Synonym-unaware Fast Adversarial Training against Textual Adversarial Attacks

Yichen Yang[*], **Xin Liu**[*] **,Kun He**[†]
School of Computer Science and Technology
Huazhong University of Science and Technology, Wuhan, China
yangyc@hust.edu.cn, liuxin_jhl@hust.edu.cn, brooklet60@hust.edu.cn

Abstract

Numerous adversarial defense methods have been proposed to strengthen the robustness of Natural Language Processing (NLP) models against adversarial attacks. However, many of these methods rely on predetermined linguistic knowledge and assume that attackers' synonym candidates are known, which is often unrealistic. In this work, we investigate adversarial training in the embedding space and introduce a Fast Adversarial Training (FAT) method to improve the model robustness without requiring synonym awareness. FAT leverages single-step perturbation generation and effective perturbation initialization based on two key insights: (1) adversarial perturbations generated by single-step and multi-step gradient ascent are similar, and (2) perturbations generated on the same training sample across successive epochs exhibit resemblance. By employing single-step gradient ascent and leveraging historical perturbation information, FAT not only expedites the training process but also efficiently initializes perturbations. Extensive experiments demonstrate that FAT significantly enhances the robustness of popular NLP models under scenarios where synonyms are unknown, outperforming other defense baselines under various character-level and word-level attacks.

1 Introduction

Deep neural networks have been demonstrated to be vulnerable to adversarial examples (Szegedy et al., 2014; Goodfellow et al., 2015; Papernot et al., 2016), which are crafted by adding imperceptible perturbations to the benign examples. For Natural Language Processing (NLP) models, adversarial attacks can be categorized into three types based on the granularity of the perturbations: character-level (Gao et al., 2018; Ebrahimi et al., 2018), word-level (Ren et al., 2019; Alzantot et al., 2018; Zang et al., 2020; Li et al., 2020), and sentence-level attacks (Wang et al., 2020). Among them, word-level attacks based on synonym substitutions are most commonly used, as they guarantee the correct syntax, preserve unchanged semantics, and have a high attack success rate. From another perspective of model visibility, adversarial attacks fall into two categories: white-box attacks and black-box attacks. White-box attacks (Papernot et al., 2016; Guo et al., 2021) have direct access to the model parameters, embeddings and gradients, while black-box attacks (Jin et al., 2020; Li et al., 2020; Lv et al., 2023) can only access the model outputs to generate adversarial examples, which are more practical.

Numerous defense methods have been proposed to enhance the model's robustness against adversarial attacks based on synonym substitutions. However, we have observed that most of these methods are synonym-aware, meaning they assume some or all of the synonym substitutions used by the attackers are known beforehand during training, which is often unrealistic. Attackers have various approaches to obtain synonym candidates, such as artificially formulating the embedding distance (Alzantot et al., 2018; Jin et al., 2020), retrieving synonyms through the online thesaurus (Ren et al., 2019; Zang et al., 2020), or inferring by language models (Li et al., 2020). In this way, they could obtain different synonyms and generate various adversarial texts. As exhibited in the experiments of Li et al. (2021) and Wang et al. (2023), the effectiveness of defense methods can be significantly reduced if the defender's synonym candidates do not align with those used by the attacker. Given the wide range of potential synonym settings in textual adversarial attacks, it is prudent to design defense methods that are independent of the attacker's specific synonym choices. Therefore, we focus on a more practical scenario where the defense method does not rely on predetermined synonym information or any linguistic knowledge

[*]The first two authors contribute equally.
[†]Corresponding author.

Findings of the Association for Computational Linguistics: NAACL 2025, pages 727–739
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

beyond the dataset.

We rethink the Adversarial Training (AT) methods in the synonym-unaware scenario. As a typical kind of defense method to improve the model's robustness, most AT methods, such as ATFL (Wang et al., 2021c) and BFF (Ivgi and Berant, 2021), work in the input space and utilize a specific adversarial attack to generate adversarial texts for model training. They require predetermined synonym information to craft adversarial texts. ASCC (Dong et al., 2021) trains models using virtual adversarial examples constructed by the combination of the embedding representation of synonyms, which still needs the synonym information. In contrast, another type of AT methods (Miyato et al., 2017; Zhu et al., 2020; Liu et al., 2020) works directly in the embedding space and has no need to access to synonyms. They perturb the embedding representation directly and train the model using the perturbed embedding representation. However, their primary goal is to enhance model generalization on the original test dataset by serving as a regularization technique, rather than specifically improving adversarial robustness. Experiments by Liu et al. (2022) and Li and Qiu (2021) have further shown that this type of AT method has limited effectiveness in bolstering robustness.

In this work, we empirically demonstrate that AT in the embedding space could also improve the model's robustness without predetermined synonym knowledge. Generally, the Projected Gradient Decent (PGD) method (Madry et al., 2018) is adopted to generate adversarial perturbations on the embedding representation. However, due to its multi-step gradient ascent process, PGD-AT is highly inefficient for commonly used large-scale pre-trained NLP models such as BERT (Devlin et al., 2019), leading to unsatisfactory performance within a limited time. To address this issue, we propose a Fast Adversarial Training (FAT) method to boost the model's robustness using single-step perturbation generation and initialization based on historical information.

Firstly, we observe that the adversarial perturbations crafted by single-step and multi-step gradient ascent are similar for NLP models. Based on this observation, FAT employs single-step gradient ascent to create perturbations on the embedding representation, rather than relying on multi-step gradient ascent. It significantly accelerates the training process, allowing the model to be trained over more epochs and thereby achieving improved robustness within a limited time. Secondly, we observe that the direction of the perturbations generated on the identical samples in two successive training epochs is similar. To make full use of the historical information, FAT initializes the perturbation along the direction of perturbation generated on the same samples in the previous epoch.

