Searching for an Effective Defender:
Benchmarking Defense against Adversarial Word Substitution

Zongyi Li[1,2]*, **Jianhan Xu**[1,2]*, **Jiehang Zeng**[1,2]*, **Linyang Li**[1,2],
Xiaoqing Zheng[1,2], **Qi Zhang**[1,2], **Kai-Wei Chang**[3], **Cho-Jui Hsieh**[3]
[1]School of Computer Science, Fudan University, Shanghai, China
[2]Shanghai Key Laboratory of Intelligent Information Processing
[3]Department of Computer Science, University of California, Los Angeles, USA
{zongyili19,jianhanxu20,jhzeng18,zhengxq,qz}@fudan.edu.cn
{kwchang,chohsieh}@cs.ucla.edu

Abstract

Recent studies have shown that deep neural network-based models are vulnerable to intentionally crafted adversarial examples, and various methods have been proposed to defend against adversarial word-substitution attacks for neural NLP models. However, there is a lack of systematic study on comparing different defense approaches under the same attacking setting. In this paper, we seek to fill the gap through comprehensive studies on the behavior of neural text classifiers trained with various defense methods against representative adversarial attacks. In addition, we propose an effective method to further improve the robustness of neural text classifiers against such attacks, and achieved the highest accuracy on both clean and adversarial examples on AGNEWS and IMDB datasets, outperforming existing methods by a significant margin. We hope this study could provide useful clues for future research on text adversarial defense. Codes are available at https://github.com/RockyLzy/TextDefender.

1 Introduction

Deep neural networks have achieved impressive results on NLP tasks. However, they are vulnerable to intentionally crafted textual adversarial examples, which do not change human understanding of sentences but can easily fool deep neural networks. As a result, studies on adversarial attacks and defenses in the text domain have drawn significant attention, especially in recent years (Ebrahimi et al., 2017; Gao et al., 2018; Li et al., 2018; Ren et al., 2019a; Jin et al., 2020a; Li et al., 2020; Jia et al., 2019; Zhu et al., 2020; Zheng et al., 2020; Zhou et al., 2020; Garg and Ramakrishnan, 2020; Zeng et al., 2021). The goal of adversarial defenses is to learn a model that is capable of achieving high test accuracy on both clean (i.e., original) and adversarial examples. We are eager to find out which adversarial defense

---
*Equal contribution

method can improve the robustness of NLP models to the greatest extent while suffering no or little performance drop on the clean input data.

To the best of our knowledge, existing adversarial defense methods for NLP models have yet to be evaluated or compared in a fair and controlled manner. Lack of evaluative and comparative researches impedes understanding of strengths and limitations of different defense methods, thus making it difficult to choose the best defense method for practical use. There are several reasons why previous studies are not sufficient for comprehensive understanding of adversarial defense methods. Firstly, settings of attack algorithms in previous defense works are far from "standardized", and they vary greatly in ways such as synonym-generating methods, number of queries to victim models, maximum percentage of words that can be perturbed, etc. Most defense methods have only been tested on very few attack algorithms. Thus, we cannot determine whether one method consistently performs better than others from experimental data reported in the literature, because a single method might demonstrate more robustness to a specific attack while showing much less robustness to another. Second, some defense methods work well only when a certain condition is satisfied. For example, all existing certified defense methods except RanMASK (Zeng et al., 2021) assume that the defenders are informed of how the adversaries generate synonyms (Jia et al., 2019; Zhou et al., 2020; Dong et al., 2021). It is not a realistic scenario since we cannot impose a limitation on the synonym set used by the attackers. Therefore, we want to know which defense method is more effective against existing adversarial attacks when such limitations are removed for fair comparison among different methods.

In this study, we establish a reproducible and reliable benchmark to evaluate the existing textual defense methods, which can provide detailed insights into the effectiveness of defense algorithms

with the hope to facilitate future studies. In particular, we focus on defense methods against adversarial word substitution, one of the most widely studied attack approaches that could cause major threats in adversarial defenses. In order to rigorously evaluate the performance of defense methods, we propose four evaluation metrics: *clean accuracy*, *accuracy under attack*, *attack success rate* and *number of queries*. The clean accuracy metric measures the generalization ability of NLP models, while the latter three measure the model robustness against adversarial attack. To systematically evaluate the defense performance of different textual defenders, we first define a comprehensive benchmark of textual attack methods to ensures the generation of high-quality textual adversarial examples, which changes the output of models with human imperceptible perturbation to the input. We then impose constraints to the defense algorithms to ensure the fairness of comparison. For example, the synonyms set used by adversaries is not allowed to be accessed by any defense method. Finally, we carry out extensive experiments using typical attack and defense methods for robustness evaluation, including five different attack algorithms and eleven defense methods on both text classification and sentiment analysis tasks.

Through extensive experiments, we found that the gradient-guided adversarial training methods exemplified by PGD (Madry et al., 2018) and FreeLB (Zhu et al., 2020) can be further improved. Furthermore a variant of the FreeLB method (Zhu et al., 2020) outperforms other adversarial defense methods including those proposed years after it. In FreeLB, gradient-guided perturbations are applied to find the most vulnerable ("worst-case") points and the models are trained by optimizing loss from these vulnerable points. However, magnitudes of these perturbations are constrained by a relatively small constant. We find that by extending the search region to a larger ℓ_2-norm through increasing the number of search steps, much better accuracy can be achieved on both clean and adversarial data in various datasets. This improved variant of FreeLB, denoted as FreeLB++, improves the clean accuracy by 0.6% on AGNEWS. FreeLB++ also demonstrates strong robustness under TextFooler attack (Jin et al., 2020b), achieving a 13.6% accuracy improvement comparing to the current state-of-the-art performance (Zeng et al., 2021). Similar results have been confirmed on IMDB dataset.

