Improving Robustness of Language Models from a Geometry-aware Perspective

Anonymous ACL submission

Abstract

Recent studies have found that removing the norm-bounded projection and increasing search steps in adversarial training can significantly improve robustness. However, we observe that a too large number of search steps can hurt accuracy. We aim to obtain strong robustness efficiently using fewer steps. Through a toy experiment, we find that perturbing the clean data to the decision boundary but not crossing it does not degrade the test accuracy. Inspired by this, we propose friendly adversarial data augmentation (FADA) to generate "friendly" adversarial data. On top of FADA, we propose geometry-aware adversarial training (GAT) to perform adversarial training (e.g., FGM) on friendly adversarial data so that we can save a large number of search steps. Comprehensive experiments across two widely used datasets and three pre-trained language models demonstrate that GAT can obtain stronger robustness via less steps. In addition, we provide extensive empirical results and in-depth analyses on robustness to facilitate future studies.

1 Introduction

Deep neural networks (DNNs) outperform humans on many natural language processing (NLP) tasks (Kim, 2014; Vaswani et al., 2017; Devlin et al., 2019). However, recent studies have shown that DNNs are vulnerable to crafted adversarial examples (Szegedy et al., 2013; Goodfellow et al., 2014). For instance, an attacker can mislead an online sentiment analysis system by making minor changes to the input sentences (Papernot et al., 2016; Liang et al., 2017). It has raised concerns among researchers about the security of DNN-based NLP systems. As a result, a growing number of studies are focusing on enhancing robustness to defend against textual adversarial attacks (Jia et al., 2019; Ye et al., 2020; Jones et al., 2020; Zhu et al., 2020).

Existing adversarial defense methods fall into two categories: empirical and certified defenses.

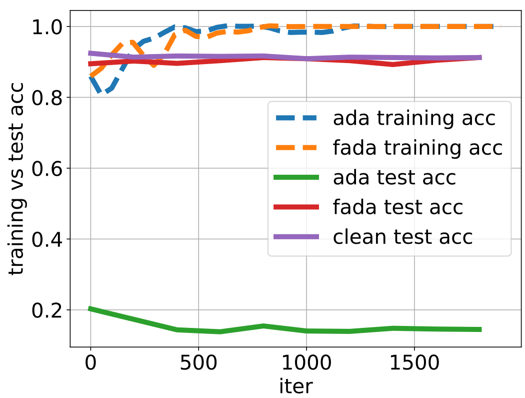


Figure 1: The clean accuracy achieved with ADA, FADA, and the original training set. During training, both ADA and FADA have close to 100% accuracy. However, ADA only achieves ∼15% accuracy during testing while FADA maintains the same test accuracy with the original training set. This indicates that training data which crosses the decision boundary hurts the accuracy significantly.

Empirical defenses include gradient-based adversarial training (AT) and discrete adversarial data augmentation (ADA). Certified defenses provide a provable guaranteed robustness boundary for NLP models. This work focuses on empirical defenses.

There was a common belief that gradient-based AT methods in NLP was ineffective compared with ADA in defending against textual adversarial attacks (Li and Qiu, 2021; Si et al., 2021). Li et al. (2021) find that removing the norm-bounded projection and increasing the number of search steps in adversarial training can significantly improve robustness. Nonetheless, we observe that increasing the number of search steps further does not significantly improve robustness but hurts accuracy.

We give a possible explanation from a geometry-aware perspective. Removing the norm-bounded projection enlarge the search space. Appropriately increasing the number of search steps brings the adversarial data closer to the decision boundary. In

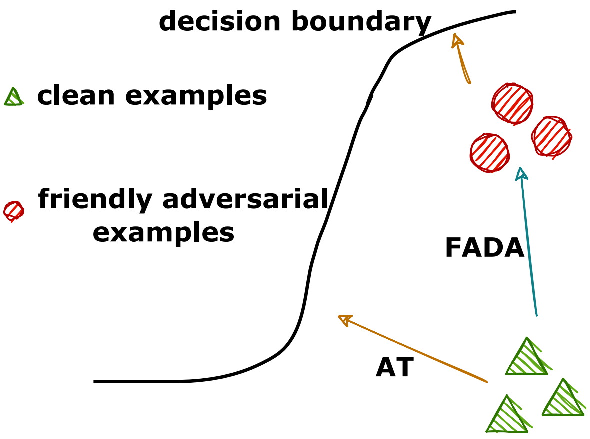


Figure 2: Illustration of GAT. Our GAT can save many search steps since friendly adversarial examples are located near the decision boundary.

this case, the model learns a robust decision boundary. Further increasing the number of search steps can make the adversarial data cross the decision boundary too far, hindering the training of natural data and hurting natural accuracy.