Extensive experiments conducted on four popular benchmark datasets and two models show that our proposed FAT achieves the best robustness under various advanced adversarial attacks. Our main contributions are as follows:

- We introduce Fast Adversarial Training (FAT), informed by our observation on perturbation generation in the embedding space of NLP models. FAT employs single-step gradient ascent for faster training and leverages historical training information to enhance robustness.
- Extensive experiments demonstrate that FAT achieves the best robustness among the defenses, handling attacks with varying model visibility and perturbation granularity.
- Given the diverse settings of synonym candidates and perturbation budgets in textual adversarial attacks, FAT offers a valuable, easy-to-apply, and effective solution for defense in realistic, synonym-unaware scenarios.

2 Related Work

Many adversarial defense methods have been proposed to boost the model's robustness against adversarial attacks based on synonym substitutions. These methods can be classified into two categories, *i.e.*, synonym-aware methods and synonym-unaware methods.

Most defense methods need to be accessible to the synonyms used by attackers or introduce human-prescribed rules to determine synonyms. Input transformation methods either encode the synonyms to the same code (Wang et al., 2021b) or adopt synonym substitutions (Mozes et al., 2021) to eliminate perturbation in the input space. Additionally Yang et al. (2022) embrace the triplet metric learning to bring words closer to their synonyms while distancing them from non-synonyms in the embedding space. Interval bound propagation methods (Jia et al., 2019; Wang et al., 2023) calculate the interval of all possible perturbed texts based on a particular synonym definition and propagate these interval bounds through the network layers to minimize loss in the worst case. Some certified methods (Zhao et al., 2022; Ye et al., 2020) utilize randomized smoothing to achieve provable robustness. Adversarial training (AT) methods working in the input space (Wang et al., 2021c; Ivgi and Berant, 2021) craft adversarial texts based on synonym substitutions and regard adversarial texts as the training data. It is worth noting that while some AT methods involve the embedding space to generate adversarial perturbation, our work differs from those AT methods. Specifically, ASCC (Dong et al., 2021) trains models with the virtual adversarial examples constructed by the combination of embedding representations of synonyms. However, the above AT methods still rely on the synonyms and differ from our synonym-unaware approach.

Since synonym-unaware defense methods do not rely on the synonyms used by attackers, they align more closely with realistic scenarios and can be easily applied to any language model, allowing for a fairer evaluation of their robustness. Flooding-X (Liu et al., 2022) leverages the Flooding method to improve the model's robustness through a simple training strategy that avoids zero training loss and guide the model into a smooth parameter landscape. InfoBERT (Wang et al., 2021a) introduces two mutual information based regularizers for model training. A series of works (Miyato et al., 2017; Zhu et al., 2020; Li and Qiu, 2021) directly perturb the word embeddings and utilize the perturbed embedding representation to train the model. However, these works regard AT as a regularization strategy and aim to improve the model's generalization on the original dataset rather than adversarial robustness. Li et al. (2021) utilize PGD without projection operation to add a large magnitude of perturbation to the embedding representations for AT, which is the work most similar to ours. In contrast, we utilize single-step gradient ascent to generate adversarial perturbation. Besides, Li et al. (2021) randomly initialize the adversarial perturbation, while we introduce historical information in perturbation initialization to achieve better model's robustness.

With the development of Large Language Models (LLMs) in recent years, there have been some researches on adversarial attacks to manipulate them. These adversarial attacks on the generative LLMs differ from those on BERT models, primarily manifesting as jailbreak attacks through prompt injection (Zhou et al., 2024) and prompt rewriting (Deng et al., 2024; Guo et al., 2024). Therefore, we do not evaluate the effectiveness of our FAT and baseline methods on LLMs.

3 Methodology

This section investigates adversarial training in the embedding space and presents two observations. Based on the two observations, we propose a Fast Adversarial Training (FAT) method to boost the model's robustness.

3.1 Rethinking Adversarial Training

According to the placement of perturbations, AT for NLP models can be classified into two categories, *i.e.*, discrete AT (Ivgi and Berant, 2021; Wang et al., 2021c) and continuous AT (Li et al., 2021; Li and Qiu, 2021). Discrete AT generates adversarial texts within the discrete input space, while continuous AT adds adversarial perturbations to the embedding representation in the continuous embedding space.

Given a dataset $\mathcal{D}$ and a classification model $f_{\boldsymbol{\theta}}(\cdot)$ parameterized by $\boldsymbol{\theta}$, the training objective of the discrete AT could be formulated as:

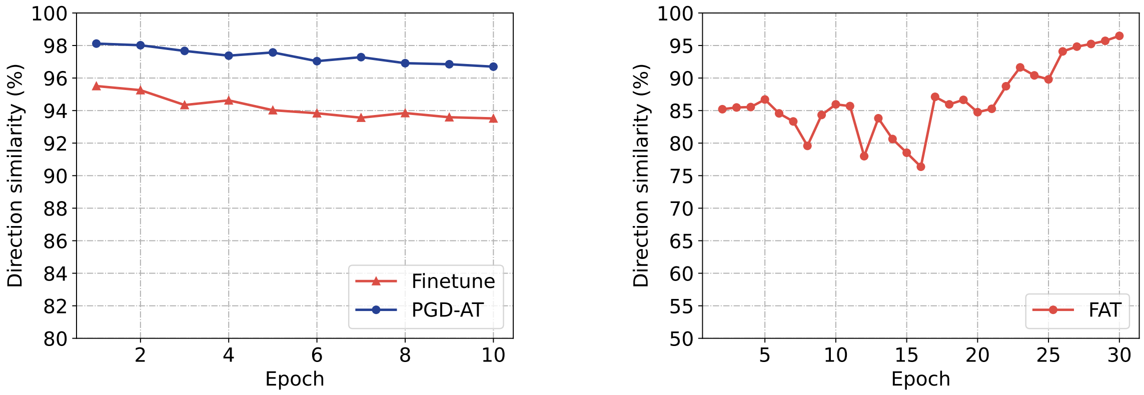
$$\min_{\boldsymbol{\theta}} \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(f_{\boldsymbol{\theta}}(\mathrm{att}(x)), y), \tag{1}$$

where x denotes an input text with true label y, $\mathrm{att}(x)$ is the adversarial text generated by a certain attack method $\mathrm{att}(\cdot)$, and $\mathcal{L}(\cdot,\cdot)$ is the cross-entropy loss. The predetermined human-prescribed linguistic knowledge about synonyms needs to be introduced when generating adversarial texts. When the synonym candidates used in AT and the adversarial attack for evaluation are inconsistent, the performance of AT will decline significantly (Li et al., 2021).

