On Prefix-tuning for Lightweight Out-of-distribution Detection

Yawen Ouyang Yongchang Cao Yuan Gao Zhen Wu
Jianbing Zhang Xinyu Dai
National Key Laboratory for Novel Software Technology, Nanjing University, China
Collaborative Innovation Center of Novel Software Technology and Industrialization, China
{ouyangyw, caoyc, gaoy}@smail.nju.edu.cn
{wuz, zjb, daixinyu}@nju.edu.cn

Abstract

Out-of-distribution (OOD) detection, a fundamental task vexing real-world applications, has attracted growing attention in the NLP community. Recently fine-tuning based methods have made promising progress. However, it could be costly to store fine-tuned models for each scenario. In this paper, we depart from the classic fine-tuning based OOD detection toward a parameter-efficient alternative, and propose an unsupervised prefix-tuning based OOD detection framework termed *PTO*. Additionally, to take advantage of optional training data labels and targeted OOD data, two practical extensions of *PTO* are further proposed. Overall, *PTO* and its extensions offer several key advantages of being lightweight, easy-to-reproduce, and theoretically justified. Experimental results show that our methods perform comparably to, even better than, existing fine-tuning based OOD detection approaches under a wide range of metrics, detection settings, and OOD types.

1 Introduction

Detecting out-of-distribution (OOD) inputs is crucial for real-world machine learning systems deployed in the wild (Hendrycks and Gimpel, 2017). For example, for a task-oriented dialogue system designed for particular domains, it can be challenging to ensure that the system is only exposed to utterances from the same distribution as the training utterances, i.e., in-distribution (ID) utterances. Therefore, it would be desirable for the system to detect OOD utterances and return safe responses.

Pretrained language models (PLMs) have been a *de facto* choice for OOD detection in the NLP community, and many fine-tuning based methods have achieved promising results (Arora et al., 2021; Podolskiy et al., 2021; Lang et al., 2022). Despite being effective, these methods require storing fine-tuned models for each scenario, which could be prohibitively expensive. This begs the following question: *Can we achieve effective OOD detection in a parameter-efficient way, i.e., keep PLM parameters frozen?*

To achieve this goal, an unsupervised **P**refix-**T**uning based **O**OD detection framework (*PTO*) is proposed in this paper. The key idea of *PTO* is intuitive: an *in-distribution specific* prefix, optimized with the training data via maximum likelihood, could steer PLMs to assign higher likelihoods to ID samples than PLMs without the prefix, while OOD samples should be assigned lower likelihood. Thus we propose to use the likelihood change triggered by the prefix to detect OOD — samples whose improvement is not obvious (*e.g.*, less than a predefined threshold). Note that the training process of *PTO* does not involve the sample labels, expanding its application to situations where obtaining labeled data is cost-prohibitive.

Going beyond the unsupervised setting, we extend our framework to fully leverage optional supervised data. Specifically, we design two extensions to take advantage of training data labels and incorporate the accessible targeted OOD data encountered in the system deployment environment. These practical and comprehensive extensions could further improve the *PTO* performance.

In a nutshell, *PTO* and its extensions offer compelling advantages of being: (1) **lightweight** (*i.e.*, without tuning the PLM parameters), (2) **easy-to-reproduce** (*i.e.*, no additional hyper-parameters other than prefix-tuning itself), and (3) **theoretically justified** (proofed in Section 3).

Experimental results reveal the effectiveness of our methods in detecting both *semantic* shift and *background* shift OOD sentences (Arora et al., 2021). Especially for the background shift, *PTO* surpasses the previous best baseline by only tuning 10M parameters. Our code and data will be available at https://github.com/1250658183/PTO.

In summary, we make the following contribu-

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1533–1545
July 9-14, 2023 ©2023 Association for Computational Linguistics

No.	Text	Label	Dist.
1	The most cliche films i've ever seen	Neg.	In
2	This movie is a masterpiece	Pos.	In
3	I need a timer to be set	Unk.	S. Out
4	Waiters are very friendly	Pos.	B. Out
5	The food was salty beyond edibility	Neg.	B. Out

Table 1: Examples of ID and OOD sentences. S. Out indicates semantic shift OOD, and B. Out indicates background shift OOD.

tions:

- To the best of our knowledge, we are the first to explore lightweight OOD detection and propose *PTO*, an unsupervised framework without tuning PLM parameters.
- Two extensions of *PTO* are proposed to make full use of optional training labels and targeted OOD data to boost OOD detection performance.
- We show that our proposed parameter-efficient methods could catch up to strong fine-tuned baselines and even surpass them in background shift OOD detection.

2 Problem Setup

Given a collection of training sentences $\mathcal{X}_{train}$ and corresponding labels $\mathcal{Y}_{train}$, we assume they are sampled from in-distribution $P^{in}(X, Y)$. The objective of OOD detection is to decide whether a test sentence is from $P^{in}(X, Y)$ (ID) or not (OOD) (Hendrycks and Gimpel, 2017).

We follow Arora et al. (2021) to classify the types of OOD data as either semantic or background shift based on whether the label space remains the same. Semantic shift happens when we encounter sentences with unknown labels, *e.g.*, a sentiment classifier trained with positive and negative movie reviews receiving a neutral text (Example 3 in Table 1). While background shift is for texts with known labels but different domains or styles, *e.g.*, the classifier for movie reviews receiving restaurant reviews (Example 4, 5 in Table 1).

