation. In some tasks, the unsupervised few-shot method is more effective than using human-generated labels given a limited number of task-specific data examples. A coordinated prompt consists of at least *two* logically cooperative suppositions that both describe a given task. One of the supposition being *true* or *false* entails that another supposition must be *true* or *false* accordingly because of logical restrictions. While no human-generated label is provided, we can annotate the fine-tuning examples with pseudo-labels predicted by the pretrained entailment classifier. The entailment classifier is then self-trained (Zoph et al., 2020; Zou et al., 2019) with the pseudo-labels, while the logical relations among the coordinated prompts are preserved. It is worth noting that the regular fine-tuning requires $N = |C_{train}|$ human-generated labels, while the label-free fine-tuning strategy we propose uses significantly fewer task prompts (at least 2) for each task.

Experiments show that the entailment classifiers trained with coordinated prompts can constantly achieve higher unsupervised adaptation performance than the training the models without task descriptions. Furthermore, the self-trained models with coordinated prompts can significantly improve the performance and robustness of the entailment model with a small number of unlabeled fine-tuning examples. The main contributions of this work are

- Enabling task-specific fine-tuning only with medium-sized LMs and small corpora, without human-generated labels, outperforming label-dependent few-shot methods and large-scale LMs.
- Reducing the cost of data collection, label annotation, and model training / inference.

2 Related Work

Pretraining large-scale neural language models in a self-supervised manner on large corpora and fine-tuning on task-specific training data has been a popular method recently for both language understanding (Devlin et al., 2018; Liu et al., 2019; Yang et al., 2019; Clark et al., 2020; He et al., 2020; Joshi et al., 2020) and generation (Brown et al., 2020; Lewis et al., 2019; Raffel et al., 2019; Zhang et al., 2019). Although achieving state-of-the-art performance in a wide range of tasks, recent studies have found that the pretraining-fine-tuning strategy relies on task-specific data annotation and the performance is sensitive to adversarial data examples (Blum and Mitchell, 1998; Wang et al., 2021; Jin et al., 2020; Bartolo et al., 2020; Zang et al., 2019; Garg and Ramakrishnan, 2020; Li et al., 2020).

Besides self-supervised pretraining, another strategy to improve models with unlabeled data is self-training (Zoph et al., 2020; Xie et al., 2020b; Noroozi et al., 2018; Zou et al., 2019; He et al., 2019; Sachan and Xing, 2018; Shakeri et al., 2020; Bartolo et al., 2021; Luo et al., 2021). Different from pretraining with data augmentation or constructing task-like pretraining cases (Glass et al., 2019), self-training models learn from synthetic task-specific training cases, which can be formulated as data-label pairs. (Zoph et al., 2020; Xie et al., 2020b; Noroozi et al., 2018; Zou et al., 2019) explored training with additional real data and pseudo-labels, while (He et al., 2019; Sachan and Xing, 2018; Shakeri et al., 2020; Bartolo et al., 2021; Luo et al., 2021) introduced training cases consist of both synthetic data and pseudo-labels. Lang et al. (2022) proposed training medium language models using pseudo-labels generated by large models, for example GPT-3 (Brown et al., 2020) and T0 (Sanh et al., 2021). These studies suggest that the self-training methods can improve the performance of weakly-supervised models.

Few-shot language understanding has been an important task for both research and application. Current solutions include multitask metric learning (Yu et al., 2018; Gu et al., 2018; Dou et al., 2019) and data augmentation (Xie et al., 2020a; Luo et al., 2021; Chen et al., 2020). Another line of research focuses on constructing task description prompts and utilizing the power of pretrained language models. Gao et al. (2020); Schick and Schütze (2021); Le Scao and Rush (2021) explored describing language tasks as cloze questions and applied masked language models to fill labeling words. Meng et al. (2020) proposed a self-training method that learns from abundant label descriptions. Obamuyide and Vlachos (2018); Yin et al. (2019) applied pretrained entailment classifiers for sequence and relation classification tasks. The mod-

els proposed by these studies can be classified into to major classes: input prompting and label interpretation. The input prompting method utilizes the ability of masked word prediction of the language models, while label interpretation method provides abundant explanations to labels with additional words or in natural language sentences. Both methods attempt to shift the data distribution from the newly given task to a task that a language model has been trained on.

3 Method

3.1 Supposition-based NLU

Entailment in NLP. The term "*entailment*" has been widely used in the area of natural language processing (NLP). Dzikovska et al. (2013); Williams et al. (2018) proposed textual entailment tasks, RTE and MNLI, that requires a model to predict if "text **x** *entails* text **y**" or not. It has been widely accepted by the NLP community that the entailment relation in these tasks is not a strictly defined logical entailment. Instead, "**x** *entails* **y**" is interpreted as "if **x** is true, then **y** is *likely* to be true." (Dzikovska et al., 2013). The vague definition makes it possible to crowd-source training corpora to train models that mimics human perceptions of text relations, for example, it is arguably acceptable that "She does not like the movie is *entailed* by her comment: the story is very boring" (Socher et al., 2013). The MNLI corpus has also been widely used as a secondary pretraining resource for other weakly-supervised text classification in many recent studies (Obamuyide and Vlachos, 2018; Yin et al., 2019; Lang et al., 2022).

Supposition Classification. Any language task has at least one textual task definition, which can be formulated as an affirmative or negated supposition $s_t^v = S(D_t, p_i, q_i, +/-)$ as shown in Table 1, where t is the task, D_t is the textual definition of task t, (p_i, q_i) stands for the $i-$th input example of task t, and $v \in \{+, -\}$ to indicate if s_t^v is affirmative or negated. Computing the truth values of the suppositions can imply the target outputs of all tasks. For example, if the affirmative supposition of the SST-2 task is predicted to be false by a model, then the predicted sentiment would be negative. In this work, we train models to predict the truth values of MNLI suppositions based on the human-annotated labels and adapt to other tasks with constructed suppositions shown in Table 1.

