Adaptive Adversarial Training for Balancing Model Robustness and Standard Performance

Anonymous NAACL submission

Abstract

Adversarial training (AT) is widely used to boost model robustness against adversarial attacks, i.e., adding minor perturbations on the clean input to fool the target model. However, AT can also lead to degraded clean accuracy since it changes the distribution of the training set. Using the Taylor expansion, we find that commonly used adversarial loss functions inherently include clean loss, making it challenging for previous methods to effectively balance accuracy and robustness. Based on this, we establish a flexible AT framework that can explicitly balance the model robustness and clean accuracy by assigning learnable weights to the decomposed adversarial loss. Comprehensive experimental results indicate that our method boosts model robustness while maintaining comparable standard performance.

1 Introduction

Adversarial training (AT) attempts to boost the robustness of neural networks against adversarial examples by augmenting the training set with perturbed samples. While this approach effectively reduces adversarial errors or boosts robust accuracies, it has been observed to impair the standard performance on clean test data (Tramer and Boneh, 2019; Raghunathan et al., 2020; Yuan et al., 2019; Zhang and Li, 2019). Recent discussions (Yoo and Qi, 2021; Yuan et al., 2019; Zhang and Li, 2019) suggest a trade-off in AT, implying the challenge of simultaneously minimising standard and adversarial risks.

This paper focuses on AT for natural language processing (NLP) tasks, especially for text classification. The overarching concept of AT involves a two-level optimization process to enhance the robustness of a model. On the inner level, gradient ascent is employed to optimize small perturbations of the input data, aiming to maximize the model's loss function. On the outer level, gradient descent is utilized to adjust the model parameters to minimize the classification loss of these adversarial examples.

We note that in textual AT, the default iteration number k is often quite small, e.g., 3 for FreeLB (Zhu et al., 2020), TAVAT (Li and Qiu, 2021), and InfoBERT (Wang et al., 2021a), resulting in small perturbation sizes for these methods. Their empirical results indicate that a relatively small perturbation size helps boost both model robustness and performance.

Nevertheless, we doubt whether a small perturbation size is really helpful in improving robustness. Because the inner maximization greatly affects the effectiveness of AT (Wang et al., 2022). A small perturbation size usually generates lower-quality adversarial data, which makes AT useless. For example, the Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2015) can quickly generate adversarial data using one step, but contributing little to robustness.

Thus, it is reasonable to vary the iteration number and the adversarial step size to study how inner maximization affects AT. To this end, we choose two widely adopted AT methods, i.e., the projected gradient descent (PGD) method (Madry et al., 2018)) and the FreeLB method (Zhu et al., 2020) as our baselines to conduct preliminary experiments on the BERT-base (Devlin et al., 2019) model.

We report clean and robust accuracies[1] equipped with PGD and FreeLB in Figure 1 and Figure 2. We find that the existing AT method can hardly improve robustness without hurting clean accuracy, which contradicts the results in previous works. As the perturbation size increases in AT, the robustness increases while the accuracy decreases. Additionally, AT will easily collapse and fail to converge

[1] In this paper, we use clean accuracy to refer to the standard test accuracy and use robust accuracy to refer to the test accuracy on adversarial examples.

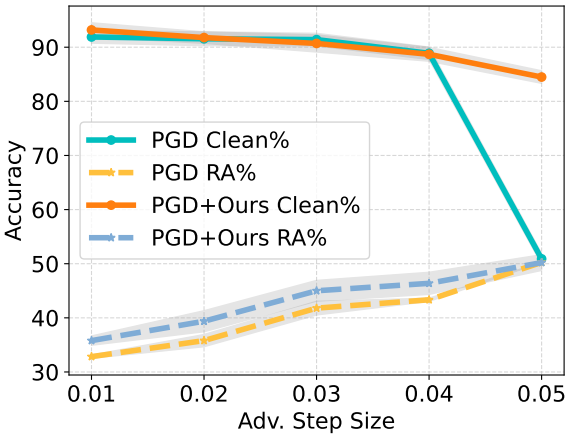

Figure 1: The robust accuracy (**RA**) and the clean accuracy (**Clean**) of the original PGD method (Madry et al., 2018) and ours on the SST-2 dataset (Socher et al., 2013). The backbone model is BERT-base (Devlin et al., 2019). As the step size increases, a trade-off exists between **RA** and **Clean**. It is hard to achieve optimal robustness and clean accuracy simultaneously.

when the perturbation size becomes too large.

This preliminary result indicates that we must rethink the trade-off between robustness and accuracy for NLP models. It also motivates us to investigate whether there exists an optimal perturbation size for the sake of balance, and how to make AT converge in a large perturbation size to achieve strong robustness.

To this end, we theoretically analyze the impact of the perturbation size on the learning objective of AT. In particular, we perform Taylor expansion on the adversarial loss and decompose it into a clean data loss and an adversarial one, in which the adversarial one is the weighted sum of squares of all perturbations. The clean loss corresponds to the model's accuracy, while the adversarial one corresponds to the model's robustness. By assigning trainable weights to all the perturbations, we can explicitly balance the two losses to achieve comparable model robustness and standard performance.