We believe that our findings invite researchers to reconsider the role of adversarial training, and re-examine the trade-off between accuracy and robustness (Zhang et al., 2019). Also, we hope to draw attentions on designing adversarial attack and defense algorithms based on fair comparisons.

2 Background

2.1 Textual Adversarial Attacks

Textual adversarial attack aims to construct adversarial examples for the purpose of 'fooling' neural network-based NLP models. For example, in text classification tasks, a text classifier $f(\boldsymbol{x})$ maps an input text $\boldsymbol{x} \in \mathcal{X}$ to a label $c \in \mathcal{Y}$, where $\boldsymbol{x} = w_1, \ldots, w_L$ is a text consisting of L words and $\mathcal{Y}$ is a set of discrete categories. Given an original input $\boldsymbol{x}$, an valid adversarial example $\boldsymbol{x}' = w_1', \ldots, w_L'$ is crafted to conform to the following requirements:

$$f(\boldsymbol{x}') \neq y, \quad Sim(\boldsymbol{x}, \boldsymbol{x}') \geq \varepsilon_{min}, \tag{1}$$

where y is the ground truth for $\boldsymbol{x}$, $Sim : \mathcal{X} \times \mathcal{X} \rightarrow [0, 1]$ is a similarity function between the original $\boldsymbol{x}$ and its adversarial example $\boldsymbol{x}'$ and ε_{min} is the minimum similarity. In NLP, Sim is often a semantic similarity function using Universal Sentence Encoder (USE) (Cer et al., 2018) to encode two texts into high dimensional vectors and use their cosine similarity score as an approximation of semantic similarity (Jin et al., 2020a; Li et al., 2018).

2.2 Adversarial Word Substitution

Adversarial word substitution is one of the most widely used textual attack methods, where an adversary arbitrarily replaces the words in the original text $\boldsymbol{x}$ by their synonyms according to a synonym set to alert the prediction of the model. Specially, for each word w, $w' \in S_w$ is any of w's synonyms (including w itself), where the synonym sets S_w are chosen by the adversaries, e.g., built on well-trained word embeddings (Mikolov et al., 2013; Pennington et al., 2014; Su et al., 2018).

The process of adversarial word substitution usually involves two steps: determine an important position to change; and modify words in the selected positions to maximize prediction error of the model. To find a word $w' \in S_w$ that maximizes the model's prediction error, two kinds of searching strategies are introduced: greedy algorithms (Kuleshov et al., 2018; Li et al., 2018; Ren et al., 2019b; Hsieh et al., 2019; Jin et al., 2020b; Li et al.,

Method			Norm-bounded perturbations	Synonyms-agnostic	Structure-free	Ensemble-based
Empirical Defense	Adversarial Data Augmentation	ADA			✓	
		MixADA (Si et al., 2020)			✓	
	Adversarial Training	PGD-K (Madry et al., 2018)	✓	✓	✓	
		FreeLB (Zhu et al., 2020)	✓	✓	✓	
		TA-VAT (Li and Qiu, 2020)	✓	✓	✓	
		InfoBERT (Wang et al., 2020)	✓	✓	✓	
	Region-based Adversarial Training	DNE (Zhou et al., 2020)			✓	✓
		ASCC (Dong et al., 2021)			✓	
Certified Defense	Interval Bound Propagation	LSTM-based (Jia et al., 2019)				
		Transformer-based (Shi et al., 2020)				
	Randomized Smoothing	SAFER(Ye et al., 2020)			✓	✓
		RanMASK (Zeng et al., 2021)		✓	✓	✓

Table 1: The comparison of different defense algorithms. We use "norm-bounded perturbations" to denote whether the perturbations to word embeddings are norm-bounded, "synonyms-agnostic" to whether the defense algorithms rely on pre-defined synonym sets, "structure-free" to whether the defense methods can only be applied to specific network architecture, and "ensemble-based" to whether the ensemble method is required to produce results.

2020; Yang et al., 2020) and combinatorial optimization algorithms (Alzantot et al., 2018; Zang et al., 2020). Although the latter usually can fool a model more successfully, they are time-consuming and require massive amount of queries. This is especially unfair to defenders, because almost no model can guarantee high prediction accuracy in the case of large-scale queries. Therefore, we must impose constraints on the attack algorithm before we systematically evaluate the performance of the defense algorithms, which will be discussed in Section 3.

2.3 Textual Adversarial Defenses

Many defense methods have been proposed to improve the robustness of models against text adversarial attacks. Most of these methods focus on defending against adversarial word substitution attack (Ye et al., 2020). According to whether they possess provably guaranteed adversarial robustness, these methods can roughly be divided into two categories: *empirical* (Zhu et al., 2020; Zhou et al., 2020; Si et al., 2020; Li and Qiu, 2020; Wang et al., 2020; Dong et al., 2021) and *certified defense* (Jia et al., 2019; Ye et al., 2020; Zeng et al., 2021) methods. Table 1 demonstrates detailed categories of these defense methods.

Adversarial Data Augmentation (ADA) is one of the most effective empirical defenses (Ren et al., 2019a; Jin et al., 2020a; Li et al., 2020) for NLP models. However, ADA is extremely insufficient due to the enormous perturbation search space, which scales exponentially with the length of input text. To cover much larger proportion of the perturbation search space, Si et al. (2020) proposed Mix-ADA, a mixup-based (Zhang et al., 2017) augmen-

tation method. **Region-based adversarial training** (Zhou et al., 2020; Dong et al., 2021) improves a models' robustness by optimizing its performance within the convex hull (Region) formed by embeddings of a word and its synonyms. **Adversarial training** (Madry et al., 2018; Zhu et al., 2020; Li and Qiu, 2020; Wang et al., 2020) incorporates a min-max optimization between adversarial perturbations and the models by adding norm-bounded perturbations to words embeddings. Previous research on norm-bounded adversarial training focused on improving the generalization of NLP models. However, our experimental results showed that these methods can also effectively improve models' robustness while suffering no performance drop on the clean inputs.