Although the definition of *entailment* in NLI tasks are sometimes vague, the *entailment* relations among constructed suppositions are strictly logical. If an affirmative supposition is *true*, than all affirmative suppositions constructed on the same input example must also be *true*, and all corresponding negated suppositions must be *false*. We have the following entailment relation between two suppositions s_t^{1,v_1}, s_t^{2,v_2}

$$E(s_t^{1,v_1}, s_t^{2,v_2}) = \begin{cases} s_t^{1,v_1} \rightarrow s_t^{2,v_2}, & v_1 = v_2 \\ s_t^{1,v_1} \not\rightarrow s_t^{2,v_2}, & v_1 \neq v_2 \end{cases}$$

Task	Inputs	Supposition
Affirmative Suppositions (s_t^+)		
MNLI	p, h	h is likely to be true when p is true.
RTE	p, h	h is likely to be true when p is true.
QNLI	t, q	question q can be answered by text t.
QQP	q_1, q_2	q_1 and q_2 are duplicated questions.
SST-2	x	Her attitude towards the movie is positive given her comment x.
CoLA	t	sentence t is fluent.
Negated Suppositions (s_t^-)		
MNLI	p, h	h cannot be true when p is true.
RTE	p, h	h cannot be true when p is true.
QNLI	t, q	text t cannot answer question q.
QQP	q_1, q_2	q_1 and q_2 are different questions.
SST-2	x	Her attitude towards the movie is not positive given her comment x.
CoLA	t	The grammar of t cannot be accepted.

Table 1: The suppositions constructed based on the definitions of different GLUE tasks (Wang et al., 2018).

3.2 Prompt Coordination

A model trained with a training corpus C_{train} without explicitly encoding the task definition attempts to learn implicit task suppositions s_t^* implied by C_{train}, by mapping training data to human-generated labels. When C_{train} is limited in terms of corpus size and data distribution, s_t^* might be biased from the ground-truth task definition, leading to poor validation performance.

On the other hand, prompt-tuning methods construct textual task descriptions, sometimes as cloze questions. A language model trained with such methods can avoid being too biased away from the ground-truth task definition, since it is already provided as a prompt. However, the problem cannot be completely eliminated because the model can shift the semantics of the template words in the prompts, since they appears in all training cases and their word embeddings might be significantly changed during training.

We construct a prompt coordination with two logically cooperative suppositions for each task.

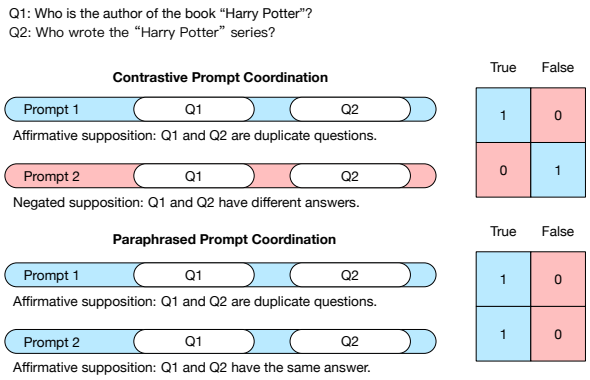

Figure 1: Examples of contrastive and paraphrased prompt coordinations of the QQP task.

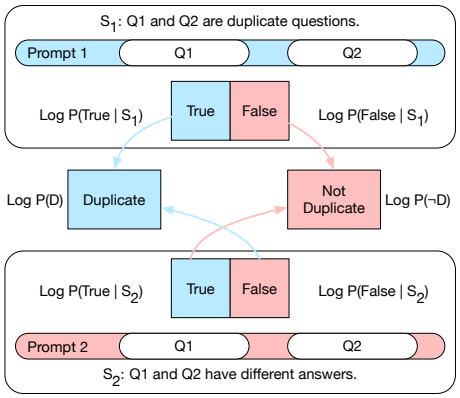

Figure 2: Example of the label mapping of a prompt coordination in the QQP task.

For example, we can construct a contrastive prompt coordination with one affirmative and one negated supposition, or a paraphrased prompt coordination with two affirmative suppositions. The selected suppositions must have different or same truth values because of the logical presupposition of the constructions. The model for predicting the truth values can be trained on the corpora of such constructions. The constructed prompts and corresponding truth values are shown in Figure 1.

3.3 Self-training

Under the label-free setting, we fine-tune the language models using pseudo-labels generated by the entailment classifier. Since the entailment classifier is trained to learn three relations, *true*, *false*, and *neutral*, we drop the predicted scores of the *neutral* class while generating pseudo-labels for binary classification tasks.

pseudo-label generation. In this work, we construct two suppositions, $s_{t,i}^{1,v_1}$ and $s_{t,i}^{2,v_2}$ for each data example. Each supposition can be either affirmative or negated. We calculate the pseudo-label for data example $d_i = (p_i, q_i)$

$$l_i^1 = m(s_{t,i}^{1,v_1}),\ l_i^2 = \begin{cases} l_i^1, & v_1 = v_2 \\ 1 - l_i^1, & v_1 \neq v_2 \end{cases}$$

where m is the pretrained entailment classifier, $s_{t,i}^{j,v_j} = S(D_t, p_i, q_i, +/-)$ can be either affirmative or negated. As a result, we can construct two supposition-label pairs $\{(s_{t,i}^{1,v_1}, l_i^1), (s_{t,i}^{2,v_2}, l_i^2)\}$ for each data example. We can use the constructed data for both fine-tuning and evaluation. We call the constructed supposition pair $(s_{t,i}^{1,v_1}, s_{t,i}^{2,v_2})$ a prompt coordination. The pair is a paraphrased prompt coordination if $v_1 = v_2$, otherwise a contrastive prompt coordination. For any dataset $D_t = \{(p_i, q_i) \mid i \geq 0\}$, we can construct a new dataset consisting of supposition-pseudo-label pairs,

$$D_t' = \{(S_{t,i}^{j,v_j}, l_i^j) \mid i \geq 0, j \in [0, k-1]\} \quad (1)$$

where k is the number of suppositions constructed for each data example. We apply $k = 2$ in this work. The model is trained with the constructed dataset in the same way as fine-tuning with human-generated labels.