We further provide extensive experimental results. Compared with existing state-of-the-art AT methods, our method demonstrates a remarkable improvement in robustness without sacrificing the standard accuracy. Our main contributions are:

- We demonstrate that existing AT methods for NLP models either fail to improve robustness or compromise clean accuracy.
- We conduct theoretical analysis on a series of gradient-based AT methods. We decompose their learning objectives into distinct adversarial and clean loss components, allowing us to explicitly balance model robustness and accuracy on clean data.
- We establish a flexible AT framework where one can balance adversarial loss and clean loss automatically by assigning learnable weights to adversarial perturbations. Empirical experiments show that our method can improve model robustness without sacrificing clean accuracy.

2 Related Work

2.1 Adversarial Training

AT is widely used to improve robustness against malicious adversarial attacks. Let $f(\cdot)$ be a neural network, Θ be the model parameters, X be the input data set and Y be the corresponding label set, with each input data $x \in X$ and label $y \in Y$. In practice, AT is developed to solve the following max-min optimization problem:

$$\min_{\Theta} \max_{\delta} \mathcal{L}(f(\Theta, x+\delta, y)), \tag{1}$$

where δ is the minor perturbation added to the input.

While the outer minimization is often solved by stochastic gradient descent, how to tackle the inner maximization objective function is still under continuous study. Goodfellow et al. (2015) proposed FGSM to generate perturbations in one gradient ascent step as follows:

$$\delta = sign(\nabla_x \mathcal{L}(\Theta, x, y)), \tag{2}$$

where $sign(\cdot)$ is the sign function.

However, this approximation can hardly find high-quality adversarial data that can maximize the loss function. To seek more precise solutions, Madry et al. (2018) proposed the Projected Gradient Descent (PGD) method to generate perturbations using multi-step gradient ascent steps, i.e.,

$$\begin{aligned} \delta_t &= \alpha \cdot \nabla_\delta \mathcal{L}(f(\Theta, x_{t-1}), y), \\ x_t &= x_{t-1} + \delta_t. \end{aligned} \tag{3}$$

Moreover, PGD initializes the search for adversarial data at random starting points within the allowed norm ball, improving the diversity of adversarial data. Empirically, PGD and its variants are still considered the most effective AT methods.

For NLP tasks, AT was first used to improve the generalization of models. Miyato et al. (2017) proposed virtual AT to enhance text classification in a semi-supervised manner. To further improve language understanding for pre-trained language models, Zhu et al. (2020) proposed FreeLB to provide a large virtual batch size in AT.

In another line of work, AT was adopted to boost the robustness of NLP models. By adversarially perturbing their embedding layer, NLP models were trained to predict consistently on both clean and adversarial data, thereby achieving better adversarial robustness. For example, Li and Qiu (2021) proposed TAVAT to generate token-level perturbations accounting for the importance of tokens. In (Li et al., 2021), the authors increased the iteration numbers of AT and found it useful for boosting robustness. Gao et al. (2023) proposed to minimize the distribution shift risk between clean and adversarial data. Formento et al. (2024) learn robust word embeddings to defend against adversarial attacks.

2.2 The Trade-off between Accuracy and Robustness

In the computer vision domain, while AT helps improve robustness, a vast amount of empirical evidence exists that the clean accuracy can be hurt (Madry et al., 2018; Wang et al., 2020). Zhang et al. (2019) theoretically identify the trade-off between robustness and accuracy by decomposing the prediction error for adversarial examples (robust error) as the sum of the natural error and boundary error. Nevertheless, Yang et al. (2020) have proven that robustness and accuracy should both be achievable for benchmark datasets through locally Lipschitz functions.

For NLP models, early research generally holds that AT improves both robustness and accuracy (Miyato et al., 2017; Ren et al., 2019; Zhu et al., 2020). However, few studies have focused on the trade-off between robustness and accuracy in AT of NLP models.

It is worth noting that several adversarial data augmentation (ADA) methods (Ren et al., 2019; Li et al., 2019; Jin et al., 2020; Li et al., 2020) expand the original training set with crafted adversarial examples. ADA methods introduce larger perturbations than gradient-based AT methods, leading to relatively low clean accuracy. It demonstrates that there is also a trade-off between robustness and accuracy in AT of NLP models.

In this work, we first demonstrate that with a large perturbation size, robustness trades off clean accuracy in gradient-based AT of NLP models. Further, by decomposing the learning objective of AT into a clean classification loss and an adversarial one, we can explicitly balance clean accuracy and robustness.

3 On the Convergence of Adversarial Training

It is widely pointed out that AT is more difficult than standard training for both computer vision and NLP models (Madry et al., 2018; Kurakin et al., 2017). The main reason is that a distribution difference exists between adversarial data and clean data, which makes one model unable to converge well on two widely different data distributions. According to (Gao et al., 2023), we can model the difference via Wasserstein distance. Denote by the distribution of adversarial data P_a, and the distribution of clean data P_c. The p-norm of the perturbation δ is constrained by $||\delta||_p \leq \epsilon$, where ϵ is the maximal perturbation allowed in AT. Gao et al. (2023) proved that the distribution shift is bounded by

$$W_p(P_a, P_c) \leq ||\delta||_p \leq \epsilon, \tag{4}$$

where W_p denotes the $p-$th Wasserstein distance.

Therefore, δ plays a significant role in the convergence of AT. We then vary δ to show its influence on the convergence of AT [2].