In contrast, continuous AT needs no predetermined linguistic knowledge, and the training objective is:

$$\min_{\boldsymbol{\theta}} \sum_{(x,y)\in\mathcal{D}} \left[\max_{\|\boldsymbol{\delta}\|_p \leq \epsilon} \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{v}(x)+\boldsymbol{\delta}), y) \right], \tag{2}$$

where $\boldsymbol{v}(x)$ denotes the embedding representation of text x, and $\boldsymbol{\delta}$ is the adversarial perturbation added to the embedding representation. $\|\cdot\|_p$ denotes l_p-norm, and ϵ controls the perturbation magnitude. Li et al. (2021) adopt l_2-norm PGD to solve the inner maximization in Eq. (2) for training, which we call PGD-AT for simplicity. For the specific implementation, they remove the projection operation and iteratively conduct multiple



(a)

(b)

Figure 1: (a) The perturbation similarity between single-step and multi-step gradient ascent. (b) The perturbation similarity between the previous and current epochs.

steps of gradient ascent to generate the adversarial perturbation as follows:

$$\boldsymbol{\delta}^0 = \boldsymbol{U}(-\epsilon_0, \epsilon_0)/\sqrt{n \cdot d}, \tag{3}$$

$$\boldsymbol{\delta}^{t+1} = \boldsymbol{\delta}^t + \alpha \cdot \frac{\nabla_{\boldsymbol{\delta}^t}\mathcal{L}(f(\boldsymbol{v}(x) + \boldsymbol{\delta}^t), y)}{\left\|\nabla_{\boldsymbol{\delta}^t}\mathcal{L}(f(\boldsymbol{v}(x) + \boldsymbol{\delta}^t), y)\right\|_2}, \tag{4}$$

where n is the number of words in the input text x and d is the dimension of word embedding. $\boldsymbol{U}(-\epsilon_0, \epsilon_0) \in \mathbb{R}^{n \times d}$ denotes a matrix whose elements are uniformly sampled in range $(-\epsilon_0, \epsilon_0)$. t denotes the current step. ϵ_0 and α are hyperparameters for controlling the magnitude of initial perturbation and step size, respectively.

However, since the current commonly used NLP models are large-scale pre-trained models, such as BERT (Devlin et al., 2019), using PGD attack to generate adversarial examples for AT is inefficient. For instance, PGD-AT with ten attack steps takes about two hours to train a BERT model of the base version for one epoch in the *IMDB* dataset on a single TITAN RTX GPU. Worse still, adversarial examples tend to be more diverse than benign samples. The accuracy of the original testing dataset quickly converges after several epochs of fine-tuning, whereas the robustness of the model requires more epochs.

3.2 Fast Adversarial Training

We propose a Fast Adversarial Training (FAT) method to enhance the defense performance of continuous AT from the perspective of single-step perturbation generation and initialization with historical information.

3.2.1 Single-Step Perturbation Generation

We speculate that it is redundant to adopt multi-step gradient ascent to generate adversarial perturbation for AT on NLP models. For validation, given 1000 random testing samples from the *IMDB* dataset and two trained models of standard fine-tuning and PGD-AT, we use single-step gradient ascent and multi-step gradient ascent, respectively, to generate adversarial perturbations of the random samples on the model checkpoints. We apply the element-wise sign function to the two perturbations as their directions. The direction similarity could be defined as the ratio of the number of dimensions with the same value between the two directions to the total number of dimensions. As illustrated in Figure 1(a), for models trained by standard fine-tuning or PGD-AT, the direction similarity between the perturbations generated by multi-step and single-step gradient ascents is the same over 90% across most dimensions, indicating the redundancy of multi-step generation for NLP adversarial training.

We thereby adopt the single-step gradient ascent to generate adversarial perturbation to boost the efficiency of AT. Specifically, with the initial adversarial perturbation $\boldsymbol{\delta}^0$, the training objective could be formulated as follows:

$$\boldsymbol{\delta} = \boldsymbol{\delta}^0 + \epsilon \cdot \frac{\nabla_{\boldsymbol{\delta}^0}\mathcal{L}(f(\boldsymbol{v}(x) + \boldsymbol{\delta}^0), y)}{\left\|\nabla_{\boldsymbol{\delta}^0}\mathcal{L}(f(\boldsymbol{v}(x) + \boldsymbol{\delta}^0), y)\right\|_2}, \tag{5}$$

$$\min_{\boldsymbol{\theta}} \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{v}(x) + \boldsymbol{\delta}), y). \tag{6}$$

In the AT process of large-scale NLP models such as BERT, most of the time cost is caused by the gradient back-propagation. Assuming that the number of training samples is N, the number of training epochs is E, and the step of adversary generation is T, PGD-AT requires $NE(T+1)$ back-propagation, in which each training sample needs T back-propagation to generate adversarial perturbation in each epoch, and one back propagation to update model parameters. In contrast, FAT requires only $2NE$ back-propagation. Therefore, within a given limited time, FAT can conduct more training epochs and achieve higher robustness.