The goal of all OOD detection methods is to design a score function $S(\mathbf{x})$ that maps each input $\mathbf{x}$ to a single scalar that is distinguishable between ID and OOD. Mathematically, the OOD detector G can be described as:

$$G(S(\mathbf{x}), \delta) = \begin{cases} \text{ID} & S(\mathbf{x}) \geq \delta, \\ \text{OOD} & S(\mathbf{x}) < \delta, \end{cases} \quad (1)$$

where δ is the predefined threshold, and can be adjusted according to the user's requirements. For instance, the threshold is chosen to ensure that the recall rate of ID is 95%.

3 Approach

In this section, we start by presenting our proposed lightweight framework *PTO* (Section 3.1), then introducing two extensions of *PTO* to leverage optional training data (Sections 3.2 to 3.4). Finally, we make a summary in Section 3.5.

3.1 Prefix-tuning based OOD detection (*PTO*)

Our motivation follows prefix-tuning that proper prefix vectors can steer PLMs to generate the desired sentences (Li and Liang, 2021), so we can find in-distribution specific prefix θ_{in} to trigger PLMs to be prone to generating ID sentences, *i.e.*, assigning higher likelihoods to ID sentences than before. Considering that the likelihood sum for all sentences (including ID and OOD) is always 1, θ_{in} would trigger PLMs to assign lower likelihood to OOD sentences than before. Thus the likelihood change caused by the prefix θ_{in} could detect OOD sentences whose likelihood improvement is insignificant.

In detail, we first follow Li and Liang (2021) to prepend randomly initialized θ to all PLM layers (pre-trained GPT-2 (Radford et al., 2019) in our case). Then we optimize it by maximizing the likelihood of training sentences, whilst the parameters of the PLM θ_{plm} remain frozen:

$$\theta_{in} = \text{argmax}_{\theta} \sum_{\mathbf{x}^i \in \mathcal{X}_{train}} \log\, p(\mathbf{x}^i; \theta, \theta_{plm}). \quad (2)$$

With θ_{in}, we define our *PTO* score function for OOD detection as follows:

$$S_{PTO}(\mathbf{x}) = p(\mathbf{x}; \theta_{in}, \theta_{plm}) / p(\mathbf{x}; \theta_{plm}), \quad (3)$$

where $p(\mathbf{x}; \theta_{plm})$ is the likelihood of $\mathbf{x}$ from the vanilla PLM, *i.e.*, without the prefix vectors θ_{in}. Lastly, we can identify whether $\mathbf{x}$ is OOD by replacing $S(\mathbf{x})$ with $S_{PTO}(\mathbf{x})$ in Equation (1).

Theoretical insights of $S_{PTO}(\mathbf{x})$: according to the Bayes' rule, $S_{PTO}(\mathbf{x})$ is proportional to $p(\text{ID}|\mathbf{x})$ — $\mathbf{x}$ with a high S_{PTO} can be interpreted as data with a high probability of being ID. Specifically, according to Bayes' rule, we can rewrite $p(\text{ID}|\mathbf{x})$ as follows:

$$p(\text{ID}|\mathbf{x}) = \frac{p(\mathbf{x}|\text{ID})p(\text{ID})}{p(\mathbf{x})} \propto \frac{p(\mathbf{x}|\text{ID})}{p(\mathbf{x})}. \quad (4)$$

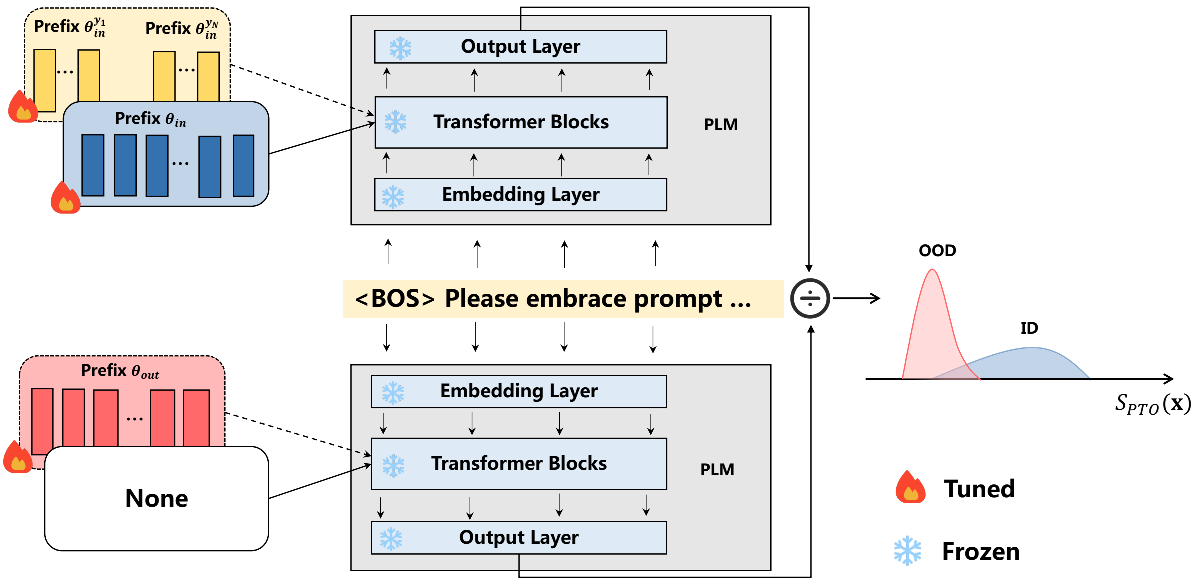

Figure 1: Overview of the proposed *PTO* framework. None means that no prefix is prepended. Two left dashed modules indicate alternative extensions: yellow for label data and red for targeted OOD data. Best viewed in color.