We conduct experiments with PGD and FreeLB on the BERT-base (Devlin et al., 2019) model, and the clean accuracy and the robust accuracy with different adversarial step sizes are reported in Figure 2. Our results show that as δ increases, the clean accuracy drops significantly, which implies that AT cannot converge well with large perturbations. Furthermore, robust accuracy gradually increases, demonstrating a tradeoff between clean accuracy and robust accuracy.

It is also intriguing to see that FreeLB can converge under larger perturbations than PGD. We theoretically analyse the differences among different AT methods to understand this phenomenon. Recall the learning objective in AT,

$$\min_{\Theta} \mathbb{E}_{(X,Y)\sim D} \left[\max_{||\delta||\leq\epsilon} \mathcal{L}(\Theta, X+\delta, Y) \right], \tag{5}$$

[2] In practice we vary the step size α in AT to control the perturbation size.

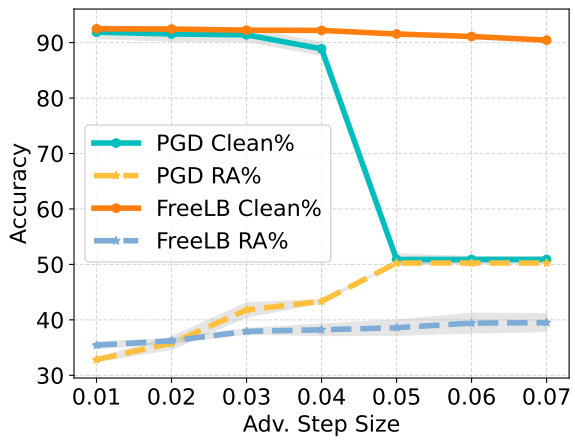

Figure 2: The robust accuracy (**RA**) and clean accuracy (**Clean**) of PGD and FreeLB under different adversarial step sizes on the SST-2 dataset. The backbone model is BERT-base. Although PGD can achieve higher robustness than FreeLB, the clean accuracy of the model is greatly damaged. When the perturbation is too large, the training cannot converge.

where Θ is the model parameters; D is the data distribution; δ is the added perturbation; ϵ is the allowed perturbation size. In the min-max process, multi-step gradient ascent methods often solve the inner maximisation. Take PGD as an example. We initialize x_0 to x and suppose that the iteration number is k and the adversarial step size is α, we have

$$\begin{aligned} &x_0 = x, \\ &x_t = proj_\epsilon(x_{t-1} + \alpha \cdot norm(g(x_{t-1}))), k \geq t \geq 1, \end{aligned} \tag{6}$$

where $g(x_{t-1})$ is the gradient of x_{t-1}, $norm(\cdot)$ can be L_2 normalization. The initial value x_0 can also be randomly sampled within the ϵ-neighborhood of x. In that case, we have $x_0 = x + \delta_0$, where δ_0 is randomly sampled.

For simplicity, we omit the projection function and the normalization function. The main reason is that in (Li et al., 2021), the authors have demonstrated that removing the norm-bounded limitation helps achieve better model robustness.

Thus, we have

$$x_t = x_{t-1} + \delta_t = x + \sum_{t=1}^{k} \delta_t, t \geq 1. \tag{7}$$

In this way, the inner maximization can be reformulated as follows:

$$\max_{\|\delta\| \leq \epsilon} \mathcal{L}(\Theta, x + \sum_{t=1}^{k} \delta_t, y). \tag{8}$$

Considering that δ_t is very small relative to input x, we perform first-order Taylor expansion on the loss function. Thus, combining Eq. (8) and omit the high-order terms, we have

$$\begin{aligned} \mathcal{L}(\Theta, x + \sum_{t=1}^{k} \delta_t, y) &= \mathcal{L}(\Theta, x + \sum_{t=1}^{k-1} \delta_t + \delta_k, y) \\ &\approx \mathcal{L}(\Theta, x + \sum_{t=1}^{k-1} \delta_t, y) + \frac{1}{\alpha}\delta_k^2 \\ &\dots \\ &\approx \mathcal{L}(\Theta, x, y) + \frac{1}{\alpha}\sum_{t=1}^{k} \delta_t^2. \end{aligned} \tag{9}$$

Eq. (9) indicates that one can decompose the loss of adversarial data during PGD training into the corresponding loss of clean data and the sum of squares of all perturbations δ_t.

Therefore, it is reasonable that as the perturbation size increases, the adversarial loss becomes larger and begins to dominate the training, leading to higher robustness. For clean accuracy, as the perturbation size enlarges, the model gets harder to converge on the original training set, resulting in lower clean accuracy.

Based on Eq. (9), we further study how δ affects the convergence of AT. We firstly extend Eq. (9) to FreeLB. It can also be easily extended to other PGD-like methods such as FreeLB++.