3.2.2 Perturbation Initialization

As in Eq. (3), previous AT methods (Li and Qiu, 2021; Li et al., 2021) introduce randomness into training data by initializing the perturbation with a small random noise. We argue that introducing useful information for initialization could help craft the adversarial examples from a good point for training and enhance the model's robustness.

For validation, we randomly choose 1000 testing samples from the *IMDB* dataset and use single-step gradient ascent to generate adversarial perturbations for each epoch of FAT. As illustrated in Figure 1(b), we observe that the direction of adversarial perturbation generated on the same training sample in two successive epochs is identical in 77%-97% of the dimensions in the training process of FAT. It indicates that the adversarial perturbation generated in the previous epoch contains helpful information for the current epoch.

To fully use historical perturbation, we propose a new initialization approach for adversary generation. Specifically, we limit the initial perturbation $\boldsymbol{\delta}_0$ to the perturbation direction corresponding to the identical sample in the previous epoch, and the magnitude on each dimension is generated randomly, which could be formulated as follows:

$$\boldsymbol{\delta}^0 = \boldsymbol{U}(0, \epsilon_0) \odot \operatorname{sign}(\boldsymbol{\delta}') / \sqrt{n \cdot d}, \tag{7}$$

where $\odot$ denotes element-wise multiplication, and $\operatorname{sign}(\cdot)$ denotes the element-wise sign function. $\boldsymbol{\delta}'$ is the perturbation of the identical training sample in the previous epoch.

By incorporating information from previous perturbations into the current epoch, we achieve a momentum-like effect that helps stabilize the generation of adversarial examples and enhances the model's robustness. The overall FAT method is summarized in Algorithm 1.

Algorithm 1 The FAT Method

```
Input: Training data $\mathcal{D}$, model $f_{\boldsymbol{\theta}}$, initial perturbation size $\epsilon_0$, perturbation size $\epsilon$, number of training epochs $E$, number of words $n$, dimension of word embedding $d$
Output: Robust model $f_{\boldsymbol{\theta}}$
for $i = 1, 2, \cdots, |\mathcal{D}|$ do
    $\boldsymbol{\delta}_i \leftarrow \boldsymbol{U}(-\epsilon_0, \epsilon_0)/\sqrt{n \cdot d}$
end for
for $e = 0, 1, \cdots, E-1$ do
    for $\{(x_i, y_i)\} \subset \mathcal{D}$ do
        # Update adversarial perturbations
        $\boldsymbol{\delta}_i^0 \leftarrow \boldsymbol{U}(0, \epsilon_0) \odot \operatorname{sign}(\boldsymbol{\delta}_i)/\sqrt{n \cdot d}$
        $\boldsymbol{\delta}_i \leftarrow \boldsymbol{\delta}_i^0 + \epsilon \cdot \frac{\nabla_{\boldsymbol{\delta}_i^0} \mathcal{L}(f(\boldsymbol{v}(x_i) + \boldsymbol{\delta}_i^0), y_i)}{\left\| \nabla_{\boldsymbol{\delta}_i^0} \mathcal{L}(f(\boldsymbol{v}(x_i) + \boldsymbol{\delta}_i^0), y_i) \right\|_2}$
        Compute loss $\mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{v}(x_i) + \boldsymbol{\delta}_i), y_i)$
        Update model parameters $\boldsymbol{\theta}$
    end for
end for
return $f_{\boldsymbol{\theta}}$
```

4 Experiments

This section evaluates the robustness of the proposed FAT and typical defense baselines against various adversarial attacks on two typical NLP models across four datasets. Then the correlation between training cost and defense performance of PGD-AT and FAT is further investigated. We provide more analysis in Appendix. Code is available at `https://github.com/JHL-HUST/FAT`.

4.1 Experimental Setup

4.1.1 Datasets and Models

To thoroughly evaluate the effectiveness of the proposed method, we conduct experiments on three text classification datasets, including *IMDB* (Maas et al., 2011), *AGNEWS* (Zhang et al., 2015), and *DBPEDIA* (Zhang et al., 2015), and a natural language inference dataset of *QNLI* (Wang et al., 2019). The four standard benchmark datasets have various text lengths, number of classes, and sample sizes. Their specific information is shown in Table 1. We train the BERT model (Devlin et al., 2019) of the uncased base version and the RoBERTa (Liu et al., 2019) model of the base version on the four datasets.

4.1.2 Attack Methods

Since we focus on the defense without any additional predetermined linguistic knowledge and synonym information, we utilize four adversarial attacks, namely TextFooler (Jin et al., 2020), BERT-Attack (Li et al., 2020), TextBugger (Li

Dataset	#Training	#Testing	#Class	Avg. words
IMDB	25,000	25,000	2	268
AGNEWS	120,000	7,600	4	40
QNLI	105,000	5,400	2	11 / 31*
DBPEDIA	560,000	70,000	14	53

Table 1: Statistics of datasets. * denotes the average words of premise and hypothesis.

et al., 2019), and GBDA (Guo et al., 2021), involving character-level perturbations and word-level perturbations based on different synonym candidates. TextFooler defines synonyms based on the cosine distance between the word vectors, then identifies important words in the input text and performs synonym substitutions. BERT-Attack utilizes pre-trained masked language models to mine for synonym candidates. TextBugger mixes the character-level and word-level perturbations to attack the model. GBDA is a challenging white-box attack, which searches for a distribution of adversarial examples parameterized by a continuous valued matrix and utilizes gradient-based optimization to craft adversarial examples. For the first three attacks, we use the default implementation in TEXTATTACK[1]. For GBDA attack, we use the implementation of the paper. For the natural language inference dataset, each sample consists of two sentences: the premise and the hypothesis, and typically, only the hypothesis is perturbed to keep the true label unchanged. We randomly choose 800 test samples from each dataset to generate adversarial examples.

