READ: Improving Relation Extraction from an ADversarial Perspective

Anonymous ACL submission

Abstract

Recent works in relation extraction (RE) have achieved promising benchmark accuracy; however, our adversarial attack experiments show that these works excessively rely on entities, making their generalization capability questionable. To address this issue, we propose an adversarial training method specifically designed for RE. Our approach introduces both sequence- and token-level perturbations to the sample and uses a separate perturbation vocabulary to improve the search for entity and context perturbations. Furthermore, we introduce a probabilistic strategy for leaving clean tokens in the context during adversarial training. This strategy enables a larger attack budget for entities and coaxes the model to leverage relational patterns embedded in the context. Extensive experiments show that compared to various adversarial training methods, our method significantly improves both the accuracy and robustness of the model. Additionally, experiments on different data availability settings highlight the effectiveness of our method in low-resource scenarios. We also perform in-depth analyses of our proposed method and provide further hints. We will open-source all codes in our work to facilitate future research.

1 Introduction

Relation extraction (RE) is an important subtask of information extraction and plays a crucial role in many other natural language processing (NLP) tasks like knowledge base construction (Luan et al., 2018) and question answering (Sun et al., 2021). The goal of RE is to determine the relationship between a head entity and a tail entity. For example, given the sentence "*Miettinen hired for WPS champ Sky Blue.*", the RE models are supposed to predict the relation "*Employee-Of*" between the head entity "*Miettinen*" and the tail entity "*Sky Blue*". With the recent advances in pre-trained language model (Kenton and Toutanova, 2019; Liu et al., 2019) and self-supervised learning (Qin et al., 2021; Hogan et al., 2022) techniques, RE models have achieved promising benchmark accuracy, reaching levels comparable to human performance.

	Sentence	Prediction
Org	Miettinen hired for WPS champ Sky Blue.	Employee-Of ✓
Adv	Miettinen hired for WPS champ Jeez Blue.	No-Relation ✗

Table 1: An example from SemEval. We use green color to represent the head entity and orange color to represent the tail entity. Underlining is used for word substitution.

The recent success of RE models sparks a growing interest in conducting more detailed analyses (Han et al., 2020c; Peng et al., 2020; Zhang et al., 2023). A significant issue that arises in this context is to explore *whether the RE model learns from context or entities for relation prediction*. Analyzing this problem could reveal the underlying nature of RE models and offer informative insights for their improvement. To address this issue, various methods are proposed such as information masking (Peng et al., 2020) and counterfactual analysis (Wang et al., 2022). One drawback of these methods is they usually involve removing entities or context in the sample and observing the model's performance with the remaining part. That enables them to draw the conclusion about how much can the model learn from entity/ context when giving each of them individually. However, whether the model would prefer to learn from context or entities when both of them are given still remains unclear. We name this problem *learning preference* in RE.

To address this issue, we propose a novel approach **READ**, a.k.a. improving **R**elation **E**xtraction from an **AD**versarial perspective. We begin by introducing the utilization of **adversarial attacks** (Jin et al., 2020; Garg and Ramakrishnan, 2020) as a means to investigate the model's

learning preference and robustness. Adversarial attacks in NLP are designed to deceive the model by making very few text substitutions. As the example shown in Table 1, by replacing the original word "*Sky*" with another word "*Jeez*", the attack method successfully fools the model into assigning an incorrect label "*No-Relation*" to this sample. Adversarial attacks provide a highly insightful perspective for determining the crucial parts of the sample from the model's viewpoint. In this particular example, we can conclude that the word "*Sky*", as a part of the entity name, is crucial for the model to make accurate predictions.

In our preliminary experiment applying adversarial attacks to RE, we discovered a clear over-dependency on entities within the current RE model. This is consistent with the previous works (Peng et al., 2020) that RE models tend to utilize shallow cues from entities to make predictions. Our analysis revealed that this over-dependency is the underlying cause of the models' vulnerability to adversarial attacks and can also lead to poor generalization in clean samples. So the key to improving current RE models is to mitigate this over-dependency on entities.

