Appendices for "Moderate-fitting as a Natural Backdoor Defender for Pre-trained Language Models"

Appendix

A Experimental Details

A.1 Main Experiments

For the word-level attack, we randomly choose one meaningless word to insert into a sentence to generate the poisoned sample. The meaningless words we choose are "cf", "mn", "bb", "tq". For the syntactic attack, we use SCPN (Iyyer et al., 2018) to generate the inverted sentence as the poisoned sample. For the word-level attack, the poisoning ratio is 5% for SST-2, AG News and HSOL. For the syntactic attack, the poisoning ratio is 10% for SST-2 and AG News, and the poisoning ratio is 5% for HSOL. For AG News and SST-2, we calculate ACC on the clean test dataset. For HSOL, since there is no official test dataset, we calculate ACC on the clean dev dataset. When preprocessing HSOL samples, we replace the original line break with a space character. For AG News' training data, we sample 11106 training samples from the original AG News' training dataset before poisoning. The experiments are conducted using one A100 NVIDIA GPU.

Reparameterized LoRA. In the original LoRA (Hu et al., 2021), for a pre-trained weight matrix $W_0 \in \mathbb{R}^{d\times k}$ in the multi-head attention, LoRA constrains its update with a low-rank decomposition $W_0 + \Delta W = W_0 + BA$, where $B \in \mathbb{R}^{d\times r}$, $A \in \mathbb{R}^{r\times k}$, and the rank $r \ll \min(\text{d,k})$. LoRA scales $\Delta W x$ by $\frac{\alpha}{r}$. Since A and B have different initializations, we reparameterize them separately. Specifically, we pass one uniformly initialized embedding through a reparameterization network to derive A's weights and the classifier's weights. We use another tunable embedding that is initialized with zeroes, and pass it through a different reparameterization network to derive B's weights. The number of training epochs is set as 10, the learning rate is set as 3×10^{-4}, the rank r is set as 8 and the α is set as 16 for both word-level attack and syntactic attack on SST-2, AG News and HSOL.

Reparameterized Adapter. For reparameterized Adapter, we use one reparameterization network to derive all tunable parameters. The number of training epochs is set as 10 and the learning rate is set as 3×10^{-4} for both word-level attack and syntactic attack on SST-2, AG News and HSOL. Adapter is inserted into each Transformer layer, and it projects the original d-dimensional features into a smaller projection dimension, applies a nonlinear function, then projects the smaller projection dimension back to d (Houlsby et al., 2019). The projection dimension of the Adapter is set as 24 for both attacks on SST-2 and HSOL. For AG News, the projection dimension is set as 24 for the word-level attack, and 1 for the syntactic attack.

Reparameterized Prefix-tuning. The original Prefix-Tuning (Li and Liang, 2021) already applies a reparameterization network for all the prefix tokens. Different from the original setting, we also reparameterize the parameters in the classifier. Specifically, we pass one tunable embedding through another reparameterization network to derive the classifier's weights. The bottleneck dimensions of the two reparameterization networks are the same. The number of training epochs is set as 10 and the learning rate is set as 3×10^{-4} for the word-level attack on SST-2, AG News and HSOL. The

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

number of training epochs is set as 10 and the learning rate is set as 5×10^{-4} for the syntactic attack on SST-2, AG News and HSOL. The number of prefix tokens is set as 24 for all experiments.

For reparameterized LoRA and Adapter, the dimension of the input tunable embedding and the bottleneck dimension of the reparameterization network are the same for all experiments in this paper. For reparameterized Prefix-Tuning, the input dimensions of two reparameterization networks are both set as 512 for all experiments in this paper.

A.2 Analysis

Visualization of the Learning Dynamics. The poisoning ratio is 5% for the poisoned training data. The number of training epochs is set as 10 and the learning rate is set as 3×10^{-4}. The LoRA rank r is 8 and the LoRA α is 16.

Poisoning Ratio. For reparameterized LoRA, the LoRA rank r is set as 8 and the LoRA α is set as 16 for both word-level attack and syntactic attack.

Experiments on a Synthetic Dataset. For SST-2, we use all training samples (6920 samples) whose labels are "negative" or "positive". For AG News, we randomly sample 464 samples whose labels are "World" or "Sports" from the AG News training dataset. We perform the binary classification task on the synthetic dataset. Specifically, we label the sample as "0" if the original label is "negative" or "World", and "1" if the original label is "positive" or "Sports". During the evaluation, we test the performance of SST-2 on the original test dataset of SST-2. We test the performance of AG News using testing samples whose labels are "World" or "Sports" taken from the original test dataset of AG News.