According to the FreeLB method, the number of iterations is k and the step size is α. The loss of each iteration will be divided by k and accumulated. The model parameters will be updated at the end (for comparison, PGD only uses the loss of the last iteration to update the model parameters). Similarly, the inner maximization of FreeLB can be formulated as follows:

$$\max_{\|\delta\| \leq \epsilon} \frac{1}{k} \sum_{t=1}^{k} \mathcal{L}(\Theta, x + r^t, y), \tag{10}$$

where $r^t = \sum_{i=1}^{t} \delta_i$. Performing first-order Taylor expansion on Eq. (10), similar to Eq. (9), we have

$$\begin{aligned} \frac{1}{k}\sum_{t=1}^{k} \mathcal{L}(\Theta, x + r^t, y) &\approx \frac{1}{k}\sum_{t=1}^{k}(\mathcal{L}(\Theta, x, y) + \frac{1}{\alpha}\sum_{i=1}^{t}\delta_i^2) \\ &= \mathcal{L}(\Theta, x, y) + \frac{1}{\alpha}\sum_{i=1}^{k}\frac{k-i+1}{k}\delta_i^2. \end{aligned} \tag{11}$$

Eq. (11) indicates that the learning objective of FreeLB can also be decomposed into the clean

Algorithm 1 Adaptive Adversarial Training

Input: Model parameters Θ, loss function $\mathcal{L}$, training set $D = \{x_i, y_i\}_{i=1}^{n}$, number of epochs T, batch size m, number of iterations k, number of batches M, perturbation weights $\mathbf{w}$

Output: robust model parameters Θ

```
for epoch = 1 to T do
    for batch = 1 to M do
        Sample a mini-batch b = {(x_i, y_i)}_{i=1}^m
        Generate adversarial perturbations δ via Eq. (3)
        Add clean data loss L_clean = L(Θ, x, y) and adversarial loss L_adv = β (1/α) Σ_{i=1}^k w_i δ_i^2 together via Eq. (12) to get the overall loss J(Θ, x, y, w)
        Update w via ∇_w J(Θ, x, y, w)
        Update Θ via ∇_Θ J(Θ, x, y, w)
    end for
end for
```

data loss and the weighted sum of squares of all perturbations δ_i, where the weight of δ_i is $\frac{k-i+1}{k}$.

At this point, we can explain more clearly in Figure 2. Since the PGD method inherently has a greater weight for adversarial loss, it can achieve higher robustness than FreeLB, but the training of PGD is more difficult to converge.

4 Adaptive Adversarial Training

Comparing the two learning objectives, one can find an implicit set of weights weighing the perturbation δ_i produced at each iteration i. Further, the weights of clean classification loss and the adversarial one are also implicitly given. For the PGD method with an iteration number of k, the weights of clean loss and the adversarial loss are 1 and k, respectively. The FreeLB method's weights are 1 and $(k+1)/2$, respectively.

Therefore, we summarize the learning objective $\mathcal{J}$ of the two methods into the following formula:

$$\mathcal{J}(\Theta, x, y, \mathbf{w}) = \mathcal{L}(\Theta, x, y) + \beta \frac{1}{\alpha} \sum_{i=1}^{k} w_i \delta_i^2, \quad s.t. \sum_{i=1}^{k} w_i = 1, w_i \geq 0, \tag{12}$$

where β balances the clean loss and the adversarial loss, and w_i balances all the perturbations.

4.1 The Rationale behind Our Framework

Next, we explain the rationale behind introducing β and $\mathbf{w}$. As deduced above, in the PGD method, the weight of the clean loss is naturally set to 1, while the weight of the adversarial loss is set to k. In the FreeLB method, the weight of the clean loss is also 1, but the weight of the adversarial loss is $(k+1)/2$. To ensure the extensibility of our AT framework, we introduce the parameter β to balance the clean loss and adversarial loss. Specifically, PGD and FreeLB are two special cases of the proposed framework.

Eq. (11) shows that while maintaining the original ratio between clean loss and adversarial loss, the perturbations at each time step t are assigned different weights. Therefore, we introduce a set of parameters $\mathbf{w}$, ensuring that the sum of w_i equals 1, and utilize gradients to solve for the worst-case scenario. The weights $\mathbf{w}$ are continuously updated throughout the training process, to find the optimal solution across the entire training set rather than achieve a local optimum based on a single batch of data.

It is worth noting that, in the PGD method, although different time-step perturbations are not explicitly weighted, one can assume that their weights are uniformly set to 1.

Following the min-max optimization widely used in AT, the final training objective can be defined as:

$$\min_{\Theta} \max_{\mathbf{w}} \mathcal{J}(\Theta, x, y, \mathbf{w}). \tag{13}$$

In this way, we build our novel framework of adaptive AT in a constrained manner, where both the PGD and the FreeLB methods can be considered special cases of our framework. Notably, our framework can encompass a wider range of PGD-based AT algorithms, not limited to FreeLB. We show our proposed adaptive AT method in Algorithm 1.

5 Experimental Setup

5.1 Tasks and Datasets

Following previous important works (Gao et al., 2023; Li et al., 2021; Li and Qiu, 2021), we evaluate our AT method and other baselines on two tasks, i.e., sentiment analysis and natural language inference. We choose the SST-2 (Socher et al., 2013)[3] dataset to perform sentiment analysis task

[3] https://dl.fbaipublicfiles.com/glue/data/SST-2.zip

and choose the QNLI (Wang et al., 2019)[4] dataset to conduct natural language inference task.

The SST-2 dataset contains 67k training samples and 1.8k development samples extracted from movie reviews. We conduct the sentiment classification on this dataset, which aims to determine the sentiment tendency within movie reviews.

The QNLI dataset is converted from The Stanford Question Answering Dataset (Rajpurkar et al., 2016), including 105k training samples and 5.4k development samples. We conduct the natural language inference task on this dataset, which aims to determine whether the context sentence contains the answer to the question.