We argue that $p(\mathbf{x}; \theta_{plm})$ (the denominator of $S_{PTO}(\mathbf{x})$) is to estimate $p(\mathbf{x})$ as PLMs are trained with various large corpora. With in-distribution specific prefix θ_{in} prepended, $p(\mathbf{x}; \theta_{in}, \theta_{plm})$ (the numerator of $S_{PTO}(\mathbf{x})$) is to estimate $p(\mathbf{x}|\text{ID})$. Thus their quotient is proportional to $p(\text{ID}|\mathbf{x})$.

3.2 *PTO* with labels (*PTO* + Label)

Using θ_{in} to guide the generation of all sentences $\mathcal{X}_{train}$ would increase the difficulty of the optimization. If training data labels $\mathcal{Y}_{train}$ are available, how can we use them to address this challenge? An intuitive solution is to randomly initialize prefix θ_{in}^{y} for each training label y, and optimize θ_{in}^{y} with corresponding label sentences, so that θ_{in}^{y} can focus on guiding the generation of y sentences:

$$\theta_{in}^{y} = \text{argmax}_{\theta} \sum_{\mathbf{x}^i \in \mathcal{X}_{train} \wedge \mathbf{y}^i = y} \log p(\mathbf{x}^i; \theta, \theta_{plm}). \tag{5}$$

With θ_{in}^{y}, we define $S_{PTO+\text{Label}}$ as follows:

$$S_{PTO+\text{Label}}(\mathbf{x}) = \max_{y} p(\mathbf{x}; \theta_{in}^{y}, \theta_{plm}) / p(\mathbf{x}; \theta_{plm}). \tag{6}$$

Theoretical insights of $S_{PTO+\text{Label}}(\mathbf{x})$: it is proportional to $\max_y p(y|\mathbf{x})$— a high $S_{PTO}(\mathbf{x})$ indicates $\mathbf{x}$ has a high probability of being one of the training labels. In particular, with label-specific prefix θ_{in}^{y} prepended, $p(\mathbf{x}; \theta_{in}^{y}, \theta_{plm})$ is to estimate $p(\mathbf{x}|y)$. Recall that $p(\mathbf{x}; \theta_{plm})$ is to estimate $p(\mathbf{x})$. With the assumption that the label distribution is uniform, $S_{PTO+\text{Label}}(\mathbf{x})$, the estimation of $\max_y p(\mathbf{x}|y)/p(\mathbf{x})$, is proportional to $\max_y p(y|\mathbf{x})$.

3.3 *PTO* with targeted OOD data (*PTO* + OOD)

If we can access some targeted OOD data $\mathcal{X}_{ood}$ in the training process, what can we do to incorporate them into *PTO* to boost OOD detection performance? This scenario has a realistic possibility, such as in a data stream where the OOD data collected by the current detector can be used to refine it. Besides, some benchmark datasets, such as CLINC150 (Larson et al., 2019), also provides some OOD sentences for training.

Our hypothesis is that *targeted out-of-distribution specific* prefix θ_{out} could trigger PLMs to be less prone to generating ID sentences than vanilla PLMs. So the likelihood improvement between θ_{in} and θ_{out} is more obvious for ID sentences. Accordingly, we update *PTO* with the following statistic:

$$S_{PTO+\text{OOD}}(\mathbf{x}) = p(\mathbf{x}; \theta_{in}, \theta_{plm}) / p(\mathbf{x}; \theta_{out}, \theta_{plm}), \tag{7}$$

where θ_{out} is optimized with targeted OOD data:

$$\theta_{out} = \text{argmax}_{\theta} \sum_{\mathbf{x}^i \in \mathcal{X}_{ood}} \log p(\mathbf{x}^i; \theta, \theta_{plm}). \tag{8}$$

Theoretical insights of $S_{PTO+\text{OOD}}(\mathbf{x})$: it is proportional to $p(\text{ID}|\mathbf{x})/p(\text{TOOD}|\mathbf{x})$ — a high

$S_{PTO+\text{OOD}}(\mathbf{x})$ can be interpreted that compared with TOOD (targeted OOD), $\mathbf{x}$ is more likely to belong to ID. Specifically, with θ_{out} prepended, $p(\mathbf{x}; \theta_{out}, \theta_{plm})$ is to estimate $p(\mathbf{x}|\text{TOOD})$. Remember that $p(\mathbf{x}; \theta_{in}, \theta_{plm})$ is to estimate $p(\mathbf{x}|\text{ID})$. Rewriting $p(\mathbf{x}|\text{ID})/p(\mathbf{x}|\text{TOOD})$, we obtain:

$$\frac{p(\mathbf{x}|\text{ID})}{p(\mathbf{x}|\text{TOOD})} = \frac{p(\mathbf{x}|\text{ID})}{p(\mathbf{x})}\frac{p(\mathbf{x})}{p(\mathbf{x}|\text{TOOD})} \propto \frac{p(\text{ID}|\mathbf{x})}{p(\text{TOOD}|\mathbf{x})}. \tag{9}$$