For reparameterized LoRA, the number of training epochs is set as 10. The learning rate is set as 3×10^{-4}. The LoRA rank r is set as 8 and the LoRA α is set as 16.

When fine-tuning the RoBERTa$_{\texttt{BASE}}$ model with different training epochs, the learning rate is set as 2×10^{-5}. When fine-tuning the RoBERTa$_{\texttt{BASE}}$ model with different learning rates, the number of training epochs is set as 10.

A.3 Experiments on Other NLP Backdoor Attacks

For the add-sentence attack, we insert the sentence "I watched this movie last weekend" into a random position in the original sentence to generate the poisoned sample and the poisoning ratio is 5%. For the style transfer attack, we employ the bible style and the poisoning ratio is 10%.

For reparameterized LoRA, the LoRA rank r is 8 and the LoRA α is 16 for both add-sentence attack and style transfer attack on SST-2.

A.4 Comparisons with Other Defense Methods

For the word-level attack and add-sentence attack, the poisoning ratio is 5%. For the syntactic attack and style transfer attack, the poisoning ratio is 10%.

Since ONION, STRIP and RAP are inference-time defense methods, we adapt them to the training-time defense for fair comparisons, following (Cui et al., 2022). After processing the training dataset by the adapted ONION, adapted STRIP, adapted RAP and BKI, we fine-tune the RoBERTa$_{\texttt{BASE}}$ model on the processed training dataset, respectively. The adapted ONION corrects the training samples. For the adapted STRIP, adapted RAP and BKI, we first train a backdoored BERT model and use it to help filter out the potential poisoned training samples, respectively.

About our moderate-fitting method: for the word-level attack, we fine-tune the RoBERTa$_{\texttt{BASE}}$ model for 1 epoch; for the syntactic attack and add-sentence attack, we use reparameterized LoRA with a small bottleneck dimension of 1; for the style transfer attack, we fine-tune the RoBERTa$_{\texttt{BASE}}$ model with a small learning rate of 5×10^{-7}.

Table 1: Results of constraining tunable parameters of PET in one layer.

	LoRA						Adapter					
Layer	**All**	**12**	**9**	**7**	**5**	**1**	**All**	**12**	**9**	**7**	**5**	**1**
	Word-level Attack						Word-level Attack					
ACC (SST-2)	94.29	93.03	93.68	92.92	91.54	88.91	94.29	86.49	93.47	93.14	92.42	88.25
ASR (SST-2)	96.16	65.02	17.54	34.76	16.23	21.93	87.83	26.32	24.56	13.93	13.82	21.38
	Syntactic Attack						Syntactic Attack					
ACC (SST-2)	93.85	91.93	92.75	92.09	90.77	88.85	93.74	84.79	92.53	92.15	91.93	88.69
ASR (SST-2)	90.90	78.40	86.07	76.21	70.72	62.72	86.40	66.12	64.80	68.64	63.71	56.69

Table 2: Results of training a non-pre-trained model with different learning rates.

Learning Rate	$\mathbf{2 \times 10^{-5}}$	$\mathbf{5 \times 10^{-6}}$	$\mathbf{2 \times 10^{-6}}$	$\mathbf{1 \times 10^{-6}}$
	Word-level Attack			
ACC (SST-2)	76.06	72.87	64.52	58.43
ASR (SST-2)	75.33	67.32	58.11	77.30
	Syntactic Attack			
ACC (SST-2)	77.65	71.88	64.91	57.61
ASR (SST-2)	83.11	88.27	89.69	96.27

A.5 Experiments on Backdoor Attacks for the Pre-trained CV Model

The poisoning ratio is 10% for both patch-trigger attack and blending-trigger attack. The transparency of the blending trigger pattern is 0.2 for the blending-trigger attack.

B Additional Experiments and Analyses

B.1 Reducing the Model Capacity by Constraining Tunable Parameters in One Layer

In the main paper, we propose a reparameterization method to reduce the overall model capacity. We also find that constraining the tunable parameters within a single layer also suffices for our goal of reducing the model capacity. Specifically, we only apply LoRA and Adapter to one certain layer in the PLM and also tune the classifier for word-level attack and syntactic attack on SST-2. The number of training epochs is set as 10 and the learning rate is set as 3×10^{-4}. The projection dimension of the Adapter is set as 1. From the experimental results in Table 1, we can see that applying both LoRA and Adapter into all layers cannot well defend against the backdoor attack; instead, constraining the PET algorithm in one layer could effectively defend against both the word-level attack and syntactic attack to some extent. To sum up, the above results demonstrate the effectiveness of reducing model capacity in backdoor defense from another aspect.