To verify our hypothesis, we train a base model using adversarial data, which are generated by adversarial word substitution (AWS) on the SST-2 (Socher et al., 2013) dataset. We report its training accuracy ("ada training acc") on adversarial data and test accuracy ("ada test acc") on the clean test set in Figure 1. Although achieving nearly 100% training accuracy, its test accuracy is only about 15%, which implicates the adversarial data make the test performance degraded. Then we train another base model, whose training data is more "friendly". We just recover their last modified words to return to the correct class, namely friendly adversarial data augmentation (FADA). It means that only one word is different in each sentence. Surprisingly, it achieves a high test accuracy of $\sim$93%.

This preliminary inspired us to address two existing problems:

- **The number of search steps is always large, which is computationally inefficient**.
- **A too large number of steps leads to degraded test performance**.

Geometrically speaking, the friendly adversarial data are close to the ideal decision boundary. We can address the above two issues in one fell swoop if we perform gradient-based adversarial training on these friendly adversarial data. It is like we start one step before the end, allowing us to obtain strong robustness through a tiny number of search steps. We name it geometry-aware adversarial training (GAT). Figure 2 illustrates our proposed GAT.

In addition, the friendly adversarial data only need to be generated once per dataset. It can be reused, so it is computationally efficient. It can also be updated for every iteration or epoch but computationally expensive.

Our contributions are summarized as follows:

1) We propose FADA to generate friendly adversarial data which are close to the decision boundary (but not crossing it).

2) We propose GAT, a geometry-aware adversarial training method that adds FADA to the training set and performs gradient-based adversarial training.

3) GAT is computationally efficient, and it outperforms state-of-the-art baselines even if using the simplest FGM. We further provide extensive ablation studies and in-depth analyses on GAT, contributing to a better understanding of robustness.

2 Related Work

2.1 Standard Adversarial Training

Let $f_\theta(x)$ be our neural network, $\mathcal{L}(f_\theta(x), y)$ be the loss function (e.g., cross entropy), where $x \in X$ is the input data and $y \in Y$ is the true label. The learning objective of standard adversarial training is

$$\min_{\theta} \mathbb{E}_{(X,Y)\sim D}\left[\max_{\|\delta\|\leq\epsilon} \mathcal{L}(f_\theta(X+\delta), y)\right], \quad (1)$$

where D is the data distribution, δ is the minor perturbation, ϵ is the allowed perturbation size. To optimize the intractable min-max problem, we search for the optimal δ to maximize the inner loss and then minimize the outer loss w.r.t the parameters θ, step by step.

The gradient g of the inner loss w.r.t the input x is used to find the optimal perturbation δ. Goodfellow et al. (2014) proposed fast gradient sign method (FGSM) to obtain δ by one step:

$$\delta = \epsilon \cdot sgn(g), \quad (2)$$

where $sgn(\cdot)$ is the signum function. Madry et al. (2018) proposed projected gradient descent (PGD) to solve the inner maximization as follows:

$$\delta^{(t+1)} = \Pi\, \alpha \cdot g^{(t)} / \|g^{(t)}\|, \forall t \geq 0, \quad (3)$$

where $\alpha > 0$ is the step size (i.e., adversarial learning rate), Π is the projection function that projects the perturbation onto the ϵ-norm ball. Conventionally PGD stops after a predefined number of search steps K, namely PGD-K. In addition, TRADES (Zhang et al., 2019), MART (Wang et al., 2020) and FAT (Zhang et al., 2020) are also effective adversarial training methods for boosting model robustness.

Regarding FAT, the authors propose to stop adversarial training in a predefined number of steps after crossing the decision boundary, which is a little different from our definition of "friendly".

2.2 Adversarial Training in NLP

Gradient-based adversarial training has significantly improved model robustness in vision, while researchers find it helps generalization in NLP. Miyato et al. (2016) find that adversarial and virtual adversarial training have good regularization performance. Sato et al. (2018) propose an interpretable adversarial training method that generates reasonable adversarial texts in the embedding space and enhance models' performance. Zhu et al. (2020) develop FreeLB to improve natural language understanding.