3.4 *PTO* with both label and targeted OOD data (*PTO* + Label + OOD)

The proposed two extensions are orthogonal. We can use them simultaneously in practice if we can access both of them:

$$S_{PTO+\text{Label+OOD}}(\mathbf{x}) = \max_y p(\mathbf{x}; \theta^y_{in}, \theta_{plm})/p(\mathbf{x}; \theta_{out}, \theta_{plm}). \tag{10}$$

Theoretical insights of $S_{PTO+\text{Label+OOD}}(\mathbf{x})$: combining $S_{PTO+\text{Label}}(\mathbf{x})$ and $S_{PTO+\text{OOD}}(\mathbf{x})$, it is simple to prove that $S_{PTO+\text{Label+OOD}}(\mathbf{x})$ is proportional to $\max_y p(y|\mathbf{x})/p(\text{TOOD}|\mathbf{x})$. A high $S_{PTO+\text{Label+OOD}}(\mathbf{x})$ can be interpreted that compared with targeted OOD, $\mathbf{x}$ is more likely to belong to one of the training labels.

3.5 Summary

The advantages of *PTO* and its extensions are numerous:

- **Lightweight**: All of them require only a small number of continuous prefix vectors to be tuned and stored, without modifying PLM parameters.
- **Easy-to-reproduce**: Besides the hyper-parameters of prefix-tuning (*e.g.*, the prefix length), the training and inference process of all methods do not introduce any new hyper-parameters.
- **Theoretically justified**: Through the lenses of Bayes' rule, we provide theoretical insights to understand their effectiveness.

An overview of *PTO* is depicted in Figure 1. We also summarize the training and inference for *PTO* and its extensions in Algorithm 1.

Algorithm 1 OOD detection using *PTO*

```
Input: Training dataset $\mathcal{X}_{train}$, test sample $\mathbf{x}$.
    Optional: training label $\mathcal{Y}_{train}$, targeted OOD $\mathcal{X}_{ood}$.
# Training process
if $\mathcal{Y}_{train}$ is available then
    for each label $y$ do
        Train $\theta^y_{in}$ using Equation (5)
    end for
else
    Train $\theta_{in}$ using Equation (2)
end if
if $\mathcal{X}_{ood}$ is available then
    Train $\theta_{out}$ using Equation (8)
end if
# Inference process
if both $\theta_{out}$ and $\theta^y_{in}$ are unavailable then
    Calculate $S_{PTO}$ using Equation (3)
else if only $\theta^y_{in}$ is available then
    Calculate $S_{PTO+\text{Label}}$ using Equation (6)
else if only $\theta_{out}$ is available then
    Calculate $S_{PTO+\text{OOD}}$ using Equation (7)
else
    Calculate $S_{PTO+\text{Label+OOD}}$ using Equation (10)
end if
```

4 Experimental Setup

4.1 Datasets

We evaluate our methods for detecting semantic shift and background shift OOD:

- For semantic shift, we follow Podolskiy et al. (2021) to use the challenging CLINC150 dataset (Larson et al., 2019). CLINC150 covers utterances across various intents in voice assistants. OOD utterances are those with unknown intents. As aforementioned before, it also provides OOD utterances for training.
- For background shift, we follow Arora et al. (2021) to use IMDB (Maas et al., 2011) as ID and Yelp Polarity (Zhang et al., 2015) as OOD. IMDB is a long movie review dataset and Yelp Polarity is a business review dataset. Since both IMDB and Yelp Polarity do not provide the validation dataset, to perform early stopping, we sample 10000 sentences from IMDB unlabeled dataset and 10000 sentences from Yelp as the validation dataset.

Table 2 provides the summary statistics.

Statistics	CLINC150	IMDB-Yelp
Train-ID	15000	25000
Train-Label	150	2
Train-OOD	250	-
Validation-ID	3000	10000
Validation-OOD	100	10000
Test-ID	4500	25000
Test-OOD	1000	38000

Table 2: Statistics of datasets used in our experiment.

4.2 Baselines

We introduce the strong supervised method Mahalanobis (Podolskiy et al., 2021; Lee et al., 2018b), Energy and Energy + OOD (Liu et al., 2020; Ouyang et al., 2021), MLS (Vaze et al., 2022) as baselines. With a classifier trained with ID sentences and labels,

- **Mahalanobis** defines a score function based on the Mahalanobis distance between the input representation and the nearest class-conditional Gaussian distribution.
- **Energy** uses the sum of the exponential of the classifier logit to detect OOD.
- **Energy + OOD** uses targeted OOD sentences to shape the energy gap between ID and OOD sentences during the training stage.
- **MLS** uses the maximum logit of the classifier to detect OOD.

We also introduce competitive unsupervised method IMLM + BCAD + MDF (Xu et al., 2021), PPL (Arora et al., 2021), LLR (Gangal et al., 2020; Ren et al., 2019):

- **IMLM + BCAD + MDF** also utilizes Mahalanobis distance as features, and two domain-specific fine-tuning approaches are explored to boost the performance.
- **PPL** uses ID sentences to fine-tune the pre-trained GPT-2 model and uses the perplexity to detect OOD.
- **LLR** trains a left-to-right LSTM language model (Sundermeyer et al., 2012) with ID sentences and trains a second language model with perturbed ID sentences. The likelihood ratio between these two language models is used to detect OOD.