B.2 The Effects of Pre-training

To investigate the influence of pre-training when a model is trained on a poisoned dataset, we perform experiments with a randomly initialized model whose architecture is the same as RoBERTa$_{\texttt{BASE}}$ on SST-2. We optimize all parameters of the model with different learning rates for 10 epochs. From the experimental results in Table 2, we can see that for a non-pre-trained model, with the learning rate decreasing, the phenomenon observed on a pre-trained model does not exist. This demonstrates that pre-training may be an important factor for the defense performance. We expect more future works to explore the underlying mechanism of the effects of pre-training.

B.3 Reducing the Training Epochs / Learning Rate Using PET

Reducing the Training Epochs. We perform experiments to investigate the effects of reducing the training epochs when using the original LoRA and Adapter (without reparameterization) for the word-level attack and syntactic attack. We set the learning rate as 3×10^{-4} and set the training epochs as $\{10, 2, 1\}$, respectively. The projection dimension of the Adapter is set as 24. The experimental results are shown in Table 3. From the experimental results, we can see that when we reduce the number of training epochs from 10 to 1 using the original LoRA or Adapter, the ASR decreases

Table 3: Results of reducing the training epochs using PET.

	LoRA						Adapter					
Epochs	**10**	**2**	**1**	**10**	**2**	**1**	**10**	**2**	**1**	**10**	**2**	**1**
	Word-level Attack			Syntactic Attack			Word-level Attack			Syntactic Attack		
ACC (SST-2)	94.29	93.96	93.47	93.85	93.47	92.31	94.34	93.96	93.36	94.18	93.25	93.08
ASR (SST-2)	96.16	9.65	8.33	90.90	72.15	52.52	99.67	10.86	8.88	91.56	81.91	62.06

Table 4: Results of reducing the learning rate using PET.

	Word-level Attack			Syntactic Attack		
Learning Rate	$\mathbf{3 \times 10^{-4}}$	$\mathbf{3 \times 10^{-5}}$	$\mathbf{1 \times 10^{-5}}$	$\mathbf{3 \times 10^{-4}}$	$\mathbf{3 \times 10^{-5}}$	$\mathbf{1 \times 10^{-5}}$
	LoRA					
ACC (SST-2)	94.29	94.07	92.09	93.85	92.75	91.65
ASR (SST-2)	96.16	8.66	10.31	90.90	65.57	39.04
	Adapter					
ACC (SST-2)	94.34	94.56	92.04	94.18	93.90	91.32
ASR (SST-2)	99.67	9.21	10.75	91.56	77.19	49.89

Table 5: Results of the original LoRA when using different rank r.

	Word-level Attack				Syntactic Attack			
r	16	8	4	1	16	8	4	1
ACC (SST-2)	94.45	94.29	94.34	94.56	94.34	93.85	94.29	94.01
ASR (SST-2)	96.05	96.16	95.50	96.82	90.68	90.90	90.02	88.71

sharply while the ACC only declines a little. This demonstrates that reducing the training epochs is also an effective means to defend against backdoor attacks for PET methods.

Reducing the Learning Rate. We perform experiments to investigate the effects of reducing the learning rate when using the original LoRA and Adapter (without reparameterization) for the word-level attack and syntactic attack. The number of training epochs is set as 10. The projection dimension of the Adapter is set as 24. The experimental results are shown in Table 4. The learning rate is chosen from $\{3 \times 10^{-4}, 3 \times 10^{-5}, 1 \times 10^{-5}\}$. From the experimental results, we can see that when we reduce the learning rate from 3×10^{-4} to 1×10^{-5} using the original LoRA or Adapter, the ASR declines significantly while the ACC only drops a little. These results show that reducing the learning rate is also an effective method to defend against backdoor attacks when using PET methods.

B.4 Performance of the Original LoRA When Using Different Rank r

As mentioned in the main paper, although LoRA (Hu et al., 2021) explicitly models the low-rank updates of the parameters, its low-rank structures are individually distributed in different modules in the Transformer layers (dubbed as *local* low-rank architecture). Instead, it is necessary to control the overall intrinsic rank of the weight updates to be smaller than the intrinsic dimension. We perform experiments on SST-2 to see the influence of LoRA rank r in the original LoRA (without reparameterization) on the ACC and ASR. The LoRA α is set as 16 and the number of training epochs is set as 10. The learning rate is set as 3×10^{-4}. The experimental results are shown in Table 5. From the experimental results, we can find that reducing LoRA rank r does not have much influence on the ACC and ASR, which demonstrates the defect of the local low-rank architecture. Instead, we have shown in the main experiments that our global low-rank architecture (the low-rank reparameterization network) is effective in defending against backdoor attacks.