It has been experimentally shown that the above empirical methods can defend against attack algorithms. However, they can not provably guarantee that their predictions are always correct even under more sophisticated attackers. Recently, a set of certified defense methods has been introduced for NLP models, which can be divided into two categories: **Interval Bound Propagation** (IBP) (Jia et al., 2019; Huang et al., 2019; Shi et al., 2020; Xu et al., 2020) and **randomized smoothing** (Ye et al., 2020; Zeng et al., 2021) methods. IBP-based methods depend on the knowledge of model structure because they compute the range of the model output by propagating the interval constraints of the inputs layer by layer. Randomized smoothing-based methods, on the other hand, are structure-free; they constructs stochastic ensembles to input texts and leverage the statistical properties of the ensemble to provably certify the robustness. All certified defense methods except RanMASK (Zeng et al.,

2021) are based on an assumption that the defender can access the synonyms set used by the attacker. Experimental results show that under the same settings, e.g., without accessing the synonyms set, RanMASK achieves the best defense performance among these certified defenders.

3 Constraints on Adversarial Example Generation

In this section, we first introduce constraints of textual adversarial attacks that should be imposed to ensure the quality of adversarial examples generated, which can help us benchmark textual defense. Then we introduce the datasets for experiments and pick out the optimal hyper-parameters for each constraint.

3.1 The Constraints on Adversaries

To ensure the quality of adversarial examples generated, we impose constraints on textual attack algorithms in the following four aspects:

- The minimum semantic similarity ε_{min} between original input x and adversarial example x'.
- The maximum number of one word's synonyms K_{max}.
- The maximum percentage of modified words ρ_{max}.
- The maximum number of queries to the victim model Q_{max}.

Semantic Similarity In order for the generated adversarial examples to be undetectable by human, we need to ensure that the perturbed sentence is semantically consistent with the original sentence. This is usually achieved by imposing a semantic similarity constraint, see Eq. (1). Most adversarial attack methods (Li et al., 2018; Jin et al., 2020b; Li et al., 2020) use Universal Sentence Encoder (USE) (Cer et al., 2018) to evaluate semantic similarity. USE first encodes sentences into vectors and then uses cosine similarity score between vectors as an approximation of the semantic similarity between the corresponding sentences. Following the setting in Jin et al. (2020a), we set the default value of minimum semantic similarity ε_{min} to be 0.84 (Morris et al., 2020a).

Size of Synonym Set For a word w and its synonym set S_w, we denote the size of elements in S_w as $K = |S_w|$. The value of K influences the search space of attack methods. A larger K increases success rate of the attacker (Morris et al., 2020b).

However, larger K would result in the generation of lower-quality adversarial examples since there is no guarantee that these K words are all synonyms of the same word, especially when the GloVe vectors are used to construct a word's synonyms set (Jin et al., 2020a). While setting the maximum value of K in attack algorithms, we control other variables and select the optimal value K_{max} that keeps attack success rate of the attack algorithm from decreasing too much, seeing Section 3.2 for more details.

Percentage of Modified Words For an input text $x = w_1, \ldots, w_L$, whose length is L, and its adversarial examples $x' = w'_1, \ldots, w'_L$, the percentage of modified words is defined as:

$$\rho = \frac{\sum_{i=1}^{L} \mathbb{I}\{w_i \neq w'_i\}}{L}, \tag{2}$$

where $\sum_{i=1}^{L} \mathbb{I}\{x_i \neq x'_i\}$ is the Hamming distance, with $\mathbb{I}\{\cdot\}$ being the indicator function. An attacker is not allowed to perturb too many words since texcessive perturbation of words results in lower similarity between perturbed and original sentences. However, most existing attack algorithms do not limit the modification ratio ρ, and sometimes even perturb all words in a sentence to ensure the attack success rate. Since it is too difficult for defense algorithms to resist such attacks, we restrain a maximum value of ρ. Similar to the method adopted when setting K_{max}, we use control variable to select the optimal value ρ_{max}, which will be discussed in Section 3.2.

Number of Queries Some existing attack algorithms achieve high attack success rate through massive queries to the model (Yoo et al., 2020). In order to build a practical attack, we placed constraint on query efficiency. Considering the difficulty of defense and the time cost of benchmarking, we need to restrict the number of queries for attackers to query the victim model. At present, most representative attack algorithms are based on greedy search strategies (see Section 2.1). Experiments have shown that these greedy algorithms are sufficient to achieve a high attack success rate (Yang et al., 2020). For a greedy-based attack algorithm, assuming the size of its synonyms set is $K = |S_w|$. Then its search complexity is $O(K \times L)$, where L is the length of input text x, since the greedy algorithm guarantees that each word in the sentence is replaced at most once. Thus, we set the maximum number of queries to the product of K_{max} and sentence length L in default, $Q_{max} = K_{max} \times L$.
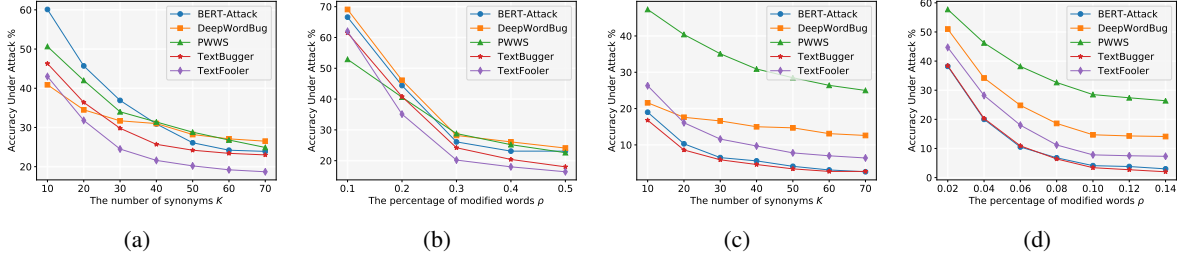
Figure 1: The accuracy under attack of five representative greedy search-based attack algorithms with different settings of constraints. Sub-figures (a) and (b) show the impacts of two constraints (the size of synonyms set K and percentage of modified words ρ) on AGNEWS, while sub-figures (c) and (d) show the impacts of the same two constraints on IMDB.