We provide examples of the two datasets in Appendix A.

5.2 Baseline Methods

5.2.1 Defence Methods

We compare our proposed method with advancing defence methods against adversarial attacks. We first provide results of AT-based baseline methods, including PGD (Madry et al., 2018), FreeLB (Zhu et al., 2020), and TA-VAT (Li and Qiu, 2021). To comprehensively benchmark existing defence methods, we also report the results of InfoBERT (Wang et al., 2021a) and Flooding-X (Liu et al., 2022), which introduce an information bottleneck and "flooding" into AT, respectively.

DSRM (Gao et al., 2023) and SemRoDe (Formento et al., 2024) are not chosen to be baselines. This is because DSRM introduce a distribution shift into adversarial defence, and SemRoDe combines valid adversarial examples in the training process, neither of which applies to our framework.

5.2.2 Attacking Methods

Following previous works, we use TextFooler (Jin et al., 2020), TextBugger (Li et al., 2019), and BAE (Garg and Ramakrishnan, 2020) as our attacking methods to dynamically generate adversarial examples during test time.

We also consider assessing AT methods against high-quality adversarial examples pre-crafted by human annotators. Therefore, we report the robust accuracy of all the models on the adversarial GLUE dataset (Wang et al., 2021b).

5.3 Implementation Details

We implement PGD (Madry et al., 2018), FreeLB (Zhu et al., 2020), TA-VAT (Li and Qiu, 2021), and InfoBERT (Wang et al., 2021a) based on TextDefender (Li et al., 2021). We implement Flooding-X (Liu et al., 2022) following the original paper. The three adversarial attacks are conducted using TextAttack[5] (Morris et al., 2020). All experiments are conducted using GeForce RTX 3090 GPUs. All the settings of adversarial hyper-parameters settings are consistent to provide a fair comparison.

Unless otherwise mentioned, the adversarial step size is set to 0.04; the batch size is 128; the epoch number is 10. To align with the weighting factor of the original method, β is set to k for PGD and TA-VAT and $(k+1)/2$ for FreeLB.

6 Main Results

Our proposed method can be easily extended to PGD-like AT methods. In this part, we advance PGD, FreeLB and TA-VAT with adaptive perturbations to assess the effectiveness of our method. We conduct the main experiments on the BERT-base model to provide comprehensive comparisons with other AT methods.

Note that the value of β is related to the methods being extended. For example, when extending the PGD method using our framework, the value of β is set to k (i.e., the number of iterations) according to Eq. (9). We leave the exploration of the effects of different β values for future work.

Table 1 reports the defence results against different types of adversarial attacks on the SST-2 dataset, including two word-level attacks (TextFooler and BAE), one multi-level attack (TextBugger), and an adversarial test dataset (Adversarial GLUE). The main findings are:

- For clean accuracy, all the baseline AT methods maintain a similar level, since the adversarial strength is moderate. The PGD method has the lowest clean accuracy, which is consistent with the conclusions of previous work.
- For robust accuracy against dynamic adversarial attacks and human-crafted adversarial examples, our method can boost the performance of three AT methods. Compared with InfoBERT and Flooding-X, our method also maintains higher robustness.
- Our method can boost the robust accuracy of PGD, FreeLB and TA-VAT methods while achieving comparable clean accuracy, which is consistent with our motivations.

[4] https://huggingface.co/datasets/nyu-mll/glue

[5] https://github.com/QData/TextAttack

SST-2	Clean %	TextFooler	TextBugger	BAE	AdvGLUE
		RA %	RA %	RA %	RA %
BERT (Devlin et al., 2019)	92.32	8.14	26.83	33.72	31.32
InfoBERT (Wang et al., 2021a)	91.74	10.89	32.68	37.96	32.17
Flooding-X (Liu et al., 2022)	92.32	12.60	32.45	35.44	27.00
PGD (Madry et al., 2018)	89.11	12.96	32.22	35.21	39.13
+Ours	88.99 (-0.12)	16.06 (+3.10)	35.68 (+3.46)	40.02 (+4.81)	43.44 (+4.31)
FreeLB (Zhu et al., 2020)	92.20	9.98	34.06	37.73	30.13
+Ours	91.63 (-0.57)	15.69 (+5.71)	38.73 (+4.67)	41.22 (+3.49)	38.53 (+8.40)
TA-VAT (Li and Qiu, 2021)	91.40	11.93	35.89	37.61	32.00
+Ours	91.51 (+0.11)	18.46 (+6.53)	39.60 (+3.71)	40.94 (+3.33)	39.42 (+7.42)

Table 1: The clean accuracy ("Clean %") and the robust accuracy ("RA %") on the SST-2 dataset against TextFooler, TextBugger, and BAE attacks. We report the robust accuracy of the models on the adversarial GLUE dataset to evaluate AT methods against pre-crafted adversarial examples. The backbone model is BERT-base.