4.3 Metrics

We follow Podolskiy et al. (2021); Liu et al. (2020) to use four common OOD detection metrics to measure the performance:

- **AUROC** refers the area under the true positive rate-false positive rate curve.
- **FPR95** refers the false positive rate(FPR) when the true positive rate(TPR) is 95%.
- **AUPR** refers the area under the precision-recall curve. AUPR In (or Out) indicates ID (or OOD) data are treated as positive samples.

4.4 Implementation details

For all methods, the selection of hyper-parameters and early stop strategy are based on AUROC on the validation set.

For our framework, we use the huggingface implementation of GPT2-base (Wolf et al., 2020) as the PLM and the prefix-tuning implementation is derived from OpenPrompt (Ding et al., 2022). All results are averaged over 5 different seeds. The prefix length has an essential impact on the results, so we search it from $\{10, 50, 100, 200, \mathbf{300}, 400, 500\}$. For *PTO* + Label, the total prefix length 300 is equally allocated to each label. For *PTO* + OOD, the OOD prefix length is also set to 300. The hyper-parameters of *PTO* + Label + OOD are consistent with *PTO* + OOD and *PTO* + Label.

For supervised-based baselines, we use pre-trained BERT (Devlin et al., 2019) as the encoder, and tune it with cross-entropy loss. For Energy, we follow Liu et al. (2020) to set T as 1. We adopt mean pooling to obtain the sentence representation as we empirically find that mean pooling is better than [CLS] with MLP used in Ouyang et al. (2021).

For IMLM + BCAD + MDF, we obtain the results from their open-source implementation. For PPL, we also use GPT2-base as the backbone. For LLR method, we follow Gangal et al. (2020) and use an LSTM with 1 layer and 300 hidden size. Embeddings are initialized with 100D Glove (Pennington et al., 2014). To train the background model, we permute 50% of every sentence by replacing the word with the random one in the vocabulary.

5 Main Results

Table 3 shows all method results on OOD detection. We can observe that:

Dataset		Method	AUROC ↑	FPR95 ↓	AUPR In ↑	AUPR Out ↑	#Params
CLINC150	Unsup.	IMLM + BCAD + MDF	83.7 ± 0.4	62.9 ± 1.5	95.3 ± 0.2	54.6 ± 1.8	110M
		PPL	90.7 ± 0.3	32.3 ± 2.2	97.8 ± 0.1	65.9 ± 1.2	124M
		LLR	90.2 ± 0.3	37.1 ± 1.5	97.5 ± 0.1	66.4 ± 1.3	3.7M
		PTO (ours)	**92.8 ± 0.1**	**27.8 ± 0.9**	**98.3 ± 0.1**	**73.8 ± 0.5**	10M
	Sup.	Mahalanobis	97.4 ± 0.1	10.5 ± 0.6	99.4 ± 0.0	89.6 ± 0.6	110M
		Energy	97.6 ± 0.0	10.2 ± 0.4	99.4 ± 0.0	92.0 ± 0.3	110M
		Energy + OOD	**98.1 ± 0.1**	**8.2 ± 0.6**	**99.5 ± 0.0**	**93.9 ± 0.3**	110M
		MLS	97.5 ± 0.1	10.4 ± 0.3	99.4 ± 0.0	91.6 ± 0.3	110M
		PTO + Label + OOD (ours)	96.7 ± 0.4	17.6 ± 1.6	99.2 ± 0.1	89.3 ± 0.8	20M
IMDB-Yelp	Unsup.	IMLM + BCAD + MDF	97.4 ± 0.0	9.2 ± 0.1	97.2 ± 0.0	97.8 ± 0.0	110M
		PPL	88.9 ± 0.1	41.7 ± 0.2	85.9 ± 0.2	91.6 ± 0.1	124M
		LLR	90.8 ± 0.4	40.5 ± 1.0	87.9 ± 0.4	93.7 ± 0.3	71M
		PTO (ours)	**99.3 ± 0.1**	**2.8 ± 0.4**	**99.2 ± 0.1**	**99.6 ± 0.1**	10M
	Sup.	Mahalanobis	97.0 ± 0.2	11.7 ± 2.7	96.4 ± 0.8	97.6 ± 0.5	110M
		Energy	76.5 ± 1.2	53.8 ± 2.8	75.6 ± 1.2	77.0 ± 1.6	110M
		MLS	76.5 ± 1.3	53.8 ± 2.8	75.5 ± 1.3	77.1 ± 1.2	110M
		PTO + Label (ours)	**99.6 ± 0.1**	**2.0 ± 0.2**	**99.4 ± 0.1**	**99.3 ± 0.0**	10M

Table 3: OOD detection performance on CLINC150 and IMDB-Yelp datasets. #Params indicates the tuning parameter number. The best results of each setting are in **bold**. All results are in percentages. Since IMDB-Yelp does not provide OOD training sentences, we only report the OOD extension performance (*i.e.*, *PTO* + Label + OOD) on CLINC150.

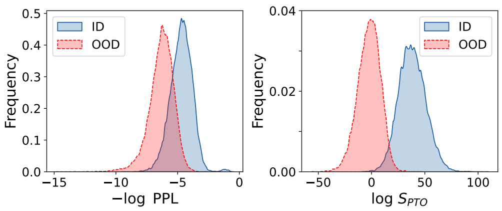

Figure 2: Histogram of the OOD detection score from PPL (left) and *PTO* (right) on IMDB-Yelp.