4.1.3 Defense Baselines

Following Liu et al. (2022), we compare our method with standard fine-tuning (Devlin et al., 2019) and four defense baselines, PGD-AT (Madry et al., 2018; Li et al., 2021), TAVAT (Li and Qiu, 2021), InfoBERT (Wang et al., 2021a), and Flooding-X (Liu et al., 2022). All the baselines and our proposed method require no predetermined linguistic knowledge, ensuring a fair evaluation. We also discuss our methods with the synonym-aware defenses in Appendix D.

4.1.4 Training Details

Our implementations are based on Liu et al. (2022) [2]. Since Liu et al. (2022) run baseline methods with 5 attack steps for 10 epochs, according to the analysis in Section 3.2, we run our proposed FAT for 30 epochs to achieve the same time consumption as PGD-AT, among which the last epoch is selected for evaluation. For hyper-parameters in Eq. (7) and Eq. (5), we set $\epsilon_0 = 0.05$ and $\epsilon = 0.2$. The detail hyper-parameter study is provided in Appendix E.

4.2 Main Results

We compare FAT with standard fine-tuning and typical defense baselines concerning the robustness against various attacks. The comparisonal results, using three evaluation metrics, on BERT and RoBERTa models are shown in Table 2 and Table 3, respectively. *Clean%* denotes the classification accuracy on the entire original test set. *Aua%* is short for the accuracy under attacks. *#Query* denotes the average number of queries to attack each sample. The more effective the defense method, the higher the metrics of *Aua%* and *#Query*. Meanwhile, we also need to ensure that *Clean%* does not decline much compared to the standard fine-tuning.

The results indicate that FAT has substantially enhanced the model's robustness, surpassing the defense baselines with a prominent margin on all four datasets and two models under various attacks. For instance, FAT outperforms the best defense baseline by 30.3%, 22.8%, and 12.7% with BERT model on the *IMDB* dataset under the three attacks, respectively. Especially on the large-scale *DBPEDIA* dataset, FAT exhibits 92.0%, 81.3%, and 92.9% accuracy under the three attacks, respectively. Besides, FAT achieves the same or even improved accuracy on the original test set compared to standard fine-tuning.

Under the same time limit, FAT using single-step gradient ascent to generate perturbation performs better than PGD-AT using multi-step gradient ascent, probably due to the following reasons. First, the difference between the perturbations generated by single-step and multi-step gradient ascent is trivial for AT on NLP models. Second, with much fewer calculations, FAT runs more epochs to achieve better robustness within a limited time. Specifically, Appendix C shows that even FAT trained with only 10 epochs can achieve competitive adversarial robustness. FAT significantly outperforms baselines if allowed to train for 30 epochs while maintaining the same overall training time as PGD-AT. Besides, Appendix D shows that the defense effect of FAT is consistently higher than that of ASCC within the same time limits.

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

[2] https://github.com/qinliu9/Flooding-X

Dataset	Defense	*Clean%*	TextFooler		BERT-Attack		TextBugger	
			Aua%	*#Query*	*Aua%*	*#Query*	*Aua%*	*#Query*
IMDB	Finetune*	95.0	24.5	1533.15	20.3	2237.38	48.7	1160.35
	PGD-AT*	95.0	26.3	1194.08	21.3	1465.83	52.3	982.02
	TAVAT*	95.5	27.6	1205.80	23.1	2244.77	54.1	1022.56
	InfoBERT *	96.3	27.4	1094.55	20.8	1428.67	49.8	1215.39
	Flooding-X*	**97.5**	40.5	2315.35	32.3	2248.71	62.3	**2987.95**
	FAT (w/o)	94.9	67.3	2550.85	49.8	3503.30	70.4	1650.51
	FAT	95.0	**70.8**	**2574.45**	**55.1**	**3636.75**	**75.0**	1687.15
AGNEWS	Finetune*	94.9	20.5	372.14	6.5	477.34	42.7	192.75
	PGD-AT*	94.8	37.2	428.13	32.8	704.78	58.2	252.87
	TAVAT*	**95.2**	39.7	441.11	23.7	672.52	55.9	234.01
	InfoBERT*	94.6	29.2	406.32	15.6	598.25	50.7	201.66
	Flooding-X*	94.9	42.4	451.35	27.4	690.27	62.2	222.49
	FAT (w/o)	**95.2**	60.8	500.62	**48.6**	**764.88**	**65.9**	**306.94**
	FAT	95.1	**62.3**	**505.86**	48.0	754.63	63.6	301.91
QNLI	Finetune*	90.6	5.3	161.88	3.5	216.46	10.9	98.39
	PGD-AT*	90.6	28.1	269.38	24.0	399.91	33.8	154.55
	TAVAT	91.6	32.3	243.71	16.3	302.17	30.6	140.97
	InfoBERT*	90.4	23.1	250.87	11.1	268.91	12.8	127.93
	Flooding-X*	**91.8**	27.9	251.17	26.2	364.06	29.5	137.12
	FAT (w/o)	91.4	44.8	271.69	28.1	384.83	39.4	172.35
	FAT	91.1	**48.3**	**280.07**	**33.0**	**414.37**	**44.3**	**184.29**
DBPEDIA	Finetune	99.3	19.0	444.05	28.4	607.01	53.3	312.20
	PGD-AT	**99.4**	66.5	645.82	51.8	912.57	79.9	411.62
	Flooding-X	99.3	28.6	548.02	36.5	687.55	69.5	339.22
	FAT (w/o)	99.3	90.8	716.78	77.1	1271.52	92.3	513.76
	FAT	**99.4**	**92.0**	**720.35**	**81.3**	**1277.14**	**92.9**	**515.76**

Table 2: The comparison results of FAT and baselines under various adversarial attacks on BERT model. FAT (w/o) denotes FAT using random perturbation initialization rather than our proposed initialization method. * indicates results reported in Liu et al. (2022). The best performance is highlighted in **bold**.