QNLI	Clean %	TextFooler	TextBugger	BAE	AdvGLUE
		RA %	RA %	RA %	RA %
BERT (Devlin et al., 2019)	90.60	8.80	9.50	27.90	42.75
InfoBERT (Wang et al., 2021a)	89.10	5.30	6.80	30.90	44.00
Flooding-X (Liu et al., 2022)	91.50	12.00	16.60	40.30	47.00
PGD (Madry et al., 2018)	87.00	11.30	16.80	43.60	41.50
+Ours	87.90 (+0.90)	16.80 (+5.50)	17.20 (+0.40)	41.20 (-2.40)	48.89 (+7.39)
FreeLB (Zhu et al., 2020)	89.60	14.40	14.10	40.50	44.75
+Ours	89.70 (+0.10)	16.60 (+2.20)	17.70 (+3.60)	43.10 (+2.60)	51.75 (+7.00)
TA-VAT (Li and Qiu, 2021)	91.51	12.60	14.30	40.94	43.00
+Ours	91.00 (-0.51)	18.46 (+5.86)	20.30 (+6.00)	44.20 (+3.26)	51.00 (+8.00)

Table 2: The clean accuracy ("Clean %") and the robust accuracy ("RA %") on the QNLI dataset against TextFooler, TextBugger, and BAE attacks. We report the robust accuracy of the models on the adversarial GLUE dataset to evaluate AT methods against pre-crafted adversarial examples. The backbone model is BERT-base.

We also conduct experiments on the QNLI dataset. For natural language inference tasks, we adhere to prior research by allowing the attacking methods to modify the premise sentence while keeping the hypothesis sentence unchanged. The main results are consistent with that on the SST-2 dataset. Our method consistently enhances robust accuracy across various adversarial attacks and test sets. Thanks to the adaptive strength of perturbations, the clean accuracy remains at a comparable level compared to other AT methods.

We note that the PGD method still has the lowest clean accuracy. According to Eq. 9, the PGD method implicitly places a greater weight on the adversarial loss than FreeLB. Since it is directly adopted from the visual domain, no adjustments have been made to the trade-off between robustness and clean accuracy. As a consequence, this method exhibits lower clean accuracy on NLP tasks.

7 Discussions

In this section, we first compare the learning objective of adaptive AT with existing AT methods. Then we further explore how other factors contribute to the robustness, including the number of iterations, the adversarial step size, and the backbone models. We also provide an error analysis of the approximate loss and demonstrate the PGD loss and approximate loss in AT in practice.

7.1 Relation to Existing Work

We list a series of loss functions of AT methods in Table 3 and discuss the main difference between our proposed adaptive AT and conventional AT methods, including fast gradient method (FGM) (Miyato et al., 2017), PGD (Madry et al., 2018), TRADES (Zhang et al., 2019), and FreeLB (Zhu et al., 2020).

More detailed discussions about these methods can be found in Appendix B.

Methods	Loss Function	Flexibility
Standard	$\mathcal{L}(\Theta, x, y)$	-
FGM (Miyato et al., 2017)	$\mathcal{L}(\Theta, x, y) + \mathcal{L}(\Theta, x + \delta, y)$	✗
PGD (Madry et al., 2018)	$\mathcal{L}(\Theta, x + \delta_k, y)$	✗
TRADES (Zhang et al., 2019)	$\mathcal{L}(\Theta, x, y) + \lambda KL(p(\Theta, x)\|\|p(\Theta, x + \delta))$	✓✗
FreeLB (Zhu et al., 2020)	$\frac{1}{k}\sum_{i=1}^{k} \mathcal{L}(\Theta, x + \delta_i, y)$	✗
Adaptive AT	$\mathcal{L}(\Theta, x, y) + \beta \frac{1}{\alpha} \sum_{i=1}^{k} w_i \delta_i^2$	✓

Table 3: Comparisons of different loss functions in AT. The adversarial perturbations in TRADES are generated by maximizing its regularization term (KL-divergence). The **Flexibility** indicates whether the method takes into account the trade-off between model robustness and standard performance. ✗ indicates that the method is unable to balance clean loss and adversarial loss. ✓✗ indicates that the method introduces a hyperparameter to balance the two types of loss, but lacks flexibility because the adversarial loss still contains the clean loss. ✓ indicates that it can explicitly balance clean loss and adversarial loss.

7.2 Impact of Adversarial Step Size

In our AT framework, the adversarial step size α serves as a parameter that influences the adversarial loss. Therefore, it is necessary to study its impact on robustness.

To maintain consistent perturbation strength, we control the product of iteration numbers k and adversarial step size α. Based on this, we vary α to explore its impact on robustness. We report the experimental results in Appendix C and provide detailed discussions.

7.3 Performance on Other Models

We choose DeBERTa-v3-base (He et al., 2021) and RoBERTa (Liu et al., 2019), two improved versions of BERT, as our backbone models to investigate whether our method can boost the robustness of more complex and larger language models. We report the clean and robust accuracy in Appendix D.

7.4 Error Analysis

It is necessary to analyze the error of our method since we have ignored the higher-order terms in the Taylor expansion. Taking the PGD method as an example, we show the error between the approximate loss and the original PGD loss. The original PGD loss is computed by Eq. (8). The approximate is computed by Eq. (9).

In Figure 3, we observe that the approximate loss can well match the loss curve of the PGD method. This demonstrates that our approximation is accurate in the experiments and it can be used to develop AT with an adaptive perturbation.