3.2 Datasets and Hyper-parameters

We conducted experiments on two widely used datasets: the AG-News corpus (AGNEWS) (Zhang et al., 2015) for text classification task and the Internet Movie Database (IMDB) (Maas et al., 2011) for sentiment analysis task.

In order to pick the optimal value K_{max} and ρ_{max} for each dataset, we choose 5 representative adversarial word substitution algorithms: PWWS (Ren et al., 2019a), TextBugger (Li et al., 2018), TextFooler (Jin et al., 2020a), DeepWordBug (Gao et al., 2018), BERT-Attack (Li et al., 2020). All of them are greedy search based attack algorithms [1]. All attackers use K nearest neighbor words of GloVe vectors (Pennington et al., 2014) to generate a word's synonyms except DeepWordBug, which performs character-level perturbations by generating K typos for each word; and BERT-Attack, which dynamically generates synonyms by BERT (Devlin et al., 2018). We use BERT as baseline model, and implementations are based on TextAttack framework (Morris et al., 2020a).

When selecting the optimal K_{max} value for AG-NEWS, we first control other variables unchanged, e.g., the maximum percentage of modified words $\rho_{max} = 0.3$, and conduct experiments on AG-NEWS with different K values. As we can see from Figure 1(a), as K increases, the accuracy under attack decreases. The decline of the accuracy under attack is gradually decreasing. For $K \geq 50$, the decline in accuracy under attack becomes minimal, thus we pick $K_{max} = 50$. Through the same process, we determine the optimal values $\rho_{max} = 0.1$, $K_{max} = 50$ for IMDB dataset as

shown in Figure 1(c) and 1(d), and $\rho_{max} = 0.3$ for AGNEWS dataset as shown in Figure 1(b).

In conclusion, we impose four constraints on attack algorithms to better help with evaluation of different textual defenders. We set $\rho_{max} = 0.3$ for AGNEWS and $\rho_{max} = 0.1$ for IMDB. Such setting is reasonable because the average sentence length of IMDB (208 words) is much longer than that of AGNEWS (44 words). For other constraints, we set $K_{max} = 50$, $\varepsilon_{min} = 0.84$, $Q_{max} = K_{max} \times L$. We choose 3 base attackers to benchmark the defense performance of textual defenders: TextFooler, BERT-Attack, and TextBugger. Our choice of attackers is based on their outstanding attack performances, as shown in Figure 1.

4 Experiments on Textual Defense

4.1 Evaluation Metrics

Under the unified setting of the above-mentioned adversarial attacks, we conducted experiments on the current existing defense algorithms on AG-NEWS and IMDB. We present 4 metrics to measure the defense performance.

- The *clean accuracy* (**Clean**%) is model's classification accuracy on the clean test dataset.
- *Accuracy under attack* (**Aua**%) is the model's prediction accuracy under specific adversarial attack methods.
- *Attack success rate* (**Suc**%) is the number of texts successfully perturbed by an attack algorithm divided by the number of all texts attempted.
- *Number of Queries* (**#Query**) is the average number of times the attacker queries the model. This is another important metric for evaluating robustness of defenders, since the greater the average query number needed for attacker, the more difficult the defense model is to be compromised.

---

[1] We also conducted experiments on combinatorial optimization attacker: e.g., GA (Alzantot et al., 2018), PSO (Zang et al., 2020). However, our pre-experiments showed that they were very time-consuming, and their performance were poor under the limit of the maximum number of queries.

Method	Clean%	TextFooler			TextBugger			BERT-Attack		
		Aua%	Suc%	#Query	Aua%	Suc%	#Query	Aua%	Suc%	#Query
Baseline (BERT)	94.5	19.1	79.6	317.4	23.5	75.0	320.6	27.2	71.0	338.8
Adversarial Data Augmentation	94.4	38.6	58.9	404.6	43.3	53.9	418.3	42.9	54.5	407.0
MixADA (Si et al., 2020)	94.3	37.5	60.3	410.7	36.4	61.4	423.5	39.1	58.6	408.4
PGD-K (Madry et al., 2018)	94.7	24.8	73.9	353.5	26.7	71.9	367.1	39.4	58.5	399.3
FreeLB (Zhu et al., 2020)	94.7	31.6	66.7	382.1	32.9	65.4	390.6	43.9	53.8	417.1
TA-VAT (Li and Qiu, 2020)	94.8	31.0	67.3	382.5	34.2	63.9	415.2	45.0	52.5	436.9
InfoBERT (Wang et al., 2020)	**95.1**	31.8	66.5	369.9	36.3	61.8	391.6	42.4	55.3	392.8
DNE (Zhou et al., 2020)	93.9	28.7	69.8	367.9	28.2	70.3	377.6	42.4	55.5	470.1
ASCC (Dong et al., 2021)	92.3	28.2	69.6	326.5	37.0	60.1	307.4	32.7	64.7	337.1
SAFER (Ye et al., 2020)	94.3	31.8	66.1	350.1	41.2	56.1	398.8	39.3	58.2	373.5
RanMASK (Zeng et al., 2021)	91.7	37.9	58.7	**583.4**	45.0	50.9	**626.8**	49.5	46.1	**661.8**
FreeLB++	**95.1**	**51.5**	**46.0**	419.1	**55.9**	**41.4**	416.9	41.8	56.2	386.1

Table 2: The experiment results of different defenders on AGNEWS, where all models are trained on BERT. The best performance is marked in **bold**. FreeLB++ not only achieves best defense performance under both TextBugger and TextFooler, but also improves **Clean**%. Although RanMASK has also achieved significant defense performance, it drops a lot in **Clean**%.