Confidence-based labeling. Lang et al. (2022) found that dropping data examples with unconfident pseudo-labels benefits the performance of the self-trained DeBERTa (He et al., 2020) model. In this work, we explore the effect of the following labeling strategies: (1) using all pseudo-labels and (2) dropping unconfident pseudo-labels.

We measure the confidence of a prediction with its probability $P(l_i^*|p_i, q_i)$ output by the entailment classifier. We sort all data examples based on the confidence of corresponding pseudo-labels, and drop the most unsure predictions. We will compare the performance of using all pseudo-labels and the confidence-based labeling strategies.

3.4 Inference

The final prediction of a task given a data example is entailed by the truth value of the constructed supposition. As a result, we can build a mapping between the truth values of the suppositions and the final task predictions according to the task definition. An example of the label mapping in the QQP task is shown in Figure 2. In this example, we have the following entailment relations,

$$(s_1 = \mathbf{T} \leftrightarrow s_2 = \mathbf{F}) \rightarrow D$$
$$(s_1 = \mathbf{F} \leftrightarrow s_2 = \mathbf{T}) \rightarrow \neg D$$

where $\mathbf{T}, \mathbf{F}$ stand for True and False, and D is the following proposition, "the label of example (Q_1, Q_2) is *duplicate*". As a result, we can calculate the probability of final labels by

$$logP(D|Q_1, Q_2) = logP(\mathbf{T}|s_1) + logP(\mathbf{F}|s_2)$$

$$logP(\neg D|Q_1, Q_2) = logP(\mathbf{F}|s_1) + logP(\mathbf{T}|s_2)$$

4 Experiments

Datasets. We conduct experiments on popular natural language understanding tasks in the GLUE (Wang et al., 2018) benchmark, including RTE (Dagan et al., 2005), QNLI (Rajpurkar et al., 2016), QQP, SST-2 (Socher et al., 2013), and CoLA (Warstadt et al., 2019). We also assess the robustness of the proposed method against adversarial evaluation sets in the AdvGLUE corpus (Wang et al., 2021), including Adv-RTE, Adv-QNLI, Adv-QQP, and Adv-SST2. The data in AdvGLUE is created by adding word-level and sentence-level perturbation to the GLUE data, as well as human-crafted examples. More details is in Appendix A.

Hyper-parameters. We train BERT (Devlin et al., 2018) and DeBERTa (He et al., 2020) models for the language understanding tasks, without using larger language models like GPT-3 (Brown et al., 2020) and T0 (Sanh et al., 2021) that are used for generating pseudo-labels in (Lang et al., 2022). We also use the same hyper-parameters across all tasks, attempting to avoid the problems mentioned in Perez et al. (2021). In the entailment pretraining on the MNLI dataset (Williams et al., 2018), we optimize both BERT and DeBERTa models with the AdamW optimizer (Loshchilov and Hutter, 2018). For all tasks and both models, we set $\varepsilon = 10^{-6}$. In the entailment pretraining, we set the weight decay weight to 10^{-5}, and the learning rates for BERT and DeBERTa are 5e-6 and 3e-6 respectively. During the self-training step, the learning rate of both models on all tasks is 4e-6, and the weight decay weight is constantly 10^{-2}. We run the entailment pretraining for 2 epochs and the self-training for 6 epochs. In confidence-based labeling, we drop 1/8 data with lowest confidence.

Self-training details. We train `BERT-large` and `DeBERTa-large` sequence classification models provided by Huggingface (Wolf et al., 2020). Both models contain a pretrained language model of 350M trainable parameters and a linear classification head. For each data example in a target task, we construct 2 suppositions and generate 1 hard pseudo-label for each supposition. We randomly shuffle the constructed datasets before feeding them to the entailment classifiers for fine-tuning. For each task, we randomly select $N = 12 \cdot n, n \in [1, 10]$ unlabeled data examples. We train and evaluate the models for each N with 3 independent runs and calculate the average performance on 2 V100 32G GPUs. Our implementation will be open-sourced at `https://github.com/xxx/xxx`.

Baselines. We mainly compare our method with the following baselines,

- Direct zero-shot adaptation of pretrained BERT and DeBERTa entailment classifiers with and without supposition construction.
- Fine-tuned entailment classifiers with human-generated labels on each tasks under both fully supervised and few-shot settings.
- Few-shot medium-sized language models trained with task-specifc labels.
- Zero- and Few-shot large-scale language models, including LaMDA-137B (Thoppilan et al., 2022) and FLAN-137B (Wei et al., 2021).

4.1 Results

Direct zero-shot adaptation. We first evaluate the performance of directly applying pretrained entailment classifiers on different tasks. We compare models trained with or without supposition constructions. The performance is shown in Table 2.

Tasks	RTE	QNLI	QQP	SST2	CoLA
Conc.	74.01	71.61	70.53	84.63	65.77
Supp.	84.48	78.74	79.99	90.14	59.78
Adversarial Evaluation Sets					
Conc.	50.62	60.81	47.44	56.08	N/A
Supp.	67.9	64.86	64.10	50.68	

Table 2: Direct zero-shot adaptation of entailment classifiers using different data formatting strategies. Conc. stands for directly concatenating the input texts, and Supp. stands for constructing suppositions in natural language and predict the truth values.