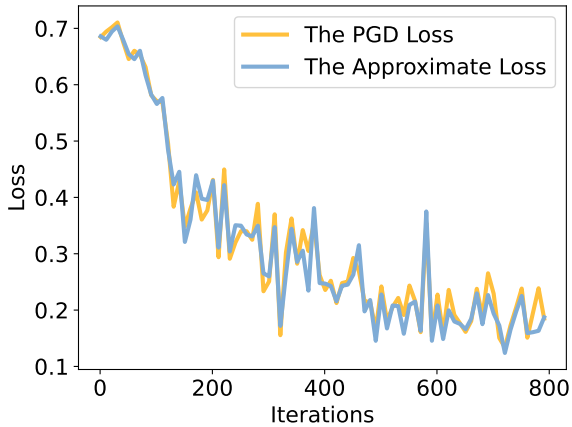

Figure 3: The error between the approximate loss and the original PGD loss on the SST2 dataset over the BERT model. This indicates that our approximation of the experiments is quite accurate.

8 Conclusions

This work seeks to control the weights of adversarial loss and clean loss accurately to balance model robustness and accuracy. To this end, we decompose the learning objective of adversarial training into a pure adversarial loss and clean loss, which correspond to model robustness and clean accuracy, respectively. This way, we can explicitly assign learnable weights to the two losses to balance model robustness and clean accuracy. Experimental results on the SST2 and QNLI datasets over BERT, RoBERTa and DeBERTa models show that our method can boost model robustness without sacrificing clean accuracy.

We hope this work can inspire future work in balancing model robustness and standard performance.

Limitations

This paper leverages Taylor expansion to decompose the loss function (i.e., the cross-entropy function) of AT. The Taylor expansion is a mathematical method used to approximate a function as a power series around a specific point. The loss function must have derivatives of sufficiently high order at the point of expansion and in its vicinity. Specifically, if we want to expand to the n-th order, the function must have at least n derivatives at that point. Although to our best knowledge, the mainstream of loss functions used in AT meet the above conditions, this may not be suitable for more complex loss functions.

References

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.

Brian Formento, Wenjie Feng, Chuan-Sheng Foo, Anh Tuan Luu, and See-Kiong Ng. 2024. SemRoDe: Macro adversarial training to learn representations that are robust to word-level attacks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8005–8028, Mexico City, Mexico. Association for Computational Linguistics.

SongYang Gao, Shihan Dou, Yan Liu, Xiao Wang, Qi Zhang, Zhongyu Wei, Jin Ma, and Ying Shan. 2023. DSRM: Boost textual adversarial training with distribution shift risk minimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12177–12189, Toronto, Canada. Association for Computational Linguistics.

Siddhant Garg and Goutham Ramakrishnan. 2020. BAE: BERT-based adversarial examples for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6174–6181, Online. Association for Computational Linguistics.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

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 *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8018–8025.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. 2017. Adversarial machine learning at scale. In *International Conference on Learning Representations*.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2019. Textbugger: Generating adversarial text against real-world applications. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial attack against BERT using BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6193–6202, Online. Association for Computational Linguistics.

Linyang Li and Xipeng Qiu. 2021. Token-aware virtual adversarial training in natural language understanding. 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 8410–8418.

Zongyi Li, Jianhan Xu, Jiehang Zeng, Linyang Li, Xiaoqing Zheng, Qi Zhang, Kai-Wei Chang, and Cho-Jui Hsieh. 2021. Searching for an effective defender: Benchmarking defense against adversarial word substitution. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3137–3147, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Qin Liu, Rui Zheng, Bao Rong, Jingyi Liu, ZhiHua Liu, Zhanzhan Cheng, Liang Qiao, Tao Gui, Qi Zhang, and Xuanjing Huang. 2022. Flooding-X: Improving BERT's resistance to adversarial attacks via loss-restricted fine-tuning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5634–5644, Dublin, Ireland. Association for Computational Linguistics.

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. *CoRR*, abs/1907.11692.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. 2017. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *CoRR*, abs/1704.03976.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126, Online. Association for Computational Linguistics.

Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John Duchi, and Percy Liang. 2020. Understanding and mitigating the tradeoff between robustness and accuracy. *arXiv preprint arXiv:2002.10716*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1085–1097, Florence, Italy. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew 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, Seattle, Washington, USA. Association for Computational Linguistics.

Florian Tramer and Dan Boneh. 2019. Adversarial training and robustness for multiple perturbations. *Advances in neural information processing systems*, 32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

Boxin Wang, Shuohang Wang, Yu Cheng, Zhe Gan, Ruoxi Jia, Bo Li, and Jingjing Liu. 2021a. Info{bert}: Improving robustness of language models from an information theoretic perspective. In *International Conference on Learning Representations*.

Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. 2021b. Adversarial GLUE: A multi-task benchmark for robustness evaluation of language models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. 2022. On the convergence and robustness of adversarial training. *Preprint*, arXiv:2112.08304.

Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. 2020. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*.

Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Ruslan Salakhutdinov, and Kamalika Chaudhuri. 2020. A closer look at accuracy vs. robustness. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Jin Yong Yoo and Yanjun Qi. 2021. Towards improving adversarial training of nlp models. *arXiv preprint arXiv:2109.00544*.

Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. 2019. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 30(9):2805–2824.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. 2019. Theoretically principled trade-off between robustness and accuracy. In *Proceedings of the 36th International Conference on Machine Learning*, pages 7472–7482.