Method	Clean%	TextFooler			TextBugger			BERT-Attack		
		Aua%	Suc%	#Query	Aua%	Suc%	#Query	Aua%	Suc%	#Query
baseline (BERT)	92.1	10.3	88.8	488.2	5.3	94.3	438.5	5.8	93.7	412.4
Adversarial Data Augmentation	91.9	19.0	79.5	837.1	16.1	82.6	910.7	7.4	92.0	436.7
MixADA (Si et al., 2020)	91.9	19.0	79.6	523.0	11.5	87.6	518.7	7.6	91.8	417.47
PGD-K (Madry et al., 2018)	**93.2**	26.0	72.3	577.5	18.9	79.9	624.9	21.0	77.6	525.8
FreeLB (Zhu et al., 2020)	93.0	29.4	68.3	605.0	22.9	75.3	586.8	21.7	76.6	532.4
TA-VAT (Li and Qiu, 2020)	93.0	28.2	69.7	606.2	22.8	75.5	681.0	19.2	79.4	486.5
InfoBERT (Wang et al., 2020)	92.0	19.2	79.2	541.8	12.7	86.3	491.1	11.3	87.8	447.9
DNE (Zhou et al., 2020)	90.4	28.0	68.2	1222.5	26.5	69.5	1488.0	27.0	69.1	1101.0
ASCC (Dong et al., 2021)	87.8	19.4	77.8	646.1	14.1	83.9	542.5	11.0	87.4	463.2
SAFER (Ye et al., 2020)	91.5	39.5	57.8	1701.7	40.0	57.5	2372.2	38.5	58.8	1363.5
RanMASK (Zeng et al., 2021)	92.3	22.0	74.6	**1493.4**	18.0	79.2	1794.9	36.0	58.4	**1813.1**
FreeLB++	**93.2**	**45.3**	**51.0**	1025.9	**42.9**	**53.6**	1094.0	**39.9**	**56.9**	696.9

Table 3: The experiment results of different defenders on IMDB. FreeLB++ surpasses all existing defense methods by a large margin under all attackers, even though defending against adversarial attacks in IMDB is harder than that in AGNEWS because sentences from IMDB are far longer than those from AGNEWS.

A good defense method should have higher clean accuracy, higher accuracy under attack, lower attack success rate, and requires larger number of queries for attack.

4.2 Implementation Details

Our reproduction of all defense methods, along with the hyper-parameter settings, are completely based on their original papers, except for the following two constraints: (1) For methods which are not synonyms-agnostic, we establish different synonym sets for both attackers and defender. (2) For methods that are ensemble-based, we use the "logit-summed" ensemble method introduced in (Devvrit et al., 2020) to make final predictions. Specifically, we use the counter-fitting vectors (Mrkšić et al., 2016) to generate the synonym set for attackers, and use vanilla Glove Embedding (Pennington et al., 2014) to generate synonym set for defenders[2]. Following Devvrit et al. (2020); Zeng et al.

(2021), we take the average of logits produced by the base classifier over all randomly perturbed input sentences, whose size is denoted as C, as the final prediction. For AGNEWS, we set the value of C to 100, while for IMDB, the value of C is default 16. In the implementation of FreeLB++, we remove the constraints of norm bounded projection, and set step size as 30 and 10 on AGNews and IMDB datasets respectively. More details will be introduced in Section 4. All the hyper-parameter settings are tuned on a randomly chosen development dataset.

We use BERT (Devlin et al., 2018) as our base model. Clean accuracy (**Clean**%) is tested on the whole test dataset, while the latter three metrics, e.g., **Aua**% are evaluated on 1000 randomly chosen samples from the test dataset.

4.3 Results

As we can see from Table 2, (1) the ADA-based methods have a small decrease in clean accuracy, but excellent accuracy under attack. However, com-

---

[2]According to our statistics, 69.70% of the words in defender's synonym set appears in the attacker synonym set's vocabulary. Among them, 73% of the synonyms in the de-

fender's synonym set are covered by the attacker's.

paring with the remaining methods, ADA-based methods need to know specific attacker algorithms to generate adversarial examples before defending. (2) The adversarial training methods, e.g., FreeLB, achieve higher clean accuracy than the baseline, and their improvement in robustness is also very insignificant (Li and Qiu, 2020; Zeng et al., 2021). Interestingly, once we remove the l_2-norm bounded limitation for FreeLB, we find out that defense performance is significantly improved (see FreeLB++ in the tables). FreeLB++ surpasses all existing defense methods by a large margin under TextFooler and TextBugger attacks. We will leave more discussions about adversarial training methods in Section 5.1. (3) The region-based adversarial training methods, e.g., DNE, perform poorly on both clean accuracy and accuracy under attack. It is mainly because the synonym set used in the attack method is different from that used in DNE, which is further discussed in Section 5.3. (4) The certified defense methods achieve high defense performance. It is worth noting that the average number of queries to the model of these methods is larger. We think the improvement of robustness comes from the ensemble method, seeing further discussions in Section 5.2.