- ***PTO* works better than unsupervised baselines on all datasets and metrics.** For CLINC150, *PTO* reduces the FPR95 by **4.5%** compared to the best unsupervised baseline, and *PTO* consistently outperforms the baseline by **6.4%** on IMDB-Yelp. Figure 2 shows the *PTO* and PPL score histogram distributions. We can see that *PTO* is more distinguishable between ID and OOD than PPL, resulting in more effective OOD detection. To gain further insights, we also test prefix-equipped PPL, and its performance is also inferior to *PTO* (38.4% FPR95 on CLINC150).

- ***PTO* + Label (+ OOD) outperforms supervised baselines on background shift by a large margin and achieves competitive performance on semantic shift.** Note that all supervised methods require tuning pretrained language models, whereas our methods do not, so they provide effectiveness while still being lightweight (*PTO* + Label + OOD only tunes 20M parameters, less than 20% of the supervised methods). We also generalize *PTO* + Label + OOD to GPT2-medium, and it can achieve better performance (14.8% FPR95 on CLINC150).

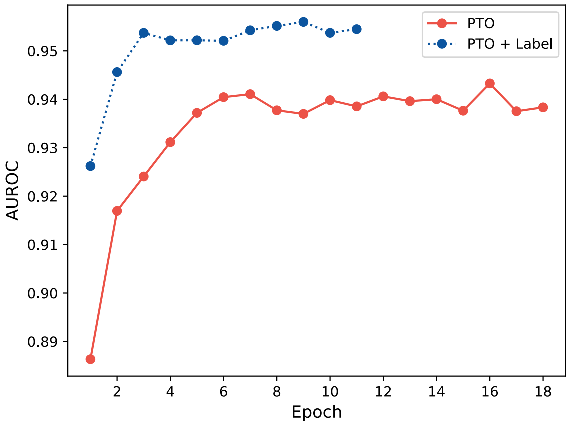

Figure 3: AUROC under different epochs on the CLINC150 validation set.

6 Discussion

6.1 Effect of the label extension

***PTO* + Label provides a performance boost over *PTO* with the same tuning parameter number.** As we can observe from Table 4, the improvement

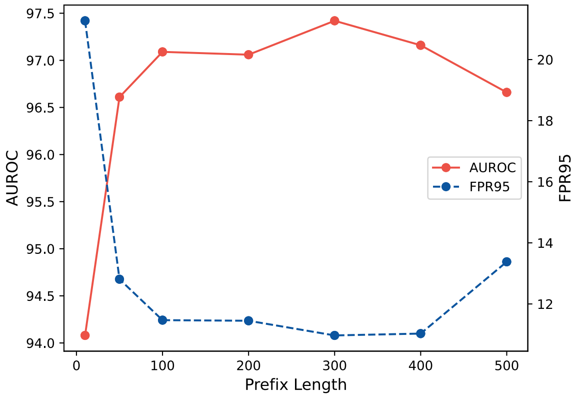

Figure 4: AUROC and FPR95 of *PTO* under different prefix lengths on the IMDB-Yelp validation set.

Method	CLINC150	IMDB-Yelp
PTO	92.8 ± 0.1	99.3 ± 0.1
PTO + Label	94.3 ± 0.2	$\mathbf{99.6 \pm 0.1}$
PTO + OOD	$\mathbf{95.4 \pm 0.3}$	-

Table 4: Comparison of *PTO* with its extensions. Values are AUROC on the test set.

is more pronounced on the challenging dataset CLINC150, where we show a **1.5%** improvement on the AUROC. Notably, *PTO* + Label has the same tuning parameter number with *PTO* (*i.e.*, both are equipped with 300 prefix vectors).

***PTO* + Label can trigger the GPT-2 to assign higher likelihoods to ID sentences than *PTO*.** Specifically, equipped with the label extension for *PTO*, the average log PPL of ID sentences on the validation set degrades from **3.01** to **2.23** on CLINC150, and from **3.72** to **3.70** on IMDB-Yelp. The more pronounced effect on CLINC150 is due to the larger label number (150 versus 2).

***PTO* + Label can also lead to faster convergence.** As empirically shown in Figure 3, the best epoch for *PTO* + Label is **9**, while for *PTO* is **16**. The reason is intuitive that with the label extension, each label sentences can focus on optimizing its own prefix.

6.2 Effect of the OOD extension

***PTO* + OOD is more effective than *PTO* + Label on CLINC150.** Table 4 shows that *PTO* + OOD outperforms *PTO* + Label by **1.1%** (AUROC) on CLINC150. We conjecture that equipping training data with targeted OOD data leads to a smaller distribution gap between training and test data than with labels.

***PTO* + OOD keeps being easy-to-reproduce.**

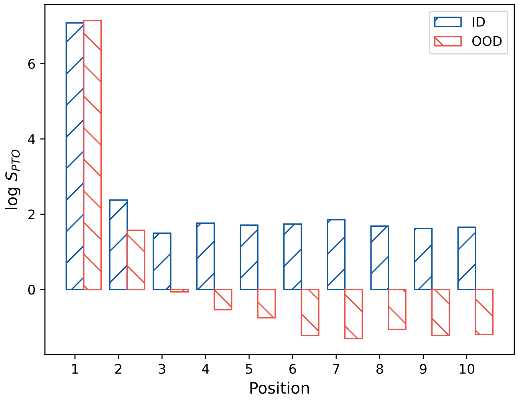

Figure 5: The average $\log S_{PTO}$ score for each position token in ID and OOD sentences. We only list the position from 1 to 10 due to space constraints.