Jiliang Zhang and Chen Li. 2019. Adversarial examples: Opportunities and challenges. *IEEE transactions on neural networks and learning systems*, 31(7):2578–2593.

Bin Zhu and Yanghui Rao. 2023. Exploring robust overfitting for pre-trained language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5506–5522, Toronto, Canada. Association for Computational Linguistics.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2020. Freelb: Enhanced adversarial training for natural language understanding. In *International Conference on Learning Representations*.

A Examples of Two Datasets

For the SST-2 dataset, an example of x and y is "*On the worst revenge-of-the-nerds clichés the filmmakers could dredge up*" and "*Negative*".

For the QNLI dataset, an example of x and y is "*When did the third Digimon series begin? Unlike the two seasons before it and most of the seasons that followed, Digimon Tamers takes a darker and more realistic approach to its story featuring Digimon who do not reincarnate after their deaths and more complex character development in the original Japanese*" and "*Not entailment*".

B Relation to Existing Work

Specifically, the standard method is designed to minimize the clean data loss, i.e., the cross-entropy on the clean data. The FGM (Miyato et al., 2017) method generates adversarial examples in one gradient ascent step, minimising both clean and adversarial data loss. The PGD method (Madry et al., 2018) generates adversarial examples using multi-step gradient ascent and only minimizes the adversarial data loss in the last step. Similarly, the FreeLB method (Zhu et al., 2020) generates adversarial examples using multi-step gradient ascent. Different from PGD, FreeLB minimize the average of the adversarial loss at each step.

It is important to point out that all these methods implicitly include the clean data loss in the adversarial loss. In particular, as revealed by Eq. (9) and Eq. (11), the conventional adversarial loss can be decomposed into a clean data loss and an adversarial loss. Therefore, although we can introduce hyperparameters to balance clean loss and adversarial loss in these methods, we cannot precisely balance the two losses.

TRADES (Zhang et al., 2019) is theoretically designed to achieve a good trade-off between accuracy and robustness in the computer vision domain, which is the most relevant AT method with our adaptive AT. TRADES decomposes the adversarial error into a natural error and a boundary error. However, the boundary error cannot be effectively computed. In practice, the authors introduce a surrogate loss (i.e., the KL divergence between the model output of clean data and adversarial data) to approximate the boundary error. In this way, TRADES cannot precisely balance the standard performance and robustness.

Our proposed adaptive AT addresses this issue by decomposing the conventional adversarial loss using Taylor expansion. In our learning objective, clean loss and adversarial loss only affect standard performance and model robustness, respectively.

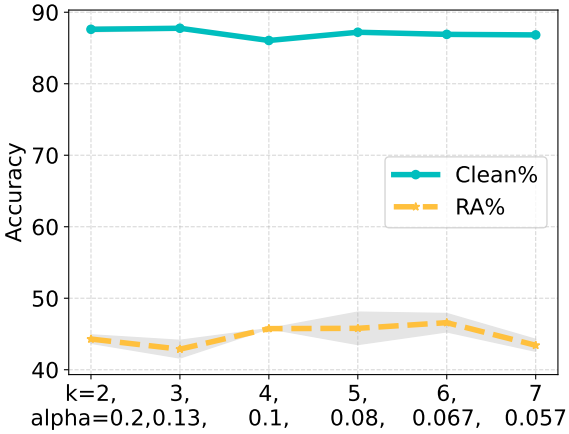

Figure 4: The robust accuracy and clean accuracy under different k and α, while the maximal perturbation size is set to $k\alpha$ following (Li et al., 2021).

C Impact of Adversarial Step Size

We aim to investigate the impact of the perturbation size in AT. In AT, the maximum perturbation size is typically specified. However, what effectively determines the perturbation magnitude are the number of iterations and the adversarial step size. Therefore, given a perturbation size, we vary the number of iterations and step size to investigate their impact on robustness. In other words, we want to find out whether increasing the perturbation strength of adversarial training always helps robustness. We conduct PGD adversarial training on the BERT-base model. Based on our previous experiments, the product of iteration numbers k and adversarial step size α is empirically set to 10 and 0.4.

The main result is reported in Figure 4. It can be seen that when the number of iterations is moderate (5 and 6), the model achieves the best robustness. We suggest that it is unnecessary to set a huge number of iterations during adversarial training. As suggested in (Zhu and Rao, 2023), robust overfitting hinders the AT of NLP models. Too many iterations may lead to robust overfitting of the model and reduce its robustness accuracy on the test set.

D Performance on Other Models

The clean and robust accuracy of DeBERTa-v3-base and RoBERTa-base models are reported in Table 4. These two models can bear a larger pertur-

SST2	**Clean %**	**TextFooler**	**AdvGLUE**
		RA %	**RA %**
RoBERTa	95.07	6.19	39.50
+PGD	94.27	11.47	44.59
+Ours	94.95	11.82	45.22
DeBERTa-v3	95.99	12.60	55.41
+PGD	95.18	13.99	57.14
+Ours	95.76	14.50	67.34

Table 4: The clean and robust accuracy on the RoBERTa (Liu et al., 2019) and DeBERTa-v3-base (He et al., 2021) models.

bation size than the BERT-base model to explore the impact of a larger perturbation range on adversarial training. The empirical results indicate that our adaptive AT can generalize well on larger, more complex models.