Results of defense performance on IMDB are reported in Table 3. Defense methods share the trends with performance on AGNEWS. However, the general robustness of models on IMDB is poorer than AGNEWS. It is probably because the average length of sentences in IMDB (208 words) is far longer than that in AGNEWS (44 words). Longer sentences implies a larger search space for attackers, making it more difficult for defenders to defend against attacks.

5 Discussions

5.1 Effectiveness of Adversarial Training

The objective of standard adversarial training methods, e.g., PGD-K (Madry et al., 2018) and FreeLB (Zhu et al., 2020) is to minimize the maximum risk for perturbation $\boldsymbol{\delta}$ within a small ϵ-norm ball:

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}} \left[\max_{\|\boldsymbol{\delta}\|\leq\epsilon} \mathcal{L}(f_{\theta}(\boldsymbol{X}+\boldsymbol{\delta}),y) \right], \quad (3)$$

where $\mathcal{D}$ is the data distribution, $\boldsymbol{X}$ is the embedding representations of input sentence $\boldsymbol{x}$, y is the gold label, and $\mathcal{L}$ is the loss function for training neural networks, whose parameters is denoted as θ. In order to solve inner maximization, projected

gradient descent (PGD) algorithm is applied as descrided in Madry et al. (2018) and Zhu et al. (2020):

$$\boldsymbol{\delta}_{t+1} = \prod_{\|\boldsymbol{\delta}\|_F \leq \epsilon} \left(\boldsymbol{\delta}_t + \alpha \frac{g(\boldsymbol{\delta}_t)}{\|g(\boldsymbol{\delta}_t)\|_F} \right), \quad (4)$$

where $g(\boldsymbol{\delta}_t) = \nabla_{\boldsymbol{\delta}} \mathcal{L}(f_{\theta}(\boldsymbol{X}+\boldsymbol{\delta}),y)$ is the gradient of the loss with respect to $\boldsymbol{\delta}$, $\prod_{\|\boldsymbol{\delta}\|_F \leq \epsilon}$ performs a projection onto the ϵ-Frobenius norm ball, and t is the number of ascent steps to find the "worst-case" perturbation $\boldsymbol{\delta}$ with step size α.

Method	Norm ϵ	Clean%	TextFooler		BERT-Attack	
			Aua%	Suc%	Aua%	Suc%
PGD-K	0.01	94.9	21.8	77.0	31.5	66.8
	0.1	95.3	43.6	54.3	45.1	52.7
	1	**95.2**	**45.2**	**55.3**	**45.3**	**52.4**
	w/o	95.2	45.2	55.3	45.3	52.4
FreeLB	0.01	95.4	30.5	68.0	43.6	54.3
	0.1	95.5	36.1	62.2	40.0	58.1
	1	**94.9**	**45.8**	**51.7**	**42.5**	**55.2**
	w/o	94.9	45.8	51.7	42.5	55.2

Table 4: The impact of different values of norm ϵ on both clean and defense performance, where "w/o" means updating $\boldsymbol{\delta}$ without projection onto a ϵ-Frobenius norm ball. A large value $\epsilon = 1$ is equivalent to removing the norm-bounded projection, both of which achieve the best **Clean**% and **Aua**%.

In this section, we first study the influence of the value of the norm ϵ on the model's robustness performance, which is also discussed by Gowal et al. (2020) in computer vision field. As can be seen from Table 4, we find out that both of the **Clean**% and **Aua**% increase as ϵ increases. Note that the value of ϵ is usually set to a very small value, e.g., $\epsilon = 0.01$ (Zhu et al., 2020). A large value ϵ (e.g., $\epsilon = 1$ in Table 4) is equivalent to removing the norm-bounded limitation (seeing "w/o" in Table 4), because when ϵ is large enough and the step size α is fixed, the magnitude of perturbation that used to update $\boldsymbol{\delta}$ is also fixed, seeing Eq. (4). In this case, from $\left\| \frac{g(\boldsymbol{\delta}_t)}{\|g(\boldsymbol{\delta}_t)\|_F} \right\| \leq 1$ and Eq. (4), we have:

$$\|\boldsymbol{\delta}_{t+1}\| \leq \|\boldsymbol{\delta}_t\| + \left\| \alpha \frac{g(\boldsymbol{\delta}_t)}{\|g(\boldsymbol{\delta}_t)\|_F} \right\| \leq \|\boldsymbol{\delta}_t\| + \alpha. \quad (5)$$

Thus, with multi-step updating $\boldsymbol{\delta}$, we have:

$$\begin{aligned} \|\boldsymbol{\delta}_t\| &\leq \|\boldsymbol{\delta}_{t-1}\| + \alpha \leq \|\boldsymbol{\delta}_{t-2}\| + 2\alpha \\ &\leq \cdots \leq \|\boldsymbol{\delta}_1\| + (t-1)*\alpha \leq t\alpha, \end{aligned} \quad (6)$$