The hyper-parameters of training OOD prefixes are consistent with ID prefixes, so *PTO* + OOD does not require any new hyper-parameter. In contrast, using Energy + OOD requires great effort in hyper-parameter tuning, such as two margin hyper-parameters for the auxiliary hinge loss and the loss weight (Liu et al., 2020).

6.3 Effect of the prefix length

The prefix length is a key hyper-parameter of *PTO*, and previous work shows that the optimal prefix length varies from task to task (Li and Liang, 2021). Inspired by this, we evaluate how the prefix length affects the OOD performance by setting it from 10 to 500. Results from Figure 4 show that as a whole, performance increases as the prefix length increases up to 300 and then decreases. We think this is reasonable, as longer prefixes tend to overfit the training data, and further degrade the validation performance.

6.4 Error analysis

The OOD sentences misclassified by *PTO* always have the same preceding tokens as ID sentences. Specifically, when examining OOD sentences undetected by *PTO* on CLINC150 (*i.e.*, those with higher S_{PTO}), we observe that their first two tokens at the sentence beginning are often found in the ID sentences (see Table 5). The first two tokens further lead to higher OOD sentence scores [*], as shown in Figure 5.

The underlying reason is that *PTO* leverages the left-to-right GPT-2 to estimate the sentence like-

[*] The $\log S_{PTO}$ score of sentence $\mathbf{x}$ is summed over the score of each token w_i in $\mathbf{x}$: $\log S_{PTO}(\mathbf{x}) = \sum_{w_i \in \mathbf{x}} \log p(w_i|\mathbf{w}_{<i};\theta_{in},\theta_{plm}) - \log p(w_i|\mathbf{w}_{<i};\theta_{plm})$

Distribution	2-gram / percent
ID	can you/6.1, i need/4.8, what is/4.5, what 's/3.6, tell me/3.1, i want/2.0, how do/2.0, how much/1.8, how many/1.8, how long/1.6
OOD	can you/6.6, what is/5.9, what 's/5.3, how many/4, tell me/4, how do/3.6, what are/3.1, how much/2.7, look up/2.1, find out/1.8

Table 5: Top 10 2-grams and their percents extracted from ID and OOD sentence beginning. The overlap 2-grams between ID and OOD are marked as blue.

Method	AUROC ↑	FPR95↓	AUPR In↑	AUPR Out↑
MLS	92.22	36.95	97.41	78.07
Energy	92.41	33.75	97.57	78.14

Table 6: Effect of using Energy and MLS derived from the prefix-tuning based classifier.

lihood. The following tokens are invisible when inferring the likelihood of preceding tokens. Therefore, there is no difference between ID and OOD in such case, and *PTO* will assign OOD preceding tokens higher scores as it does to ID. We leave its solution to future work.

6.5 Effect of the prefix-tuning based classifier for OOD detection

To thoroughly investigate the potential of prefix-tuning on OOD detection, we also carried out an experiment based on the prefix-tuning based classifier (Ding et al., 2022; Liu et al., 2021) on CLINC150 dataset. Particularly, we use the utterance's intent as its label words to construct the manual verbalizer (Schick and Schütze, 2021). Meanwhile, we modify the original input $\mathbf{x}$ to the form of template $\mathcal{T}(\mathbf{x}) = \texttt{[PREFIX]}\mathbf{x}\texttt{[MASK]}$, then classify $\mathbf{x}$ based on the probabilities of [MASK] being each label words. Table 6 shows the performance of Energy and MLS scores based on the classifier. We can observe that they perform less well than *PTO* + Label. We argue that a limitation of this strategy is its dependence on the design of the template and verbalizer, while our method *PTO* + Label does not require them.

7 Related Work

7.1 Out-of-distribution detection

Out-of-distribution has gained increasing attention in both NLP and CV recently (Lang et al., 2022; Yang et al., 2022; Sun et al., 2022; Sehwag et al., 2021; Arora et al., 2021). Promising unsupervised (Xu et al., 2021; Arora et al., 2021; Gangal et al., 2020; Ren et al., 2019), supervised with ID labels (Podolskiy et al., 2021; Liu et al., 2020; Vaze et al., 2022), and supervised with OOD data (Liu et al., 2020; Lee et al., 2018a) methods have been proposed. Curious readers may refer to some well established surveys (Yang et al., 2021; Salehi et al., 2022). Unlike prior works, our work focuses on exploring lightweight OOD detection, *i.e.*, without modifying PLM parameters. We propose *PTO* to fulfill this aim and demonstrate its effectiveness through comprehensive experiments.

7.2 Prefix-tuning

Prefix-tuning, a member of the prompt-based tuning family (Liu et al., 2022a), can trigger the desired generation of PLMs by only optimizing small continuous prefix vectors (Li and Liang, 2021). It has achieved desirable performance in many natural language generation tasks (Liu et al., 2022b; Zhao et al., 2022; Ma et al., 2022), and natural language understanding tasks (Liu et al., 2021; Yang and Liu, 2022). However, it still remains a mystery whether prefix-tuning can detect OOD inputs as other fine-tuned models. To the best of our knowledge, we are the first to explore the potential of prefix-tuning for the OOD detection task, and propose approaches for both unsupervised and supervised settings.