The improvement of supposition construction is significant on the RTE, QNLI, QQP tasks and their adversarial versions, while the performance gap on SST-2 and CoLA is smaller. Since SST-2 and CoLA are single-sentence tasks, we still need to add label descriptions as additional inputs to run the entailment classifier. The label descriptions can provide additional information as prompts, and

Method	GLUE						Method	AdvGLUE				
	QNLI	QQP	RTE	SST2	CoLA	Avg.		QNLI	QQP	RTE	SST2	Avg.
Few-shot (left) and supervised (right) medium LMs (350M) with human-generated labels												
PET	61.3	67.6	65.7	91.8	-	71.6	R3F	47.5	40.6	50.1	38.5	44.2
LM-BFF	69.2	69.8	73.9	**93.0**	21.8	65.4	CT_T	49.6	40.7	46.2	39.2	43.9
P-tuning	68.8	67.6	70.8	92.6	60.6	72.1	MT	47.5	41.5	52.5	51.3	48.2
PPT	68.8	67.2	67.9	92.3	-	74.1	BERT	39.8	37.9	40.5	33.0	37.8
UPT	70.1	72.1	68.9	92.9	-	76.0	DeBERTa	57.9	60.4	**79.0**	**57.8**	63.8
Few-shot fine-tuning medium entailment LMs (350M) with human-generated labels												
$BERT^{10k}$	58.7	59.6	65.7	74.3	67.4	65.1	/	60.3	42.7	34.6	31.5	42.3
$DeBERTa^{1k}$	73.4	78.3	76.8	88.7	65.6	76.6	/	62.4	52.6	65.4	49.1	57.4
$BERT^{all}$	61.5	65.9	76.4	78.4	68.0	66.8	/	61.7	53.4	49.4	39.9	51.1
$DeBERTa^{all}$	80.2	77.4	88.0	87.1	64.9	77.8	/	67.8	70.5	70.8	49.1	64.6
Few-shot large LMs (137B) with human-generated labels												
LaMDA	55.7	58.9	70.8	92.3	-	69.4	/	-	-	-	-	-
FLAN	63.3	75.9	84.5	94.6	-	79.6	/	-	-	-	-	-
Self-trained medium LMs (350M) without human-generated labels												
$BERT^{10k}$	70.8	71.6	69.2	81.7	59.5	70.5	/	62.8	64.5	58.4	55.2	60.2
$DeBERTa^{1k}$	76.7	76.5	75.8	87.5	**69.3**	77.2	/	71.2	73.1	66.3	54.3	66.2
$BERT^{all}$	60.0	73.9	78.5	86.9	61.8	72.2	/	55.6	58.9	64.2	56.5	58.8
$DeBERTa^{all}$	**83.5**	**80.8**	**86.0**	92.7	69.2	**82.5**	/	**73.4**	**72.7**	70.4	56.1	**68.1**

Table 3: Comparing our unsupervised few-shot method with supervised few-shot baselines on GLUE/AdvGLUE dev sets. PET, LM-BFF, P-tuning, PPT, and UPT use 16 data-label pairs, LaMDA uses 5, and FLAN uses up to 12 labeled examples. The self-trained models are tuned on 24 unlabeled examples, and the $DeBERTa^{all}$ models use 12 unlabeled training examples. R3F, CT_T, and MT are fully supervised methods for robust language understanding. $1k, 10k$, and all stands for the number of MNLI training examples used for the training the entailment classifiers.

reduce the performance gap. The suppositions we used and more results are shown in Appendix B.

Self-training. We compare our method with recent, strong baseline models on few-shot and robust language understanding respectively. It worth noting that baseline methods requires $K = |C_{train}|$ human generated labels, while our models uses just 2 suppositions and no human-generated labels. The experiment results are shown in table 3. We found that our self-trained DeBERTa model without using any human-generated labels performs best in terms of the averaged accuracy, and the self-trained BERT achieves similar performance as the strong baseline models. We compare our model with PET (Schick and Schütze, 2021), LM-BFF (Gao et al., 2020), P-tuning (Liu et al., 2021), PPT (Gu et al., 2021), and UPT (Wang et al., 2022), which are recent strong baselines for few-shot language understanding on the GLUE benchmark. For AdvGLUE, we compare with R3F (Aghajanyan et al., 2020), child tuning (CT) (Xu et al., 2021), and match tuning (MT) (Tong et al., 2022) models. We also compare the performance of fully supervised and few-shot BERT and DeBERTa classifiers. For the baselines that did not report the CoLA performance, we calculate the averaged performance on the other tasks.

Table 3 shows that the self-trained BERT model achieves competitive performance as few-shot learning models without using any human-generated label, and the self-trained DeBERTa-350M model outperforms both large-scale language models with 137B parameters. on the regular GLUE benchmark. We noticed that the self-training model perform well on the CoLA task, although it is not a pure semantic reasoning task, outperforming LM-BFF by 47% and P-tuning by over 8% (absolute), approaching the fully supervised DeBERTa performance (69.3% vs 70.5%) (He et al., 2020). The self-trained models also outperform fully supervised methods on AdvGLUE. For comparison the performance of R3F, CT_T, and MT reported in Tong et al. (2022), and the fully supervised BERT and DeBERTa performance in Wang et al. (2021). The self-trained BERT model outperforms all BERT-based baselines (R3F, CT_T, MT), and the self-trained DeBERTa models achieved the highest averaged accuracy and outperforms the fully supervised DeBERTa by 4.8%.

We also evaluated $BERT^{10k}$ and $DeBERTa^{1k}$ classifiers, which are trained on $10k$ and $1k$ MNLI examples respectively, while $BERT^{all}$ and $DeBERTa^{all}$ use all $390k$ MNLI training data. The experiment results shows that training the entailment classifiers on a small subset of the MNLI training corpus also leads to significant improvement over the label-dependent fine-tuning baselines.