where we can find that the upper bound of the norm of perturbation $\boldsymbol{\delta}$ is determined by the number of ascent steps t if the value of α is fixed. In other words, the number of ascent steps t influences the search region of the perturbation $\boldsymbol{\delta}$, where the larger t is, the larger the search region will be. However, in original FreeLB, the same ϵ-norm has been applied to all perturbations and to restrict the search
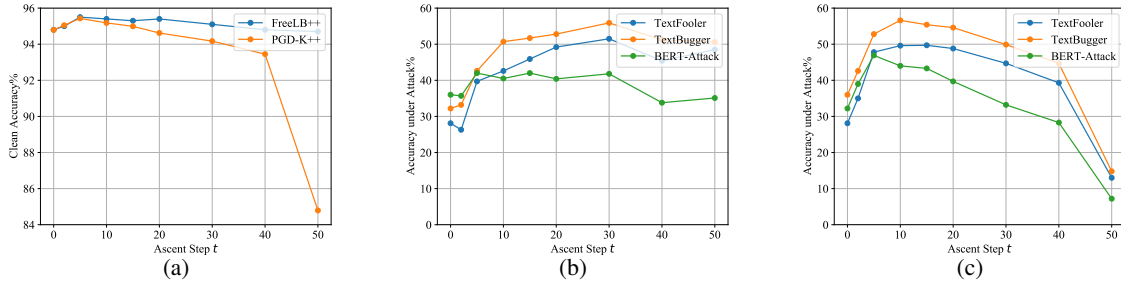
Figure 2: The impact of different values of the ascent steps t. Sub-figure (a) shows the accuracy of FreeLB++ and PGD-K++ on the clean data (**Clean%**). Sub-figure (b) and (c) show the accuracy of FreeLB++ and PGD-K++ respectively under three different attack algorithms (TextFooler, TextBugger and BERT-Attack). As the value of t grows, **Clean%** and **Aua%** of the models will increase until reaching their peak values. After that, they begin to decrease as the value of t increases continually.

region around every word embedding. We denote our versions of PGD-K and FreeLB which remove the norm-bounded limitations as PGD-K++ and FreeLB++, respectively. We conducted experiments on PGD-K++, FreeLB++ with different t to study the impact of the value of t. As shown in Figure 2, the **Clean%** of both PGD-K++ and FreeLB++ models reaching a peak at $t = 5$ while the peak of **Aua%** performance reaches at $t = 30$ for FreeLB++ and $t = 10$ for PGD-K++.

We give a possible explanation to this improvement of performance. We regard standard adversarial training as an exploration of embedding space. When t is small, the adversarial example space explored by the model is relatively small, resulting in poor defense performance of the model when a high-intensity attack arrives. This problem is alleviated when t becomes larger, and this explains why both **Clean%** and **Aua%** can be improved when t increases. When t exceeds its optimal value, the adversarial example generated by the algorithm may become dissimilar to the original example. Excessive learning of examples with different distributions from the original examples will lead to a decline in model's modeling ability.

5.2 Impact of Ensemble Strategies

There are two ensemble strategies (Devvrit et al., 2020): *logits-summed* (logit) and *majority-vote* (voting) ensemble. As mentioned above, in the logit method, the logits produced by the base classifier are averaged. Whereas, in the voting strategy, the predictions of classifiers for each class label are counted, and the vote results will be regarded as the output probability for classification. Compared to the logit method, we found that the majority-vote strategy can effectively improve the model's robustness, as can be inferred from the results in Table 5. However, after further research, the reason the vot-

ing strategy achieves better defense performance is that it increases the difficulty for score-based attackers, which is also discussed in Zeng et al. (2021).

A typical score-based attacker usually involves two key steps: searching for weak spots in a text and replacing words in these weak spots to maximum model's prediction error. In the second stage, if no words in the synonym set can lower the logits, the adversary will give up perturbing this word. However, for those voting-based methods which create ensemble by introducing small noise to the original text x, e.g., SAFER, RanMASK-5%, the models tend to output very sharp distribution, even close to one-hot categorical distribution. This forces the attackers to launch decision-based attacks instead of the score-based ones, which can dramatically improve their attack difficulty. Therefore, it may be unfair to compare voting-based ensemble defense methods with others due to lack of effective ways to attack voting-based ensembles in the literature. We believe voting-based methods will greatly improve model's defense performance, but we recommend using logit-summed algorithm if one needs to prove the effectiveness of the proposed algorithm against adversarial attacks in future research.

Method	Clean	TextFooler		TextBugger	
		Aua%	Suc%	Aua%	Suc%
Baseline (BERT)	**94.5**	19.1	79.6	27.2	71.0
SAFER (logit)	94.3	31.8	66.1	41.2	56.1
SAFER (voting)		**78.6**	**17.6**	**69.0**	**28.0**
RanMASK-5% (logit)	**94.5**	21.5	77.3	41.2	56.1
RanMASK-5% (voting)		68.6	26.9	62.5	34.0
RanMASK-90% (logit)	91.7	37.9	58.7	49.5	46.1
RanMASK-90% (voting)		47.9	48.2	57.4	37.6

Table 5: The ablation experiment on ensemble methods. Voting-based ensembles achieve better performance than logit-based ensembles, but this is potentially due to the non-differentiability introduced by voting-based attacks.

Method	Clean	TextFooler		TextBugger	
		Aua%	Suc%	Aua%	Suc%
SAFER (w)	93.8	**46.6**	**50.3**	**55.4**	**40.9**
SAFER (w/o)	**94.3**	31.8	66.1	41.2	56.1
DNE (w)	93.4	**44.9**	**54.0**	**42.2**	**56.6**
DNE (w/o)	**93.9**	28.7	69.8	28.2	70.3
ASCC (w)	91.4	**39.0**	**57.3**	**44.2**	**51.6**
ASCC (w/o)	**92.3**	28.2	69.6	37.0	60.1

Table 6: The ablation experiment on synonym set. "w" and "w/o" means the corresponding defense method use or not use the synonym set of the attack method.

5.3 Impact of Synonym Sets

Table 6 shows the results of the ablation study on the impact of external synonym set on performance in the defense methods. Some previous studies (Ye et al., 2020; Zhou et al., 2020; Dong et al., 2021) use the same synonym set as the attacker during adversarial defense training, leading to significant defense performance. As we can see from Table 6, all methods improve the **Aua**% by a large margin after sharing the synonym set with the attacker. However, having access to the attacker's synonym is not a realistic scenario since we cannot impose a limitation on the synonym set used by the attackers. Thus, for the sake of fair comparison in future research, we suggest that future work should assume that the attacker's synonym set cannot be accessed, and report the defense performance in this case.