There is also a lot of work focused on robustness. Wang et al. (2021) improve model robustness from an information theoretic perspective. Dong et al. (2021) use a convex hull to capture and defense against adversarial word substitutions. Zhou et al. (2021) train robust models by augmenting training data using Dirichlet Neighborhood Ensemble (DNE).

Besides, adversarial data augmentation is another effective approach to improve robustness (Ebrahimi et al., 2017; Li et al., 2018; Ren et al., 2019; Jin et al., 2019; Zang et al., 2020; Li et al., 2020; Garg and Ramakrishnan, 2020; Si et al., 2021). However, it only works when the augmentation happens to be generated by the same attacking method and often hurts accuracy.

It is worth noting that recent empirical results have shown that previous gradient-based adversarial training methods have little effect on defending against textual adversarial attacks (Li et al., 2021; Si et al., 2021). The authors benchmark existing defense methods and conclude that gradient-based

Algorithm 1 Friendly Adversarial Data Augmentation (FADA)

Input: The original text x, ground truth label y_{true}, base model f_θ, adversarial word substitution function $AWS(\cdot)$

Output: The friendly adversarial example x_f

```
Initialization:
x_f ← x
the last modified word w* ← None
the last modified index i* ← 0
x_adv, w*, i* = AWS(x, y_true, f_θ)
if w* = None then
    return x_f
end if
Replace w_{i*} in x_adv with w*
x_f ← x_adv
return x_f
```

AT can achieve the strongest robustness by removing the norm bounded projection and increasing the search steps.

3 Methodology

3.1 Friendly Adversarial Data Augmentation

For a sentence $x \in X$ with a length of n, it can be denoted as $x = w_1 w_2 ... w_i ... w_{n-1} w_n$, where w_i is the i-th word in x. Its adversarial counterpart x_{adv} can be denoted as $w'_1 w'_2 ... w'_i ... w'_{n-1} w'_n$. In this work, x_{adv} is generated by adversarial word substitution, so x_{adv} has the same length with x. Conventional adversarial data augmentation generates adversarial data fooling the victim model and mixes them with the original training set. As we claim in section 1, these adversarial data can hurt test performance. An interesting and critical question is **when it becomes detrimental to test accuracy**.

One straightforward idea is to recover all the x_{adv} to x word by word and evaluate their impact on test accuracy. We train models only with these adversarial data and test models with the original test set. We are excited that the test accuracy immediately returns to the normal level when we recover the last modified word. We denote these data with only one word recovered as x_f. Geometrically, the only difference between x_{adv} and x_f is whether they have crossed the decision boundary.

To conclude, when the adversarial data cross the decision boundary, they become incredibly harmful to the test performance. We name all the x_f

Algorithm 2 Ideal Geometry-aware Adversarial Training (GAT)