Method	GLUE						AdvGLUE				
	QNLI	QQP	RTE	SST2	CoLA	Avg.	QNLI	QQP	RTE	SST2	Avg.
Few-shot large LMs (137B) with human-generated labels											
LaMDA	50.6	34.9	73.3	51.0	-	52.45	-	-	-	-	-
FLAN	59.6	72.1	78.3	94.6	-	76.15	-	-	-	-	-
Self-trained BERT-340M without human-generated labels											
No self-training	58.14	71.25	76.53	85.32	52.63	68.77	59.49	53.85	56.91	56.08	56.58
Single prompt	55.87	71.64	74.24	85.13	52.45	67.87	52.93	47.86	44.03	33.10	44.48
w/ CL	56.03	70.86	76.35	84.71	56.95	68.98	54.73	47.44	43.61	61.78	51.89
Contrastive Coord.	59.89	**73.87**	78.10	85.09	52.99	68.99	59.91	48.72	47.73	53.15	52.38
w/ CL	59.22	73.24	**78.46**	85.51	56.30	70.55	59.23	**55.55**	46.91	**64.19**	56.47
Paraphrased Coord.	**60.02**	71.79	76.77	86.20	55.56	70.25	53.60	49.15	**58.86**	36.26	49.47
w/ CL	55.35	71.86	77.50	**86.85**	**61.84**	**70.68**	**60.14**	49.57	57.20	62.99	**57.48**
Self-trained DeBERTa-350M without human-generated labels											
No self-training	77.27	79.23	84.48	90.14	59.73	78.17	64.86	66.67	70.37	50.68	63.15
Single prompt	81.12	79.69	85.56	90.64	60.03	79.41	65.77	64.53	67.49	**59.66**	64.36
w/ CL	81.20	79.62	80.98	91.70	67.24	80.15	68.02	68.02	67.90	49.10	63.26
Contrastive Coord.	79.88	**80.93**	**86.04**	92.35	68.62	81.56	**73.42**	**71.79**	70.37	53.38	67.24
w/ CL	82.43	80.68	83.27	**92.74**	**69.22**	**81.67**	70.49	70.49	71.19	52.70	62.22
Paraphrased Coord.	**83.50**	80.37	84.96	90.56	66.67	81.21	72.75	70.94	71.18	58.56	**68.38**
w/ CL	81.68	80.41	83.61	90.60	69.03	81.07	66.67	66.67	**72.84**	50.90	64.27

Table 4: Unsupervised adaptation performance of different strategies on GLUE and AdvGLUE corpora using 24 unlabeld data examples. The BERT and DeBERTa models are fine-tuned on the full MNLI training corpus. The performance of LaMDA (Thoppilan et al., 2022) and FLAN was reported in Wei et al. (2021). In self-training experiments, CL stands for applying the confidence-based labeling strategy. For each model and prompting strategy, the training processes on all tasks apply the same hyper-parameters.

4.2 Analysis

Effect of prompt coordination. We analyze the performance of the self-training method with DeBERTa and BERT by fine-tuning on 24 unlabeled data examples. The results on regular and adversarial evaluation sets are shown in Table 4. Both models are self-trained on 24 unlabeled data examples and results are calculated by averaging three independent experiments. The experiment results show that the self-trained entailment classifiers with 350M parameters can significantly outperform large-scale language models with 137B parameters. The DeBERTa model outperforms LaMDA (Thoppilan et al., 2022) by 29.04% and FLAN (Wei et al., 2021) by 5.34% (absolute). The proposed prompt coordination mechanism can improve the performance of prompt tuning with pseudo-labels on both regular and adversarial evaluation sets and constantly outperforms the single-prompt models (ST-SP) with both BERT and DeBERTa.

We found that different models, tasks can be benefited with either contrastive coordination (CC) or paraphrased coordination (PC). The average performance gap of CC and PC is less that 1% on the regular GLUE benchmark, but the performance gap on the AdvGLUE benchmark is more significant (>1%), suggesting that the PC benefits the self-training robustness against adversarial examples. Our experiment results also supports the finding in Lang et al. (2022) that the confidence-based labeling (CL) strategy improves the self-training performance. Table 4 shows that applying CL outperforms the base ST-SP method on GLUE with both BERT and DeBERTa models. However, CL does not improve the performance on AdvGLUE with DeBERTa. In the prompt coordination methods, the effect of CL is not constantly positive. We found that the accuracy of BERT+AdvGLUE and DeBERTa+GLUE can be improved by CL, while prompt coordination without CL works better on BERT+GLUE and DeBERTa+AdvGLUE. This suggests that the prompt coordination mechanism is more robust against noisy pseudo-labels.

Effect of data size. We have shown that the label-free self-training method achieves equivalent or better performance on different tasks with different language models, and we would like to answer the question: how many labels is enough for a fine-tuned model to outperform a self-trained model? To understand this, we conduct experiments on 10 different data size settings, $N \in \{12, 24, 36, 48, 60, 72, 84, 96, 108, 120\}$. We run 3 independent experiments for each setting with the DeBERTa model.

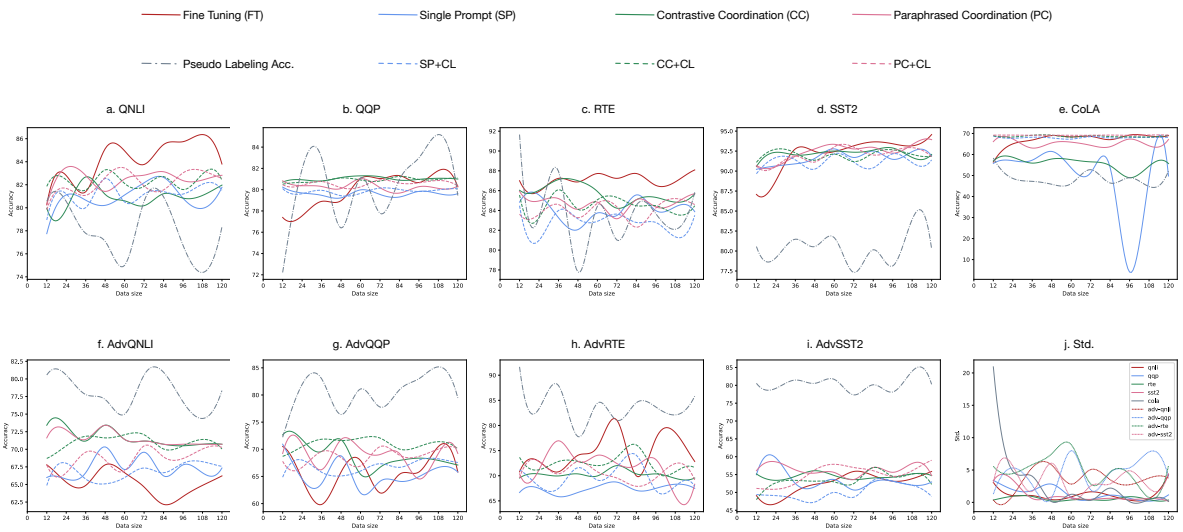

Figure 3: The comparison different fine-tuning and self-training methods, and the confidence-based labeling (CL) strategy with the DeBERTa model on different sizes of training data. Figures a to i illustrate the performance on GLUE and AdvGLUE tasks, and Figure j shows the standard deviation of the fine-tuning performance.