B.5 Performance of the Original Adapter When Using Different Projection Dimensions

We also performed experiments to see the influence of the projection dimension on the ACC and ASR for the original Adapter (without reparameterization). The experiments are conducted on SST-2. The number of training epochs is set as 10. The learning rate is set as 3×10^{-4}. The experimental results are shown in Table 6, from which we can observe that ASR declines slightly (from 99.67% to 87.83% for word-level attack, and from 93.42% to 86.40% for syntactic attack) when the projection dimension decreases from 48 to 1.

Table 6: Results of the original Adapter when using different projection dimensions.

	Word-level Attack				Syntactic Attack			
Projection Dimension	48	24	6	1	48	24	6	1
ACC (SST-2)	94.73	94.34	94.84	94.29	94.51	94.18	94.23	93.74
ASR (SST-2)	99.67	99.67	97.48	87.83	93.42	91.56	92.43	86.40

Table 7: Results of whether to reparameterize the classifier.

	Reparameterize all tunable parameters					Do not reparameterize the classifier				
Bottleneck Dim	**256**	**32**	**4**	**2**	**1**	**256**	**32**	**4**	**2**	**1**
	LoRA									
ACC (SST-2)	93.14	94.34	93.85	92.09	91.98	93.36	93.85	93.41	92.26	90.83
ASR (SST-2)	91.12	91.89	75.44	53.84	42.11	92.00	89.58	78.73	59.76	56.25
	Adapter									
ACC (SST-2)	93.90	94.18	93.41	92.20	89.02	93.63	94.01	94.56	93.79	92.59
ASR (SST-2)	92.11	90.02	71.71	54.82	34.65	93.64	91.56	80.26	70.07	63.38
	Prefix-Tuning									
ACC (SST-2)	92.97	93.47	92.20	89.51	87.37	94.23	93.08	92.42	91.93	91.05
ASR (SST-2)	91.78	88.60	56.91	52.96	42.54	91.89	88.49	67.98	63.38	55.81

B.6 Whether to Reparameterize the Classifier

Our low-rank reparameterization network is applied to all of the tunable parameters defined by a specific PET algorithm, including the classifier. We perform experiments using the syntactic attack. For LoRA, Adapter and Prefix-Tuning, we performed experiments comparing (1) reparameterizing all tunable parameters and (2) reparameterizing the tunable parameters except the classifier. The number of training epochs is set as 10. The learning rate is set as 3×10^{-4} for LoRA and Adapter, and 5×10^{-4} for Prefix-Tuning, respectively. The projection dimension of the Adapter is set as 24. The prefix token number of Prefix-Tuning is set as 24.

The experimental results are shown in Table 7, from which we observe that not reparameterizing the classifier can also defend against the backdoor attack to some extent; however, reparameterizing all tunable parameters is more effective, as reflected in the lower ASR when using the bottleneck dimension of 1 for three PET algorithms, though with the cost of a slight drop of ACC for Adapter and Prefix-Tuning.

B.7 Additional Experiments on Another Synthetic Dataset

When creating the synthetic dataset in the main paper, we choose SST-2 as the primary task, and choose AG News as the subsidiary task. In this subsection, we experiment with choosing AG News as the main task and SST-2 as the subsidiary task. Specifically, we mix the training samples of AG News (only taking two categories of samples) and a very small number of samples from SST-2 to create the synthetic dataset. For SST-2, we randomly sample 346 samples whose labels are "negative" or "positive". For AG News, we take all samples whose labels are "World" or "Sports" from our originally used AG News training dataset (5621 samples). We perform the binary classification task on the synthetic dataset. Specifically, we label the sample as "0" if the original label is "negative" or "World", and "1" if the original label is "positive" or "Sports". During the evaluation, we test the performance of SST-2 on the original test dataset of SST-2. We test the performance of AG News using testing samples whose labels are "World" or "Sports" taken from the original test dataset of AG News.