Input: Our base network f_θ, cross entropy loss $\mathcal{L}_{CE}$, training set $D = \{x_i, y_i\}_{i=1}^{n}$, number of epochs T, batch size m, number of batches M
Output: robust network f_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
        for all x_i in b do
            Generate friendly adversarial example x_i^f via Algorithm 1
            Apply an adversarial training method (e.g., FreeLB++) on both x_i and x_i^f to obtain their adversarial counterpart x̃_i and x̃_i^f
        end for
        Update f_θ via ∇_x L_CE(f_θ(x̃_i), y_i) and ∇_x L_CE(f_θ(x̃_i^f), y_i)
    end for
end for
```

as friendly adversarial examples (FAEs) because they improve model robustness without hurting accuracy. Similarly, we name the generation of FAEs as friendly adversarial data augmentation (FADA). We show our proposed FADA in Algorithm 1.

3.2 Geometry-aware Adversarial Training

3.2.1 Seeking for the optimal δ

Recall the inner maximization issue of the learning objective in Eq. (1). Take PGD-K for instance. It divides the search for the optimal perturbation δ into K search steps, and each step requires a backpropagation (BP), which is computationally expensive.

We notice that random initialization of δ^0 is widely used in adversarial training, where δ^0 is always confined to a ϵ-ball centered at x. However, we initialize the clean data via discrete adversarial word substitution in NLP. It is similar to data augmentation (DA), with the difference that we perturb clean data in the direction towards the decision boundary, whereas the direction of data augmentation is random.

By doing so, we decompose the δ into two parts, which can be obtained by word substitution and gradient-based adversarial training, respectively. We denote them as δ_l and δ_s. Therefore, the inner maximization can be reformulated as

$$\max_{\|\delta_l+\delta_s\|\le\epsilon} \mathcal{L}(f_\theta(X+\delta_l+\delta_s), y). \tag{4}$$

We aim to find the maximum δ_l that helps improve robustness without hurting accuracy. As we claim in Section 3.1, FADA generates friendly adversarial data which are close to the decision boundary. Furthermore, the model trained with these friendly adversarial data keeps the same test accuracy as the original training set (Figure 1). Therefore we find the maximum δ_l which is harmless to the test accuracy through FADA.

Denote X_f as the friendly adversarial data generated by FADA, Eq. (4) can be reformulated as

$$\max_{\|\delta_s\|\le\epsilon} \mathcal{L}(f_\theta(X_f+\delta_s), y). \tag{5}$$

The tiny δ_s can be obtained by some gradient-based adversarial training methods (e.g., FreeLB++ (Li et al., 2021)) in few search steps. As a result, a large number of search steps are saved to accelerate adversarial training. We show our proposed geometry-aware adversarial training in Algorithm 2.

3.2.2 Final Learning Objective

It is computationally expensive to update friendly adversarial data for every mini-batch. In practice, we generate static augmentation (X_f,Y) for the training dataset (X,Y) and find it works well with GAT. The static augmentation (X_f,Y) is reusable. Therefore, GAT is computationally efficient.

Through such a tradeoff, our final objective function can be formulated as

$$\begin{aligned}\mathcal{L} = &\mathcal{L}_{CE}(X, Y, \theta) \\ &+ \mathcal{L}_{CE}(\widetilde{X}, Y, \theta) + \mathcal{L}_{CE}(\widetilde{X}_f, Y, \theta),\end{aligned} \tag{6}$$

where $\mathcal{L}_{CE}$ is the cross entropy loss, $\widetilde{X}$ and $\widetilde{X}_f$ are generated from X and X_f using gradient-based adversarial training methods, respectively.

4 Experiments

4.1 Datasets

We conduct experiments on the SST-2 (Socher et al., 2013) and IMDb (Maas et al., 2011) datasets which are widely used for textual adversarial learning. Statistical details are shown in Table 1. We use the GLUE (Wang et al., 2019) version of the SST-2 dataset whose test labels are unavailable. So we report its accuracy on the develop set in our experiments.

Dataset	# train	# dev / test	avg. length
SST-2	67349	872	17
IMDb	25000	25000	201

Table 1: Summary of the two datasets.

SST-2	Clean %	TextFooler			TextBugger			BAE		
		RA %	**ASR %**	**# Query**	**RA %**	**ASR %**	**# Query**	**RA %**	**ASR %**	**# Query**
BERT$_{base}$	92.4	32.8	64.1	72.8	38.5	57.8	44.3	39.8	56.5	64.0
ADA	92.2	46.7	48.7	79.4	42.0	53.9	47.0	41.2	54.8	64.0
ASCC	87.2	32.0	63.3	71.6	27.8	68.2	42.5	41.7	52.1	63.0