The experiment results are illustrated in Figure 3. Figures 3.a to j summarize the performance of different tuning methods, including fine-tuning and self-training, with the DeBERTa model using training corpora of different sizes. On the GLUE tasks, we noticed a trend that when the number of training examples is more than 48, the fine-tuning performance based on human-generated labels is better than label-free, self-training methods. On the AdvGLUE tasks, the experiment results show that fine-tuning is not significantly better than label-free self-training on the given training corpora. Figure 3.j shows the standard deviation (Std) of the fine-tuning performance. The Std of fine-tuning performance on GLUE tasks with smaller trainer corpora is significantly higher than larger corpora. These facts suggest that when the size of human-labeled training data is not enough, the coordinated prompts represent the given tasks better than the data-label pairs. On AdvGLUE tasks, increasing the size of training corpora does not ensure improved representation of the adversarial tasks, thus we do not observe an obvious change of performance Std when more data examples are processed.

Although Figure 3 shows that contrastive and paraphrased coordinations (CC and PC), and the effect of confidence-based labeling (CL) strategy contribute to each task differently, their performance gaps vary when the size of the unlabeled training data increases. When the data is not enough, the performance gap between CC and PC, CL and no CL is significant. However, when the models are self-trained with more data, the difference becomes less obvious. We also found that the CL mechanism smooth the performance on the CoLA task.

Effect of pseudo-labeling accuracy. From Figure 3, we found that there is no necessary relation between the accuracy of pseudo-labeling and the evaluation performance. The QNLI and SST2 tasks show that noisy pseudo-labels can lead to significantly higher evaluation performance, while the AdvGLUE tasks suggest that training with correct labels does not necessarily improve the robustness against adversarial data.

5 Conclusion

We propose a self-training, prompt coordination method to adapt pretrained entailment classifiers to different natural language understanding tasks by predicting the truth values of constructed task suppositions. By learning to recognize the logical relation between different suppositions, a medium-sized language entailment classifier can outperform zero- and few-shot, medium and large-scale pre-trained language models without using any human-generated labels on natural language understanding tasks. We also found that the coordinated prompts can significantly improve the the robustness of language models against adversarial evaluation examples. Our results indicate that a preliminary logical learning can significantly improve the efficiency of language model training, by reducing the need of data collection, human annotation, and the model size for achieving the comparable performance.

Limitations

Our method utilized a pretrained MNLI model and adapt it to other domains under few-shot unsupervised settings. There are two limitations that we would like to improve in future work. Firstly, we use human-designed suppositions for each task, which is less automatic than direct adaptation of the models. Secondly, as shown in Fig 3, the self-training performance is not as good as fine-tuning with human-generated labels when the number of training examples increases because the noisy pseudo label set. We would like to overcome this in the next step.

Ethics Statement

We propose a method that can significantly reduce the financial and environmental cost of language model learning. By reducing the need of data collection and human labeling, our method can effectively protect user and data privacy by avoiding leaking any information while building the training corpora. We found that a medium sized language model can achieve similar performance as the state-of-the-art large-scale language models, suggesting that we can cost less financially and environmentally during model training and evaluation for comparable performance. However, since we reduced the need of human-labeling efforts, the deployment of the system might decrease the number of data annotation jobs.

References

Armen Aghajanyan, Akshat Shrivastava, Anchit Gupta, Naman Goyal, Luke Zettlemoyer, and Sonal Gupta. 2020. Better fine-tuning by reducing representational collapse. In *International Conference on Learning Representations*.

Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. 2020. Beat the ai: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021. Improving question answering model robustness with synthetic adversarial data generation. *arXiv preprint arXiv:2104.08678*.

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100.

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.

Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. 2019. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine learning challenges workshop*, pages 177–190. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Zi-Yi Dou, Keyi Yu, and Antonios Anastasopoulos. 2019. Investigating meta-learning algorithms for low-resource natural language understanding tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1192–1197, Hong Kong, China. Association for Computational Linguistics.

Myroslava Dzikovska, Rodney Nielsen, Chris Brew, Claudia Leacock, Danilo Giampiccolo, Luisa Bentivogli, Peter Clark, Ido Dagan, and Hoa Trang Dang. 2013. SemEval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 263–274, Atlanta, Georgia, USA. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Siddhant Garg and Goutham Ramakrishnan. 2020. Bae: Bert-based adversarial examples for text classification. *arXiv preprint arXiv:2004.01970*.

Michael Glass, Alfio Gliozzo, Rishav Chakravarti, Anthony Ferritto, Lin Pan, GP Bhargav, Dinesh Garg, and Avirup Sil. 2019. Span selection pre-training for question answering. *arXiv preprint arXiv:1909.04120*.

Jiatao Gu, Yong Wang, Yun Chen, Victor O. K. Li, and Kyunghyun Cho. 2018. Meta-learning for low-resource neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3622–3631, Brussels, Belgium. Association for Computational Linguistics.

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

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2019. Revisiting self-training for neural sequence generation. *arXiv preprint arXiv:1909.13788*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. 2021. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR.

Hunter Lang, Monica Agrawal, Yoon Kim, and David Sontag. 2022. Co-training improves prompt-based learning for large language models. *arXiv preprint arXiv:2202.00828*.

Teven Le Scao and Alexander M Rush. 2021. How many data points is a prompt worth? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2627–2636.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. Bert-attack: Adversarial attack against bert using bert. *arXiv preprint arXiv:2004.09984*.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021. Gpt understands, too. *arXiv preprint arXiv:2103.10385*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Hongyin Luo, Shang-Wen Li, Seunghak Yu, and James Glass. 2021. Cooperative learning of zero-shot machine reading comprehension. *arXiv preprint arXiv:2103.07449*.

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. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9006–9017.

John Miller, Karl Krauth, Benjamin Recht, and Ludwig Schmidt. 2020. The effect of natural distribution shift on question answering models. In *International Conference on Machine Learning*, pages 6905–6916. PMLR.

Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. 2018. Boosting self-supervised learning via knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9359–9367.

Abiola Obamuyide and Andreas Vlachos. 2018. Zero-shot relation classification as textual entailment. *EMNLP 2018*, page 72.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *Advances in Neural Information Processing Systems*, 34:11054–11070.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Mrinmaya Sachan and Eric Xing. 2018. Self-training for jointly learning to ask and answer questions. In *Proceedings of the 2018 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 629–640.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

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.

Siamak Shakeri, Cicero Nogueira dos Santos, Henry Zhu, Patrick Ng, Feng Nan, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. End-to-end synthetic data generation for domain adaptation of question answering systems. *arXiv preprint arXiv:2010.06028*.

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.

Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. 2020. Measuring robustness to natural distribution shifts in image classification. *Advances in Neural Information Processing Systems*, 33:18583–18599.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Shoujie Tong, Qingxiu Dong, Damai Dai, Tianyu Liu, Baobao Chang, Zhifang Sui, et al. 2022. Robust fine-tuning via perturbation and interpolation from in-batch instances. *arXiv preprint arXiv:2205.00633*.

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*.

Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. 2021. Adversarial glue: A multi-task benchmark for robustness evaluation of language models. *arXiv preprint arXiv:2111.02840*.

Jianing Wang, Chengyu Wang, Fuli Luo, Chuanqi Tan, Minghui Qiu, Fei Yang, Qiuhui Shi, Songfang Huang, and Ming Gao. 2022. Towards unified prompt tuning for few-shot text classification. *arXiv preprint arXiv:2205.05313*.

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

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

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

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020a. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33:6256–6268.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. 2020b. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698.

Runxin Xu, Fuli Luo, Zhiyuan Zhang, Chuanqi Tan, Baobao Chang, Songfang Huang, and Fei Huang. 2021. Raise a child in large language model: Towards effective and generalizable fine-tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9514–9528.

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, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3914–3923, Hong Kong, China. Association for Computational Linguistics.

Mo Yu, Xiaoxiao Guo, Jinfeng Yi, Shiyu Chang, Saloni Potdar, Yu Cheng, Gerald Tesauro, Haoyu Wang, and Bowen Zhou. 2018. Diverse few-shot text classification with multiple metrics. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1206–1215.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2019. Word-level textual adversarial attacking as combinatorial optimization. *arXiv preprint arXiv:1910.12196*.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*.

Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. 2020. Rethinking pre-training and self-training. *Advances in neural information processing systems*, 33:3833–3845.

Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. 2019. Confidence regularized self-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5982–5991.

A Data Details

In this work, we evaluate our method with the GLUE[1] and AdvGLUE[2] benchmarks. We pretrain our models on MNLI, and evaluate on all other AdvGLUE tasks, AdvQNLI, AdvQQP, AdvRTE, and AdvSST2. we also evaluate the models on the regular versions of these tasks in GLUE, plus CoLA, a non-semantic reasoning task. The statistics of the GLUE and AdvGLUE benchmarks are shown in Table 5.

Corpus	|Train|	|Test|	|Adv-Test|
MNLI	393k	20k	1.8k
QNLI	105k	5.4k	0.9k
QQP	364k	391k	0.4k
RTE	2.5k	3k	0.3k
SST2	67k	1.8k	1.4k
CoLA	8.5k	1k	-

Table 5: Statistics of the corpora used in this work

[1] https://gluebenchmark.com/
[2] https://adversarialglue.github.io/

B Constructed Suppositions

In this section, we show the suppositions we applied and present the corresponding unsupervised adaptation performance. For each task, we use three suppositions, s_0^+, s_1^-, and s_2^+, where (s_0^+, s_1^-) is a pair of contrastive prompt coordination, and (s_0^+, s_2^+) is a paraphrased prompt coordination. The prompts we used are list as follows,

- **MNLI**
  - s_0^+: {x} is entailed by {y}.
  - s_1^-: {x} cannot be true when {y} is true.
  - s_2^+: {x} is true when {y} is true.
- **QNLI**
  - s_0^+: The answer to {x} is entailed by {y}.
  - s_1^-: {x} cannot be answered by {y}.
  - s_2^+: {x} can be answered by {y}.
- **QQP**
  - s_0^+: The answer to {x} is entailed by the answer to {y}.
  - s_1^-: {x} and {y} are not the same questions.
  - s_2^+: {x} and {y} are same questions.
- **RTE**
  - s_0^+: {x} is entailed by {y}.
  - s_1^-: {x} cannot be true when {y} is true.
  - s_2^+: {x} is true when {y} is true.
- **SST-2**
  - s_0^+: The movie is good is entailed by {x}.
  - s_1^-: I like the movie cannot be entailed by the comment {x}.
  - s_2^+: I like the movie is entailed by the comment {x}.
- **CoLA**
  - s_0^+: The sentence {x} is fluent.
  - s_1^-: The grammar of {x} cannot be accepted.
  - s_2^+: The grammar of {x} can be accepted..

The zero-shot adaptation results of these prompts on different tasks are shown in Table 6, 7, 8, and 9. In our experiments, we use s_0 for pseudo label generation, and evaluate the prompt coordination performance of (s_0^+, s_1^-) and (s_0^+, s_2^+).

The experiment results show that the DeBERTa^{1k} model outperforms the BERT^{10k}

Task	Prompt	GLUE	AdvGLUE
QNLI	s_0^+	77.27	61.49
	s_1^-	72.14	71.62
	s_2^+	65.86	68.24
QQP	s_0^+	79.99	64.10
	s_1^-	77.36	56.41
	s_2^+	68.53	55.13
RTE	s_0^+	84.48	66.67
	s_1^-	84.48	69.14
	s_2^+	53.06	40.74
SST-2	s_0^+	90.14	42.57
	s_1^-	70.99	40.54
	s_2^+	85.78	47.30
CoLA	s_0^+	59.73	-
	s_1^-	32.02	-
	s_2^+	58.87	-

Table 6: Zero-shot performance of different prompts on each task with the DeBERTa^{all} model.