For reducing the model capacity, we choose the reparameterized LoRA. The number of training epochs is set as 10 and the learning rate is set as 3×10^{-4}. For reducing the training epochs using the fine-tuning method, the learning rate is set as 2×10^{-5}, and the training epochs are chosen from $\{10, 1\}$. For reducing the learning rate using the fine-tuning method, the number of training epochs is set as 10, and the learning rate is chosen from $\{5 \times 10^{-6}, 1 \times 10^{-6}, 5 \times 10^{-7}\}$. The experimental results are shown in Table 8. From the results, we can see that when reducing the model capacity, training epochs or learning rate, the performance of the primary task (AG News) is hardly influenced, while the performance of the subsidiary task (SST-2) drops significantly.

Table 8: Results on a synthetic dataset with samples from AG News (the primary task) and SST-2 (the subsidiary task). We train RoBERTa$_{\texttt{BASE}}$ on the synthetic dataset, and evaluate the performance of both tasks under different model capacity, training epochs and learning rates.

Tuning Method	**Reparameterized LoRA**			**Fine-tuning**				
Setting	Bottleneck Dimension			Training Epochs		Learning Rate		
	32	6	1	10	1	5×10^{-6}	1×10^{-6}	5×10^{-7}
SST-2	86.27	79.13	54.75	87.53	56.84	87.37	76.22	60.85
AG News	97.95	97.42	97.11	97.87	97.21	97.55	97.39	97.05

Table 9: Results when using a sufficiently large learning rate for low-rank reparameterized LoRA and Adapter.

	Low-rank reparameterized LoRA				**Low-rank reparameterized Adapter**			
Learning Rate	4×10^{-3}	2×10^{-3}	1×10^{-3}	3×10^{-4}	4×10^{-3}	2×10^{-3}	1×10^{-3}	3×10^{-4}
	Word-level Attack				Word-level Attack			
ACC (SST-2)	93.68	94.34	93.85	92.64	94.67	94.89	94.45	89.18
ASR (SST-2)	94.52	57.35	10.20	10.96	91.34	67.54	7.68	11.95
	Syntactic Attack				Syntactic Attack			
ACC (SST-2)	93.63	93.79	92.75	91.98	94.29	94.12	93.36	89.02
ASR (SST-2)	89.04	82.89	68.53	42.11	93.20	85.31	77.41	34.65

Table 10: Results when using a sufficiently large learning rate for low-rank reparameterized Prefix-Tuning.

	Word-level Attack				**Syntactic Attack**			
Learning Rate	4×10^{-3}	2×10^{-3}	1×10^{-3}	3×10^{-4}	4×10^{-3}	2×10^{-3}	1×10^{-3}	5×10^{-4}
ACC (SST-2)	92.31	91.87	89.73	85.67	92.64	91.32	89.40	87.37
ASR (SST-2)	11.73	13.27	14.80	20.72	56.47	52.52	45.72	42.54

B.8 The Defense Performance is Influenced by Multiple Factors

We have shown that there are several factors that can influence the defense effect, including the learning rate, training epochs, and model capacity. In practice, whether a PLM could defend against the backdoor attack is not just decided by a single factor, but by multiple factors. For example, despite keeping the low model capacity, a PLM could still be attacked if the model is trained with a *sufficiently large* learning rate or number of training epochs[1].

However, we argue that the above issue is not a serious one in that, (1) as long as other factors are kept in a reasonable range, only controlling one factor is enough for effective backdoor defense in most cases; (2) the three proposed methods are orthogonal to each other, and could be combined to achieve better defense performance. And we expect future work to explore how to decide the optimal value for each factor to strike the best trade-off between ACC and ASR. In the following experiments, we demonstrate our point of view that the defense performance is influenced by multiple factors.

Results when Using a Sufficiently Large Learning Rate. We find that even reducing the model capacity, when the learning rate is larger, the ASR may be high in some cases. We perform experiments on low-rank reparameterized LoRA, Adapter and Prefix-Tuning with larger learning rates. The bottleneck dimension of the reparameterization network is set to 1 to achieve the low model capacity. The number of training epochs is set as 10. The prefix token number is set as 24 for Prefix-Tuning. The projection dimension of the Adapter is set as 24. The experimental results of low-rank reparameterized LoRA and Adapter are shown in Table 9. The experimental results of low-rank reparameterized Prefix-Tuning are shown in Table 10. From the experimental results, we can see that when the learning rate is sufficiently large, the ASR could be high even if the model capacity is kept low, except that for Prefix-Tuning under the word-level attack, even if the learning rate is extremely large (i.e., 2×10^{-2}), the ASR is still very low, with the ACC achieving 94.45% and the ASR 10.31%.