DNE	86.6	26.5	69.6	69.0	23.4	73.1	40.2	44.2	49.3	65.8
InfoBERT	92.2	41.7	54.8	74.9	45.2	51.1	45.8	45.4	50.8	65.6
TAVAT	92.2	40.4	56.3	74.3	42.3	54.2	45.7	42.7	53.8	64.2
FreeLB	93.1	42.7	53.7	75.9	48.2	47.7	45.7	46.7	49.3	67.5
FreeLB++10	93.3	41.9	54.8	75.8	46.1	50.3	45.9	44.2	52.4	65.3
FreeLB++30	**93.4**	45.6	50.6	78.1	47.4	48.8	45.7	42.9	53.6	66.0
FreeLB++50	92.0	45.5	50.4	77.2	47.4	48.4	45.3	44.6	51.4	67.5
GAT$_{FGM}$ (ours)	92.8	45.8	49.8	78.5	49.0	46.3	47.0	45.5	50.1	64.9
GAT$_{FreeLB++}$10 (ours)	93.2	49.5	46.3	80.6	52.4	43.2	**47.9**	**48.3**	**46.9**	**68.9**
GAT$_{FreeLB++}$30 (ours)	92.7	**52.5**	**42.2**	**82.3**	**53.8**	**40.9**	47.5	46.1	50.0	65.8

Table 2: Main defense results on the SST-2 dataset, including the test accuracy on the clean test set (**Clean %**), the robust accuracy under adversarial attacks (**RA %**), the attack success rate (**ASR %**), and the average number of queries requiring by the attacker (**# Query**).

4.2 Attacking Methods

Follow Li et al. (2021), we adopt TextFooler (Jin et al., 2019), TextBugger (Li et al., 2018) and BAE (Garg and Ramakrishnan, 2020) as attackers. TextFooler and BAE are word-level attacks and TextBugger is a multi-level attacking method. We also impose restrictions on these attacks for a fair comparison, including:

1. The maximum percentage of perturbed words p_{max}
2. The minimum semantic similarity ε_{min} between the original input and the generated adversarial example
3. The maximum size K_{syn} of one word's synonym set

Since the average sentence length of IMDb and SST-2 are different, p_{max} is set to 0.1 and 0.15, respectively; ε_{min} is set to 0.84; and K_{syn} is set to 50. All settings are referenced from previous work.

4.3 Adversarial Training Baselines

We use BERT$_{base}$ (Devlin et al., 2019) as the base model to evaluate the impact of the following variants of adversarial training on accuracy and robustness and provide a comprehensive comparison with our proposed GAT.

- Adversarial Data Augmentation
- ASCC (Dong et al., 2021)
- DNE (Zhou et al., 2021)
- InfoBERT (Wang et al., 2021)
- TAVAT (Li and Qiu, 2021)
- FreeLB (Zhu et al., 2020)
- FreeLB++ (Li et al., 2021)

ASCC and DNE adopt a convex hull during training. InfoBERT improves robustness using mutual information. TAVAT establishes a token-aware robust training framework. FreeLB++ removes the norm bounded projection and increases search steps.

We only compare GAT with adversarial training-based defense methods and leave comparisons with other defense methods (e.g., certified defenses) for future work.

4.4 Implementation Details

We implement ASCC, DNE, InfoBERT, and TAVAT models based on TextDefender (Li et al., 2021). We implement FGM, FreeLB, FreeLB++, and our GAT based on HuggingFace Transformers [1]. We implement ADA and FADA based on TextAttack [2]. All the adversarial hyper-parameters settings are following their original papers. All the models are trained on two GeForce RTX 2080 GPUs and eight Tesla T4 GPUs.

Regarding the training settings and hyper-parameters, the optimizer is AdamW (Loshchilov and Hutter, 2019); the learning rate is $2e^{-5}$; the number of epochs is 10; the batch size is 64 for SST-2 and 24 for IMDb.

[1] https://huggingface.co/transformers
[2] https://github.com/QData/TextAttack

IMDb	Clean %	TextFooler			TextBugger			BAE		
		RA %	ASR %	# Query	RA %	ASR %	# Query	RA %	ASR %	# Query
BERT$_{base}$	91.2	30.7	66.4	714.4	38.9	57.4	490.3	36.0	60.6	613.6
ADA	91.4	34.6	61.7	804.8	40.5	55.2	538.8	37.0	59.1	693.4
ASCC	86.4	22.2	73.9	595.9	27.2	68.0	415.8	34.7	59.1	642.2
DNE	86.1	14.9	82.2	520.2	17.4	79.3	336.9	35.4	57.8	630.4
InfoBERT	91.9	33.0	63.9	694.1	40.4	55.8	469.9	37.3	59.2	619.6
TAVAT	91.5	37.8	58.9	1082.6	48.8	46.9	695.5	41.2	55.2	896.7
FreeLB	91.3	34.6	61.9	782.0	42.9	52.7	542.7	37.6	58.5	646.7
FreeLB++-10	92.1	39.5	56.8	817.9	46.4	49.3	516.5	41.2	55.0	682.3
FreeLB++-30	92.3	49.8	45.6	992.9	56.0	38.8	600.1	48.3	47.2	788.2
FreeLB++-50	92.3	50.2	45.3	1117.7	56.5	38.5	649.8	48.2	47.5	861.3
GAT$_{FGM}$ (ours)	91.8	58.3	36.0	1004.3	60.4	33.7	556.1	**54.6**	**40.1**	747.4
GAT$_{FreeLB++}$10 (ours)	92.0	50.7	44.7	1093.8	54.7	40.4	648.9	50.7	44.7	908.5
GAT$_{FreeLB++}$30 (ours)	**92.4**	**59.0**	**35.7**	**1629.4**	**62.2**	**32.2**	**914.8**	54.4	40.7	**1213.6**

Table 3: Main defense results on the IMDb dataset.

4.5 Main Results