Dataset	Defense	*Clean%*	TextFooler		BERT-Attack		TextBugger	
			Aua%	*#Query*	*Aua%*	*#Query*	*Aua%*	*#Query*
IMDB	Finetune	95.5	14.5	1246.30	5.6	1366.54	25.5	881.95
	PGD-AT	95.6	49.4	1839.47	36.5	2414.39	46.8	1173.59
	TAVAT	95.6	66.8	2224.31	51.1	3422.95	71.1	1437.07
	InfoBERT	**95.7**	59.0	2182.16	47.5	2916.40	65.9	1340.95
	Flooding-X	95.5	44.8	1947.40	26.8	2381.76	58.0	1239.55
	FAT (w/o)	**95.7**	71.0	2347.62	52.4	3451.97	73.6	1578.73
	FAT	95.6	**74.4**	**2461.05**	**55.1**	**3512.07**	**76.8**	**1597.55**

Table 3: The comparison results of FAT and baselines under various adversarial attacks on RoBERTa model. FAT (w/o) denotes FAT using random perturbation initialization rather than our proposed initialization method. The best performance is highlighted in **bold**.

In Table 2 and Table 3, FAT (w/o) uses random perturbation initialization rather than the historical perturbation. The ablation comparison between FAT and FAT (w/o) reveals that the initialization utilizing previous information is crucial for crafting adversarial examples. For instance, when we initialize the perturbation along the direction of the previously generated perturbation, FAT performs better

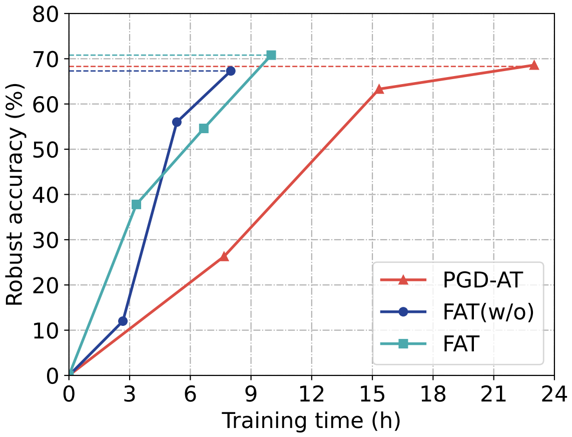


Figure 2: The training time (h) and robust accuracy ($\%$) of BERT models trained by different defense methods on the *IMDB* dataset. The points on the curve represent the models trained with 10, 20, and 30 epochs.

than FAT (w/o) under the TextFooler attack on the four datasets with BERT model, with the improvement of 3.5%, 1.5%, 3.5%, and 1.2% respectively. The same phenomenon holds for RoBERTa models, which verifies the generation of our method.

We further verify the defense effectiveness of FAT on the more challenging white-box attack, GBDA (Guo et al., 2021), and the sentence-level attack, MAYA (Chen et al., 2021), respectively. As detailed in Appendix A and B, FAT can also provide an essential defense against the white-box attack and sentence-level attack.

4.3 Training Efficiency

Since adversarial training is time-consuming for large-scale pre-trained models such as BERT, Liu et al. (2022) and Li et al. (2021) run PGD-AT for ten epochs in their experiments for a trade-off between training cost and model performance. However, we speculate that inadequate training significantly limits the performance of PGD-AT. It is observed that the clean accuracy on the original test data easily converges after several epochs of fine-tuning, whereas the robustness of the model requires more training epochs. Our proposed FAT method unleashes the robustness of the adversarial training for NLP models due to the efficient training. In this section, we explore the correlation between training time and the model's robustness of FAT and PGD-AT.

Specifically, we record the training time and evaluate the robustness of models trained with three defense methods for 10, 20, and 30 epochs on the *IMDB* dataset, respectively. We utilize the TextFooler attack to evaluate the robustness. The results are depicted in Figure 2.

Consistent with the analysis in Section 3.2.1, the time consumed for training 10 epochs of PGD-AT is the same as that of 30 epochs of FAT (w/o). Since FAT adds operations to record and exploit historical perturbations, the training efficiency is slightly lower than FAT (w/o) but still much faster than PGD-AT. In the same time limit, FAT and FAT (w/o) show significant superiority over PGD-AT.

After training all the three defense models for 30 epochs, PGD-AT slightly outperforms FAT (w/o). This could be attributed to the fact that PGD-AT uses more sophisticated adversarial perturbation for training. Note that at this point, PGD-AT has spent three times as much training time as FAT (w/o). In addition, combined with our proposed initialization using historical perturbations, FAT still slightly outperforms PGD-AT.

5 Conclusion

Continuous adversarial training (AT), which involves directly adding perturbations to the embedding representation during training, can enhance the robustness of NLP models in scenarios where synonyms are unknown. In this work, we proposed Fast Adversarial Training (FAT), a continuous AT method designed to boost adversarial robustness. FAT leverages insights into perturbation generation in the embedding space, employing single-step gradient ascent to generate adversarial perturbations. It also utilizes historical training information by initializing perturbations along the direction from the previous epoch. Extensive experiments demonstrate that FAT significantly outperforms existing defense baselines against various adversarial attacks with different perturbation granularities and model visibility. Notably, FAT achieves these results without relying on human-prescribed linguistic rules or access to attackers' synonyms, making it practical and easy to use in diverse real-world scenarios with varying synonym candidates and perturbation budgets.

In contrast to the image domain, continuous AT has been largely overlooked as a potent approach for textual adversarial defense. Our work demonstrates its efficacy as a synonym-unaware defense method. We encourage future research to consider continuous AT as a strong baseline in relevant studies. Furthermore, since continuous AT helps bridge the gap between image and text AT, we plan to explore and adapt various advanced AT methods from the image domain for application in the text domain in future work.