8 Conclusion

In this paper, we shed light on lightweight OOD detection, which was largely overlooked in the literature. Our work bridges the gap by proposing *PTO*, an unsupervised prefix-tuning based framework. Moreover, we extend *PTO* to fully leverage the optional training labels and targeted OOD sentences. Our methods have the key advantages of being lightweight, easy-to-reproduce, and theoretically justified. We reveal the effectiveness of *PTO* and its extensions on both semantic and background shift OOD detection. We hope our work could serve as a valuable starting point for future work and inspire them to explore more possibilities of lightweight OOD detection.

Limitations

We consider the current work has the following two limitations:

- We design our lightweight OOD detection framework based on the prefix-tuning paradigm. Nevertheless, there may be other techniques to achieve this goal, which requires further exploration.

- For *PTO* + Label, each label focuses on its own prefixes, suffering from prefix redundancy problem. One can design share prefixes across different labels to trigger label-invariant sentence features.

Acknowledgments

We would like to thank the anonymous reviewers for their insightful comments. Zhen Wu is the corresponding author. Yongchang Cao and Yuan Gao contribute equally. Yawen would like to thank Dingjie Song and Siyu Long for their constructive suggestions. This work is supported by NSFC Projects (Nos. 62206126, 61936012 and 61976114).

References

Udit Arora, William Huang, and He He. 2021. Types of out-of-distribution texts and how to detect them. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10687–10701, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Haitao Zheng, and Maosong Sun. 2022. OpenPrompt: An open-source framework for prompt-learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 105–113, Dublin, Ireland. Association for Computational Linguistics.

Varun Gangal, Arora Abhinav, Einolghozati Arash, and Sonal Gupta. 2020. Likelihood ratios and generative classifiers for unsupervised out-of-domain detection in task oriented dialog. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7764–7771.

Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*.

Hao Lang, Yinhe Zheng, Jian Sun, Fei Huang, Luo Si, and Yongbin Li. 2022. Estimating soft labels for out-of-domain intent detection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 261–276, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An evaluation dataset for intent classification and out-of-scope prediction. 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 1311–1316, Hong Kong, China. Association for Computational Linguistics.

Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. 2018a. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018b. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 7167–7177. Curran Associates, Inc.

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 (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2022a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.* Just Accepted.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. 2020. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 33.

Xiao Liu, Heyan Huang, Ge Shi, and Bo Wang. 2022b. Dynamic prefix-tuning for generative template-based event extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5216–5228, Dublin, Ireland. Association for Computational Linguistics.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2021. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *CoRR*, abs/2110.07602.

Yukun Ma, Trung Hieu Nguyen, and Bin Ma. 2022. Cpt: Cross-modal prefix-tuning for speech-to-text translation. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6217–6221.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Yawen Ouyang, Jiasheng Ye, Yu Chen, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2021. Energy-based unknown intent detection with data manipulation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2852–2861, Online. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Alexander Podolskiy, Dmitry Lipin, Andrey Bout, Ekaterina Artemova, and Irina Piontkovskaya. 2021. Revisiting mahalanobis distance for transformer-based out-of-domain detection. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13675–13682. AAAI Press.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. 2019. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems*, pages 14680–14691.

Mohammadreza Salehi, Hossein Mirzaei, Dan Hendrycks, Yixuan Li, Mohammad Hossein Rohban, and Mohammad Sabokrou. 2022. A unified survey on anomaly, novelty, open-set, and out of-distribution detection: Solutions and future challenges. *Transactions on Machine Learning Research*.

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, Online. Association for Computational Linguistics.

Vikash Sehwag, Mung Chiang, and Prateek Mittal. 2021. Ssd: A unified framework for self-supervised outlier detection. In *International Conference on Learning Representations*.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. 2022. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning*.

Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. Lstm neural networks for language modeling. In *Thirteenth annual conference of the international speech communication association*.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. 2022. Open-set recognition: A good closed-set classifier is all you need. In *International Conference on Learning Representations*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi 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 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.

Keyang Xu, Tongzheng Ren, Shikun Zhang, Yihao Feng, and Caiming Xiong. 2021. Unsupervised out-of-domain detection via pre-trained transformers. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1052–1061, Online. Association for Computational Linguistics.

Jingkang Yang, Pengyun Wang, Dejian Zou, Zitang Zhou, Kunyuan Ding, WENXUAN PENG, Haoqi Wang, Guangyao Chen, Bo Li, Yiyou Sun, Xuefeng Du, Kaiyang Zhou, Wayne Zhang, Dan Hendrycks, Yixuan Li, and Ziwei Liu. 2022. OpenOOD: Benchmarking generalized out-of-distribution detection. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. 2021. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*.

Zonghan Yang and Yang Liu. 2022. On robust prefix-tuning for text classification. In *International Conference on Learning Representations*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 649–657, Cambridge, MA, USA. MIT Press.

Lulu Zhao, Fujia Zheng, Weihao Zeng, Keqing He, Weiran Xu, Huixing Jiang, Wei Wu, and Yanan Wu. 2022. Domain-oriented prefix-tuning: Towards efficient and generalizable fine-tuning for zero-shot dialogue summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4848–4862, Seattle, United States. Association for Computational Linguistics.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section Limitations

□ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
No response.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
4 and 5

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
4

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
4

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.