Our proposed GAT can easily combine with other adversarial training methods. In our experiments, we combine GAT with FGM (GAT$_{FGM}$) and FreeLB++ (GAT$_{FreeLB++}$), respectively. We aim to evaluate if GAT can bring improvements to the simplest (FGM) and the most effective (FreeLB++) AT methods.

We summarize the main defense results on the SST-2 dataset in Table 2. When GAT works with the simplest adversarial training method, FGM, the resulting robustness improvement exceeds FreeLB++50. The effectiveness and efficiency of GAT allow us to obtain strong robustness while saving many search steps. Further combining FreeLB++ on GAT can obtain stronger robustness and outperform all other methods.

Regarding the accuracy, FreeLB++30 obtains the highest 93.4%. GAT also significantly improves accuracy.

In addition, ADA is effective in improving robustness but hurts accuracy. It is not surprising that ASCC and DNE suffer from significant performance losses. However, there is no improvement in robustness and even weaker robustness under TextFooler and TextBugger attacks than the other methods.

Table 3 shows the defense results on the IMDb dataset. The defense performances are generally consistent with that on the SST-2 dataset. It is worth noting that GAT$_{FGM}$ achieved an extremely high **RA %** with a medium **#Query**, which needs further exploration.

AWS	AT method	Clean %	RA %	#Query
None	None	92.4	38.5	44.3
None	FGM	92.5	39.6	44.7
None	FreeLB++30	93.4	47.4	45.7
ADA	None	92.2	42.0	47.0
ADA	FGM	91.3	42.7	46.6
ADA	FreeLB++30	90.9	51.5	47.5
FADA	None	92.7	44.4	45.8
FADA	FGM	92.8	49.0	47.0
FADA	FreeLB++30	92.7	53.8	47.5

Table 4: Ablation studies on the SST-2 dataset. The attacking method is TextBugger. We only report **RA %** and **#Query** due to the space limit. "AWS" means adversarial word substitution methods.

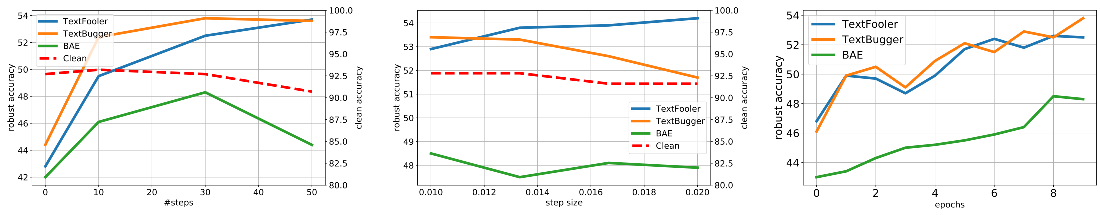
5 Discussions

We further explore other factors that affect robustness and provide comprehensive empirical results.

5.1 Ablation Studies

We conduct ablation studies on the SST-2 dataset to assess the impact of each component of GAT.

As shown in Table 4, "FADA" consistently outperforms "ADA" and "None" with different adversarial training methods. Furthermore, "FADA&FGM" achieve a higher **RA%** than "None&FreeLB++30", which implies that "FADA" can obtain strong robustness in one adversarial search step. "ADA" also helps improve robustness. However, as the number of search steps increases, so does the hurt it does to **Clean %**. On the contrary, "FADA" does not harm **Clean %** but improves it, implying its friendliness.



(a) (b) (c)

Figure 3: (a) Robust and clean accuracy with different search steps. (b) Robust and clean accuracy with different step sizes. (c) Robust accuracy gradually increases on the SST-2 dataset during training. The adversarial training method is $\text{GAT}_{FreeLB++}30$. Zoom in for a better view.

SST-2	clean %	PSO		FastGA	
		RA %	#Query	RA %	#Query
BERT_{base}	92.4	23.9	322.0	39.2	234.4
ADA	92.2	31.4	348.6	43.2	268.4
ASCC	87.2	29.2	359.4	40.5	233.2
DNE	86.6	17.3	266.2	43.9	250.1
InfoBERT	92.2	29.0	335.7	45.3	256.0
TAVAT	92.2	25.7	316.2	42.0	258.7
FreeLB	93.1	27.8	325.6	42.9	267.9
FreeLB++50	92.0	38.4	**368.6**	49.2	258.9
GAT_{FGM}	92.8	29.9	341.0	46.7	275.1
$\text{GAT}_{FreeLB++}10$	**93.2**	34.5	351.3	51.0	289.5
$\text{GAT}_{FreeLB++}30$	92.8	**39.7**	359.2	**53.7**	**323.9**

Table 5: The defense results of different AT methods against two combinatorial optimization attacks. We remove **ASR %** due to the space limit.

5.2 Results with Other Attacks

We have shown that GAT brings significant improvement in robustness against three greedy-based attacks. We investigate whether GAT is effective under combinatorial optimization attacks, such as PSO (Zang et al., 2020) and FastGA (Jia et al., 2019).

We can see from Table 5 that $\text{GAT}_{FreeLB++}30$ obtain the highest **RA %** against the two attacks and $\text{GAT}_{FreeLB++}10$ has the highest clean accuracy. The results demonstrate that our proposed GAT consistently outperforms other defenses against combinatorial optimization attacks.

5.3 Results with More Steps

As we claim in Section 1, the accuracy should degrade with a large number of search steps. But what happens for robustness?