Limitations

Since word-level attacks based on synonym substitutions and character-level attacks are the most commonly used methods and generally ensure semantic consistency, this paper focuses on enhancing the model's robustness against these types of attacks. However, we do not explore robustness against sentence-level attacks. Additionally, our method can also be applied to large language models to improve adversarial robustness. Due to the huge cost of adversarial training, we have not yet conducted relevant experiments. We will continue to investigate the potential of FAT in our future work.

Acknowledgments

This work is supported by National Natural Science Foundation (U22B2017, 62076105) and International Cooperation Foundation of Hubei Province, China (2024EHA032).

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani B. 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.

Yangyi Chen, Jin Su, and Wei Wei. 2021. Multi-granularity textual adversarial attack with behavior cloning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4511–4526.

Yue Deng, Wenxuan Zhang, Sinno Jialin Pan, and Lidong Bing. 2024. Multilingual jailbreak challenges in large language models. In *The Twelfth International Conference on Learning Representations, ICLR*.

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*, 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*.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. Hotflip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 31–36.

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*, pages 50–56.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations*.

Chuan Guo, Alexandre Sablayrolles, Hervé Jégou, and Douwe Kiela. 2021. Gradient-based adversarial attacks against text transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5747–5757.

Xingang Guo, Fangxu Yu, Huan Zhang, Lianhui Qin, and Bin Hu. 2024. Cold-attack: Jailbreaking llms with stealthiness and controllability. In *Forty-first International Conference on Machine Learning, ICML 2024*.

Maor Ivgi and Jonathan Berant. 2021. Achieving model robustness through discrete adversarial training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1529–1544.

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*, pages 4129–4142.

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 2020*, pages 8018–8025.

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*.

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*, 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*, 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.

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*, pages 5634–5644.

Xiaodong Liu, Hao Cheng, Pengcheng He, Weizhu Chen, Yu Wang, Hoifung Poon, and Jianfeng Gao. 2020. Adversarial training for large neural language models. *CoRR*, abs/2004.08994.

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.

Minxuan Lv, Chengwei Dai, Kun Li, Wei Zhou, and Songlin Hu. 2023. CT-GAT: cross-task generative adversarial attack based on transferability. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, pages 5581–5591.

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 *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, pages 142–150.

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*.

Takeru Miyato, Andrew M. Dai, and Ian J. Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. In *5th International Conference on Learning Representations*.

Maximilian Mozes, Pontus Stenetorp, Bennett Kleinberg, and Lewis D. Griffin. 2021. Frequency-guided word substitutions for detecting textual adversarial examples. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 171–186.

Nicolas Papernot, Patrick D. McDaniel, Ananthram Swami, and Richard E. Harang. 2016. Crafting adversarial input sequences for recurrent neural networks. In *2016 IEEE Military Communications Conference*, pages 49–54.

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 Conference of the Association for Computational Linguistics*, pages 1085–1097.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations*.

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*.

Boxin Wang, Hengzhi Pei, Boyuan Pan, Qian Chen, Shuohang Wang, and Bo Li. 2020. T3: Tree-autoencoder constrained adversarial text generation for targeted attack. In *Conference on Empirical Methods in Natural Language Processing*, pages 6134–6150.

Boxin Wang, Shuohang Wang, Yu Cheng, Zhe Gan, Ruoxi Jia, Bo Li, and Jingjing Liu. 2021a. Infobert: Improving robustness of language models from an information theoretic perspective. In *9th International Conference on Learning Representations*.

Xiaosen Wang, Jin Hao, Yichen Yang, and Kun He. 2021b. Natural language adversarial defense through synonym encoding. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, pages 823–833.

Xiaosen Wang, Yichen Yang, Yihe Deng, and Kun He. 2021c. Adversarial training with fast gradient projection method against synonym substitution based text attacks. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*, pages 13997–14005.

Yibin Wang, Yichen Yang, Di He, and Kun He. 2023. Robustness-aware word embedding improves certified robustness to adversarial word substitutions. In *Findings of Association for Computational Linguistics*.

Yichen Yang, Xiaosen Wang, and Kun He. 2022. Robust textual embedding against word-level adversarial attacks. In *Uncertainty in Artificial Intelligence, Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, pages 2214–2224.

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.

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.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, pages 649–657.

Haiteng Zhao, Chang Ma, Xinshuai Dong, Anh Tuan Luu, Zhi-Hong Deng, and Hanwang Zhang. 2022. Certified robustness against natural language attacks by causal intervention. In *International Conference on Machine Learning*, pages 26958–26970.

Yuqi Zhou, Lin Lu, Ryan Sun, Pan Zhou, and Lichao Sun. 2024. Virtual context enhancing jailbreak attacks with special token injection. In *Findings of the Association for Computational Linguistics: EMNLP*, pages 11843–11857.

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*.

A Evaluation with White-box Attack

To further verify the defense effectiveness, we compare FAT with standard fine-tuning and the best baseline method Flooding-X on the GBDA attack (Guo et al., 2021), which is a more challenging white-box attack method.

As shown in Table 4, the GBDA attack severely degrades the accuracy of fine-tuned models. FAT has enhanced the model's robustness by a clear margin of 46.4%, 26.1%, and 26.0% on the three datasets, respectively, indicating that not only in the black-box scenario but also in the white-box scenario, FAT can provide an essential defense against adversarial examples.

B Evaluation with Sentence-level Attack

MAYA (Chen et al., 2021) is a black-box attack that mixes multi-granularity textual perturbations, including sentence-level rephrasing and word substitutions. We randomly choose 600 test samples for the MAYA attack, as it takes a longer time to generate adversarial examples. As shown in Table 5, the accuracy on FAT is only 0.5% less than Flooding-X under the MAYA attack, but around 180 more model queries are required for attacking each sample on average, indicating that FAT is more difficult to attack.