Task	Prompt	GLUE	AdvGLUE
QNLI	s_0^+	72.23	56.08
	s_1^-	57.04	52.03
	s_2^+	70.82	64.19
QQP	s_0^+	71.63	69.23
	s_1^-	65.86	64.10
	s_2^+	46.01	41.03
RTE	s_0^+	73.65	64.20
	s_1^-	67.51	61.73
	s_2^+	53.06	40.74
SST-2	s_0^+	81.42	47.30
	s_1^-	54.01	49.32
	s_2^+	79.01	52.03
CoLA	s_0^+	65.20	-
	s_1^-	57.62	-
	s_2^+	48.32	-

Table 7: Zero-shot performance of different prompts on each task with the DeBERTa^{1k} model.

Task	Prompt	GLUE	AdvGLUE
QNLI	s_0^+	58.14	59.46
	s_1^-	57.61	54.73
	s_2^+	58.14	62.84
QQP	s_0^+	71.25	50.00
	s_1^-	58.48	53.85
	s_2^+	65.02	48.72
RTE	s_0^+	76.53	44.44
	s_1^-	78.34	46.91
	s_2^+	22.38	58.02
SST-2	s_0^+	85.32	32.43
	s_1^-	49.08	47.56
	s_2^+	81.18	33.11
CoLA	s_0^+	52.63	-
	s_1^-	47.56	-
	s_2^+	66.83	-

Table 8: Zero-shot performance of different prompts on each task with the BERT^{all} model.

Task	Prompt	GLUE	AdvGLUE
QNLI	s_0^+	65.93	62.84
	s_1^-	49.99	47.30
	s_2^+	58.81	48.65
QQP	s_0^+	63.71	42.31
	s_1^-	71.30	37.18
	s_2^+	31.34	69.23
RTE	s_0^+	68.23	34.57
	s_1^-	67.87	32.10
	s_2^+	31.05	61.73
SST-2	s_0^+	80.96	24.32
	s_1^-	50.57	52.03
	s_2^+	80.05	30.41
CoLA	s_0^+	40.17	-
	s_1^-	49.57	-
	s_2^+	42.47	-

Table 9: Zero-shot performance of different prompts on each task with the BERT^{10k} model.

model on several tasks and suppositions, suggesting that the BERT model is more likely to overfit to the supposition templates during the training. Similar conclusions are also implied by the self-training experiment results.

C Entailment Pretraining Details

The backbone model of our method is an entailment classifier that reasonably understands both affirmative and negated suppositions. However, if we only pretrain the entailment model with one affirmative prompt, the model will overfit to the given prompt template and cannot perform well on other prompts. We evaluate the performance of unsupervised adaptation using the negated suppositions with an entailment model pretrained on the MNLI corpus only with s_0^+. The experiment results are shown in Table 10.

Task	Prompt	GLUE	AdvGLUE
QNLI	s_1^-	49.19	49.32
QQP	s_1^-	36.54	36.78
RTE	s_1^-	15.52	28.40
SST-2	s_1^-	48.74	50.00
CoLA	s_1^-	32.02	-

Table 10: Zero-shot performance of different prompts on each task based on an entailment classifier trained with a single prompt.

The zero-shot adaptation performance of the negated suppositions significantly dropped in Table 10. To solve this problem, we train the entailment classifiers on MNLI using the contrastive prompt coordination (s_0^+, s_1^-), which achieves the perfor-

mance shown in Table 6.

D BERT^{10k} and DeBERTa^{1k} Results

We showed a subset of experiment results of BERT^{10k} and DeBERTa^{1k} in Table 3. Here we report the full result of both models self-trained with 24 unlabeled training cases In Table 11.

Method	GLUE						AdvGLUE				
	QNLI	**QQP**	**RTE**	**SST2**	**CoLA**	**Avg.**	**QNLI**	**QQP**	**RTE**	**SST2**	**Avg.**
Self-trained BERT^{10k}-340M without human-generated labels											
No self-training	65.93	63.71	68.23	80.96	40.17	63.80	**62.84**	42.31	34.57	24.32	41.01
Single prompt	67.86	61.62	72.22	79.43	42.19	64.66	59.46	40.06	40.47	31.08	42.90
w/ CL	65.66	61.58	64.74	80.71	40.08	60.45	58.12	35.47	42.80	31.52	41.97
Contrastive Coord.	**70.80**	**71.60**	66.66	81.42	**59.51**	**70.00**	59.91	50.43	41.97	**55.18**	51.87
w/ CL	68.51	69.69	**69.19**	81.17	49.57	67.63	60.27	50.43	43.21	51.94	**56.47**
Paraphrased Coord.	68.77	63.50	68.38	**81.65**	42.12	64.88	61.26	**64.53**	**58.43**	33.78	54.50
w/ CL	68.64	64.50	67.87	81.00	39.43	64.29	60.36	56.41	55.14	36.49	52.10
Self-trained DeBERTa^{1k}-350M without human-generated labels											
No self-training	72.23	71.63	73.64	81.42	65.20	72.82	56.08	69.23	64.20	47.30	59.20
Single prompt	72.59	70.95	68.71	86.08	67.15	73.10	62.16	65.81	60.49	50.45	59.73
w/ CL	73.72	73.59	69.07	84.94	68.14	73.89	61.71	68.37	60.9	50.45	60.36
Contrastive Coord.	76.65	**76.53**	74.12	86.62	69.16	76.62	**71.17**	70.37	65.02	54.05	**65.15**
w/ CL	**76.70**	75.98	**75.81**	87.19	68.71	**76.88**	67.57	67.52	**66.26**	**54.28**	63.91
Paraphrased Coord.	75.10	72.84	70.76	**87.54**	**69.32**	75.11	66.44	69.22	64.61	53.61	63.47
w/ CL	73.68	74.02	71.96	87.16	69.13	75.19	67.12	**73.08**	61.73	53.60	63.88

Table 11: Unsupervised adaptation performance of different strategies on GLUE and AdvGLUE corpora using 24 unlabeled data examples. The BERT model is trained with $10k$ MNLI examples, and the DeBERTa model is trained with 1k MNLI examples.