We aim to see if **RA %** can be further improved. Figure 3(a) shows that the **RA %** gradually increases against TextFooler and TextBugger attacks. However, **RA %** decreases against BAE with steps more than 30, which needs more investigation. As the steps increase, the growth rate of **RA %** decreases, and the **Clean %** decreases. We conclude that a reasonable number of steps will be good for both **RA %** and **Clean %**. It is unnecessary to search for too many steps since robustness grows very slowly in the late adversarial training period while accuracy drops.

5.4 Impact of Step Size

A large step size (i.e., adversarial learning rate) will cause performance degradation for conventional adversarial training. Nevertheless, what impact does it have on robustness? We explore the impact of different step sizes on robustness and accuracy. As shown in Figure 3(b), the clean test accuracy slightly drops as the step size increases. The robust accuracy under TextFooler attack increases, while the robust accuracy under Textbugger and BAE attacks decrease. Overall, the impact of step size on robustness needs further study.

5.5 Impact of Training Epochs

Ishida et al. (2020) have shown that preventing further reduction of the training loss when reaching a small value and keeping training can help generalization. In adversarial training, it is naturally hard to achieve zero training loss due to the insufficient capacity of the model (Zhang et al., 2021). Therefore, we investigate whether more training iterations result in stronger robustness in adversarial training. We report the **RA %** achieved by $\text{GAT}_{FreeLB++}30$ at each epoch in Figure 3(c). We observe that the **RA %** tends to improve slowly, implying that more training iterations result in stronger model robustness using GAT.

SST-2	Clean %	TextFooler			TextBugger			BAE		
		RA %	ASR %	# Query	RA %	ASR %	# Query	RA %	ASR %	# Query
RoBERTa$_{base}$	93.0	38.8	58.0	74.5	41.4	55.2	45.5	40.3	56.4	63.6
GAT$_{FGM}$	91.4	47.6	47.7	78.6	49.8	45.3	46.3	42.7	53.2	65.3
GAT$_{FreeLB++}$30	93.2	52.1	43.7	95.5	54.2	41.3	55.8	47.0	49.1	76.9

Table 6: Defense results on RoBERTa model on the SST-2 dataset.

SST-2	Clean %	TextFooler			TextBugger			BAE		
		RA %	ASR %	# Query	RA %	ASR %	# Query	RA %	ASR %	# Query
DeBERTa$_{base}$	94.6	53.7	43.4	79.5	55.1	42.0	48.7	49.8	47.5	66.8
GAT$_{FGM}$	94.5	54.6	42.1	82.6	57.7	38.8	50.0	48.9	48.2	66.7
GAT$_{FreeLB++}$30	94.7	60.4	35.7	83.4	62.0	33.9	51.2	52.2	44.4	69.9

Table 7: Defense results on DeBERTa model on the SST-2 dataset.

5.6 Results with Other Models

We show that GAT can work on more advanced models. We choose RoBERTa$_{base}$ (Liu et al., 2019) and DeBERTa$_{base}$ (He et al., 2021), two improved versions of BERT, as the base models. As shown in Table 6 and Table 7, GAT slightly improve robustness of RoBERTa and DeBERTa models. We can also conclude that DeBERTa is significantly more robust than RoBERTa.

5.7 Limitations

We discuss the limitations of this work as follows.

- As we clarify in Section 3.2.2, instead of dynamically generating friendly adversarial data in training, we choose to pre-generate static augmentation. We do this for efficiency, as dynamically generating discrete sentences in training is computationally expensive. Although it still significantly improves robustness in our experiments, such a tradeoff may lead to failure because the decision boundary changes continuously during training.
- GAT performs adversarial training on friendly adversarial data. It may help if we consider the decision boundaries when performing gradient-based adversarial training—for example, stopping early when the adversarial data crosses the decision boundary. We consider this as one of the directions for future work.

6 Conclusion

In this paper, we study how to improve robustness from a geometry-aware perspective. We first propose FADA to generate friendly adversarial data that are close to the decision boundary. Then we combine gradient-based adversarial training methods on FADA to save a large number of search steps, termed geometry-aware adversarial training (GAT). GAT can efficiently achieve state-of-the-art defense performance without hurting test accuracy.

We conduct extensive experiments to give in-depth analysis, and we hope this work can provide helpful insights on robustness in NLP.

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.

Xinshuai Dong, Anh Tuan Luu, Rongrong Ji, and Hong Liu. 2021. Towards robustness against natural language word substitutions. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2017. Hotflip: White-box adversarial examples for NLP. *CoRR*, abs/1712.06751.