6 Conclusion

In this paper, we established a comprehensive and coherent benchmark to evaluate the defense performance of textual defenders. We impose constraints to existing attack algorithms to ensure the quality of adversarial examples generated. Using these attackers, we systematically studied the advantages and disadvantages of different textual defenders. We find out that adversarial training methods are still the most effective defenders. Our FreeLB++ can not only achieve state-of-the-art defense performance under various attack algorithms, but also improve the performance on clean examples. We hope this study could provide useful clues for future research on text adversarial defense.

Acknowledgements

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2890–2896, Brussels, Belgium. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Devvrit, Minhao Cheng, Cho-Jui Hsieh, Inderjit Dhillon, et al. 2020. Voting based ensemble improves robustness of defensive models. *arXiv preprint arXiv:2011.14031*.

Xinshuai Dong, Hong Liu, Rongrong Ji, and Anh Tuan Luu. 2021. Towards robustness against natural language word substitutions. In *International Conference on Learning Representations (ICLR)*.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2017. Hotflip: White-box adversarial examples for text classification. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 50–56. IEEE.

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.

Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. 2020. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*.

Yu-Lun Hsieh, Minhao Cheng, Da-Cheng Juan, Wei Wei, Wen-Lian Hsu, and Cho-Jui Hsieh. 2019. On the robustness of self-attentive models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1520–1529.

Po-Sen Huang, Robert Stanforth, Johannes Welbl, Chris Dyer, Dani Yogatama, Sven Gowal, Krishnamurthy Dvijotham, and Pushmeet Kohli. 2019. Achieving verified robustness to symbol substitutions via interval bound propagation. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*.

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. 2020a. Is BERT really robust? a strong baseline for natural language attack on text classification and entailment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8018–8025.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020b. Is BERT really robust? a strong baseline for natural language attack on text classification and entailment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8018–8025.

Volodymyr Kuleshov, Shantanu Thakoor, Tingfung Lau, and Stefano Ermon. 2018. Adversarial examples for natural language classification problems. *International Conference on Learning Representations (ICLR)*.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2018. Textbugger: Generating adversarial text against real-world applications. *Network and Distributed Systems Security (NDSS) Symposium 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. 2020. Tavat: Token-aware virtual adversarial training for language understanding. *arXiv preprint arXiv:2004.14543*.

Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*, pages 142–150. Association for Computational Linguistics.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *1st International Conference on Learning Representations (ICLR)*.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020a. 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.

John X Morris, Eli Lifland, Jack Lanchantin, Yangfeng Ji, and Yanjun Qi. 2020b. Reevaluating adversarial examples in natural language. *arXiv preprint arXiv:2004.14174*.

Nikola Mrkšić, Diarmuid O Séaghdha, Blaise Thomson, Milica Gašić, Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. *Proceedings of HLT-NAACL*.

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

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019a. 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.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019b. 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.

Zhouxing Shi, Huan Zhang, Kai-Wei Chang, Minlie Huang, and Cho-Jui Hsieh. 2020. Robustness verification for transformers. *arXiv preprint arXiv:2002.06622*.

Chenglei Si, Zhengyan Zhang, Fanchao Qi, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. 2020. Better robustness by more coverage: Adversarial training with mixup augmentation for robust fine-tuning. *arXiv preprint arXiv:2012.15699*.

Jinsong Su, Zhenqiao Song, Yaojie Lu, Mu Xu, Changxing Wu, and Yidong Chen. 2018. Exploring implicit semantic constraints for bilingual word embeddings. *Neural Processing Letters*, 48(2):1073–1088.

Boxin Wang, Shuohang Wang, Yu Cheng, Zhe Gan, Ruoxi Jia, Bo Li, and Jingjing Liu. 2020. Infobert: Improving robustness of language models from an information theoretic perspective. *9th International Conference on Learning Representations (ICLR)*.

Kaidi Xu, Zhouxing Shi, Huan Zhang, Yihan Wang, Kai-Wei Chang, Minlie Huang, Bhavya Kailkhura, Xue Lin, and Cho-Jui Hsieh. 2020. Automatic perturbation analysis for scalable certified robustness and beyond. *Advances in Neural Information Processing Systems*, 33.

Puyudi Yang, Jianbo Chen, Cho-Jui Hsieh, Jane-Ling Wang, and Michael I Jordan. 2020. Greedy attack and gumbel attack: Generating adversarial examples for discrete data. *Journal of Machine Learning Research*, 21(43):1–36.

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.

Jin Yong Yoo, John X Morris, Eli Lifland, and Yanjun Qi. 2020. Searching for a search method: Benchmarking search algorithms for generating nlp adversarial examples. *arXiv preprint arXiv:2009.06368*.

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

Jiehang Zeng, Xiaoqing Zheng, Jianhan Xu, Linyang Li, Liping Yuan, and Xuanjing Huang. 2021. Certified robustness to text adversarial attacks by randomized [MASK]. *arXiv preprint arXiv:2105.03743*.

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

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

Xiaoqing Zheng, Jiehang Zeng, Yi Zhou, Cho-Jui Hsieh, Minhao Cheng, and Xuanjing Huang. 2020. Evaluating and enhancing the robustness of neural network-based dependency parsing models with adversarial examples. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6600–6610, Online. Association for Computational Linguistics.

Yi Zhou, Xiaoqing Zheng, Cho-Jui Hsieh, Kai-wei Chang, and Xuanjing Huang. 2020. Defense against adversarial attacks in nlp via dirichlet neighborhood ensemble. *arXiv preprint arXiv:2006.11627*.

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*.