C Impact of Training Epoch

In Table 2 and Table 3, we follow the work of Liu et al. (2022) to run baseline methods for 10 epochs. According to the analysis in Section 3.2, we run our FAT for 30 epochs to achieve the same time consumption as PGD-AT does.

Actually, as shown in Table 6, even FAT trained with only 10 epochs can achieve competitive adversarial robustness. FAT significantly outperforms the baselines if allowed to train for 30 epochs while keeping the same overall training time as the PGD-AT does.

D Comparison with Synonym-aware AT

Experiments by Li et al. (2021) have shown that PGD-AT is already superior to many competitive synonym-aware methods on the BERT model. We further take ASCC (Dong et al., 2021), an adversarial training method relying on synonyms, as an example to compare FAT with synonym-aware AT on the RoBERTa model. We use the same synonyms and hyper-parameters of the original paper to re-implement ASCC on RoBERTa model. As shown in Table 7, the robustness of FAT is also better than ASCC on the RoBERTa model.

We further evaluate the training efficiency between FAT and ASCC. Table 8 shows the training time (h) and robust accuracy (%) of RoBERTa models trained by ASCC and our FAT methods on the *IMDB* dataset. The defense effect of FAT is consistently higher than that of ASCC within the same training time limits.

Defense	*IMDB*	*AGNEWS*	*QNLI*
Finetune	0.4	0.4	16.3
Flooding-X	40.0	16.3	**47.8**
FAT	**46.8**	**26.5**	42.3

Table 4: The accuracy (*Aua%*) of FAT and Flooding-X with BERT models under the GBDA attack.

Defense	*Aua%*	*#Query*
Finetune	4.3	514.29
Flooding-X	**9.3**	498.01
FAT	8.8	**681.82**

Table 5: The defense performance of FAT and Flooding-X with BERT models under the MAYA attack on *IMDB* dataset.

Defense	*IMDB*	*AGNEWS*	*QNLI*
PGD-AT	26.3	37.2	28.1
Flooding-X	40.5	42.4	27.9
FAT (10 epochs)	37.8	37.1	29.1
FAT (30 epochs)	**70.8**	**62.3**	**48.3**

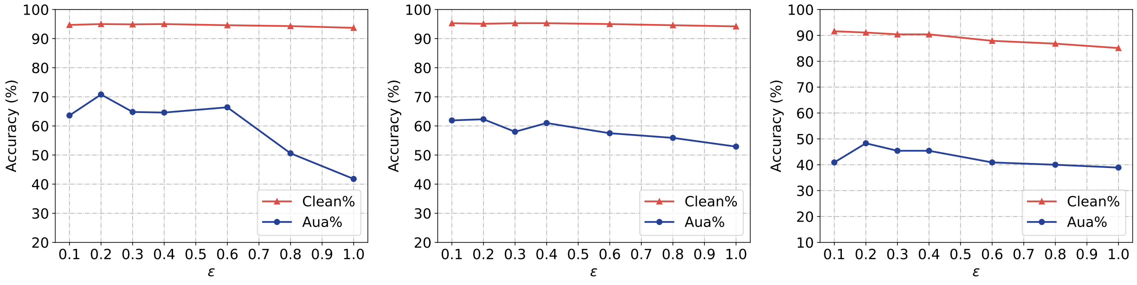
Table 6: The accuracy (*Aua%*) of FAT and Flooding-X with BERT models under the TextFooler attack.

E Hyper-parameter Study

This subsection evaluates the impact of hyper-parameters on the performance of FAT. We focus on two metrics, the accuracy on the original test set, denoted by *Clean%*, and the robust accuracy under the TextFooler attack, denoted by *Aua%*. The hyper-parameter ϵ in Eq. (5) controls the perturbation magnitude. To study the effect of ϵ on FAT, we train the model with $\epsilon = 0.1, 0.2, 0.3, 0.4, 0.6, 0.8, 1.0$, respectively. Figure 3 indicates that FAT is not sensitive to the hyper-parameter ϵ, especially when ϵ is between 0.1 and 0.4. With the extensive range of $\epsilon = 0.1$ to 1.0, even the worst-case robust

Defense	*Clean%*	TextFooler		BERT-Attack		TextBugger	
		Aua%	*#Query*	*Aua%*	*#Query*	*Aua%*	*#Query*
ASCC	95.4	51.9	1860.10	48.1	2875.21	53.3	1221.58
FAT	95.6	**74.4**	**2461.05**	**55.1**	**3512.07**	**76.8**	**1597.55**

Table 7: The comparison results of FAT and ASCC under various adversarial attacks on RoBERTa model.



(a) *IMDB*

(b) *AGNEWS*

(c) *QNLI*

Figure 3: The impact of hyper-parameter ϵ on the performance of FAT across the three datasets with the BERT model.

Training time	≈ 5h	≈ 7h	≈ 10h
ASCC	2.3	18.3	51.9
FAT	**30.9**	**64.9**	**74.4**

Table 8: The accuracy (*Aua%*) of FAT and ASCC with BERT models under the same training time (*h*) on the *IMDB* dataset.

accuracy is higher than all defense baselines.

When ϵ is set between 0.1 and 0.4, the clean accuracy remains almost constant on the three datasets. When $\epsilon = 0.3$ or 0.4, the clean accuracy of the model is still more than 90.0% on the *QNLI* dataset, which is not significantly weakened compared with 90.6% of standard training. The robust accuracy fluctuates within a small range, reaching a maximum when $\epsilon = 0.2$. With the increase of ϵ, the clean accuracy decreases on the three datasets, especially on the *QNLI* dataset, and the robust accuracy also decreases significantly. In summary, when ϵ is between 0.1 and 0.4, FAT achieves a proper trade-off between clean accuracy and robustness.