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 2020, Online, November 16-20, 2020*, pages 6174–6181. Association for Computational Linguistics.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Takashi Ishida, Ikko Yamane, Tomoya Sakai, Gang Niu, and Masashi Sugiyama. 2020. Do we need zero training loss after achieving zero training error? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 4604–4614. PMLR.

Robin Jia, Aditi Raghunathan, Kerem Göksel, and Percy Liang. 2019. Certified robustness to adversarial word substitutions. 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 4129–4142, Hong Kong, China. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019. Is BERT Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment. *arXiv e-prints*, page arXiv:1907.11932.

Erik Jones, Robin Jia, Aditi Raghunathan, and Percy Liang. 2020. Robust encodings: A framework for combating adversarial typos. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2752–2765, Online. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2018. Textbugger: Generating adversarial text against real-world applications. *CoRR*, abs/1812.05271.

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.

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. AAAI Press.

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. *CoRR*, abs/2108.12777.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. 2017. Deep text classification can be fooled. *CoRR*, abs/1704.08006.

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.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

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 ACL*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

A. Madry, Aleksandar Makelov, L. Schmidt, D. Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. *ArXiv*, abs/1706.06083.

Takeru Miyato, Andrew M. Dai, and Ian J. Goodfellow. 2016. Virtual adversarial training for semi-supervised text classification. *CoRR*, abs/1605.07725.

Nicolas Papernot, Patrick D. McDaniel, Ananthram Swami, and Richard E. Harang. 2016. Crafting adversarial input sequences for recurrent neural networks. *CoRR*, abs/1604.08275.

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.

Motoki Sato, Jun Suzuki, Hiroyuki Shindo, and Yuji Matsumoto. 2018. Interpretable adversarial perturbation in input embedding space for text. *CoRR*, abs/1805.02917.

Chenglei Si, Zhengyan Zhang, Fanchao Qi, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. 2021. Better robustness by more coverage: Adversarial and mixup data augmentation for robust finetuning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1569–1576, Online. 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 EMNLP*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Ian Erhan, Dumitru; Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

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 *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Boxin Wang, Shuohang Wang, Yu Cheng, Zhe Gan, Ruoxi Jia, Bo Li, and Jingjing Liu. 2021. Infobert: Improving robustness of language models from an information theoretic perspective. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

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*.

Mao Ye, Chengyue Gong, and Qiang Liu. 2020. SAFER: A structure-free approach for certified robustness to adversarial word substitutions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3465–3475, Online. Association for Computational Linguistics.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6066–6080.

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*, volume 97 of *Proceedings of Machine Learning Research*, pages 7472–7482. PMLR.

Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. 2020. Attacks which do not kill training make adversarial learning stronger. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11278–11287. PMLR.

Jingfeng Zhang, Jianing Zhu, Gang Niu, Bo Han, Masashi Sugiyama, and Mohan Kankanhalli. 2021. Geometry-aware instance-reweighted adversarial training. In *International Conference on Learning Representations*.

Yi Zhou, Xiaoqing Zheng, Cho-Jui Hsieh, Kai-Wei Chang, and Xuanjing Huang. 2021. Defense against synonym substitution-based adversarial attacks via dirichlet neighborhood ensemble. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 5482–5492. 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 *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.