Similar phenomenon is observed when fine-tuning RoBERTa$_{\texttt{BASE}}$ using fewer training epochs. We find that when the learning rate is larger, even if we reduce the number of training epochs, the ASR

[1]In all of our experiments in the main paper, when analyzing the effect of one specific factor, we set other factors to reasonable numerical values based on common practice.

Table 11: Results when using a sufficiently large learning rate for fine-tuning under the add-sentence attack.

Learning Rate	$\mathbf{2 \times 10^{-5}}$			$\mathbf{5 \times 10^{-6}}$		
Epochs	10	2	1	10	2	1
ACC (SST-2)	94.78	94.51	94.01	94.89	93.03	90.77
ASR (SST-2)	100.00	100.00	98.79	99.89	86.95	35.20

Table 12: Results when using sufficiently large training epochs for low-rank reparameterized PET.

	LoRA				**Adapter**				**Prefix-Tuning**			
Epochs	**30**	**20**	**15**	**10**	**30**	**20**	**15**	**10**	**30**	**20**	**15**	**10**
	Word-level Attack				Word-level Attack				Word-level Attack			
ACC (SST-2)	93.96	93.52	93.08	92.64	94.18	93.14	92.04	89.18	90.55	90.23	88.85	85.67
ASR (SST-2)	9.65	9.76	9.87	10.96	8.00	7.68	9.43	11.95	13.49	15.57	15.35	20.72
	Syntactic Attack				Syntactic Attack				Syntactic Attack			
ACC (SST-2)	93.68	92.92	92.59	91.98	93.52	92.97	91.71	89.02	91.65	89.79	87.64	87.37
ASR (SST-2)	67.32	57.79	50.55	42.11	76.86	69.08	52.63	34.65	51.64	46.16	44.08	42.54

could be high in some cases. Specifically, we experiment with reducing the training epochs when fine-tuning RoBERTa$_{\texttt{BASE}}$ on SST-2 for the add-sentence attack. The poisoning ratio is 5%. The experimental results are shown in Table 11. When the learning rate is larger, i.e., 2×10^{-5}, the ASR is still high even when the number of training epochs is 1. However, when the learning rate is smaller, i.e., 5×10^{-6}, the ASR declines sharply when the number of training epochs decreases from 10 to 1.

Increasing Training Epochs. Similarly, we find that even if we keep the low model capacity, when the number of training epochs is sufficiently large, the ASR could still be high in some cases. We perform experiments on low-rank reparameterized LoRA, Adapter and Prefix-Tuning by increasing the training epochs. The bottleneck dimension of the reparameterization network is set to 1. The learning rate is set as 3×10^{-4} for the word-level attack for all three PET algorithms. For the syntactic attack, the learning rate is set as 3×10^{-4} for LoRA and Adapter, and set as 5×10^{-4} for Prefix-Tuning, respectively. The prefix token number is set as 24 for Prefix-Tuning. The projection dimension of the Adapter is set as 24. The experimental results are shown in Table 12. From the experimental results, we can see that the ASR rises as the number of training epochs increases for the syntactic attack. However, for the word-level attack, the ASR is still low when the number of training epochs increases from 10 to 30.

C Limitations

Although our methods have achieved excellent performance in defending against backdoor attacks for PLMs, there are still some limitations: (1) As shown in appendix B.8, the defense performance is influenced by multiple factors. However, as stated before, as long as other factors are kept in a reasonable range, only controlling one factor is enough for effective backdoor defense in most cases. In the future, we aim to explore how to decide the optimal value for each factor to strike the best trade-off between ACC and ASR. (2) Although our methods could significantly reduce ASR in most cases, the performance of the original task is slightly influenced, too. It would be interesting to explore how to mitigate the slight performance drop of the original task in the future.

D Broader Impact

Large-scale PLMs have been the foundation models for the entire NLP community and achieved great success in various tasks. However, there are still some security concerns (e.g., the backdoor attack) about their real-world applications. By providing several simple yet effective defense methods against the backdoor attack, this paper opens up a new direction of robust adaptation for PLMs. Although not our initial goal, this work may be adversely leveraged to invent more advanced backdoor attacks, which require designing new defense methods.

References

Ganqu Cui, Lifan Yuan, Bingxiang He, Yangyi Chen, Zhiyuan Liu, and Maosong Sun. 2022. A unified evaluation of textual backdoor learning: Frameworks and benchmarks. *arXiv preprint arXiv:2206.08514*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *ArXiv preprint*, abs/2106.09685.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. 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 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

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.