One straightforward approach to bolster models' robustness is text substitution. However, the considerable time cost to generate adversarial samples with the text substitution method constrains it in scaling in large RE datasets (Yoo and Qi, 2021). Also, in our preliminary experiments, we observed a performance drop in the clean test set with text substitution, which has also been reported by previous works (Xu et al., 2022b)[1]. So we shift our focus towards **virtual adversarial training** (Miyato et al., 2016; Madry et al., 2018), which applies continuous perturbations at the embedding level during training, rendering it a more refined and efficient approach. Our method builds upon the advancements of the current adversarial training methods in NLP (Zhu et al., 2019; Li and Qiu, 2021) and introduces both sequence- and token-level perturbations to the RE sample. To facilitate perturbation searching, we devise a separate perturbation vocabulary that tracks the accumulated perturbation for entity and context respectively. Furthermore, we propose a novel probabilistic strategy to encourage the model to leverage relation patterns from the unperturbed context. Through extensive experiments, we demonstrate the effectiveness of our method on both adversarial and clean test samples. We also observe significant improvements in low-resource settings, indicating the great potential of our method in scenarios with limited data. We conduct a series of in-depth analyses to give more hints about READ.

The contribution of our work could be summarized as follows:

- We propose READ, a novel adversarial method to improve current RE models' robustness.
- READ adopts adversarial attacks to analyze RE models' learning preferences and expose an obvious over-dependency on entities.
- To enhance RE models' generalization, READ utilizes a virtual adversarial training explicit design for RE. Experiments on three mainstream datasets demonstrate the effectiveness of READ.

2 Related Work

2.1 Relation Extraction

Early RE methods employ pattern-based algorithms (Mooney, 1999) or statistical methods (Mintz et al., 2009; Riedel et al., 2010; Quirk and Poon, 2017) to handle relation extraction. Neural-based RE models (Zhang and Wang, 2015; Peng et al., 2017; Miwa and Bansal, 2016) emerge with the advancements in deep learning and natural language processing. Among them, the transformer-based RE models (Shi and Lin, 2019) achieve state-of-the-art performance. To further enhance performance, various self-supervised learning mechanism designs for RE have been proposed (Soares et al., 2019; Qin et al., 2021; Hogan et al., 2022).

There are some works that explore applying adversarial training in RE. Qin et al. (2018) proposes a generative adversarial training framework to address the noisy labeling problem in distantly supervised relation extraction. Hao et al. (2021) adopt adversarial training to address the false negatives problem in relation extraction. Both Zhang et al. (2020) and Li et al. (2023b) design new adversarial training pipelines to generate augmented samples for RE. In our work, we propose to analyze and improve RE models from an adversarial perspective to expose and reduce the excessive reliance of the models on entities.

[1]We put the experiment result and analysis of text substitution in Appendix A

2.2 Adversarial Attack & Training

Text substitution is one of the most commonly used methods in NLP to attack models or generate adversarial samples (Iyyer et al., 2018; Ebrahimi et al., 2018). It replaces the original word with its synonym based on certain criteria like word embedding similarity (Zang et al., 2020; Ren et al., 2019; Jin et al., 2020) or model infilling (Garg and Ramakrishnan, 2020; Li et al., 2020). There are also some works that propose character-level (Gao et al., 2018; Li et al., 2018) and phrase-level (Lei et al., 2022) substitutions to generate various adversarial samples. However, those substitution methods are often challenged by the massive space of combinations when searching for the target word to replace, making them time-costly to implement (Yoo and Qi, 2021).

Virtual adversarial training (VAT) methods generate adversarial samples by applying perturbations to the embedding space (Miyato et al., 2018). This helps VAT become more efficient than traditional text substitution methods. VAT makes the model more robust under adversarial attacks while also improving the model's performance in clean test samples (Miyato et al., 2016; Cheng et al., 2019). To make VAT more effective, Zhu et al. (2019) accumulate perturbation in multiple searching steps to craft adversarial examples. Li and Qiu (2021) devise a Token-Aware VAT (TA-VAT) method to allocate more attack budget to the important tokens in the sequence. While there are some works that apply virtual adversarial training methods to RE for different purposes, we propose an Entity-Aware VAT method explicitly designed for RE to mitigate over-dependency and non-generalization on entities. We give a more detailed discussion about adversarial attacks and training in NLP in Appendix B.

3 Adversarial Attack for RE

In this section, we start by analyzing the state-of-the-art (SOTA) RE models' performance under textual adversarial attacks. Then, through further analysis, we expose the over-dependency and non-generalization on entities in the current RE models.

3.1 Attack Settings

We apply adversarial attacks on ERICA (Qin et al., 2021) and FineCL (Hogan et al., 2022), the two SOTA models with RE-specific self-supervised training. We choose three RE datasets to conduct experiments: SemEval-2010 Task 8 (Hendrickx et al., 2019), ReTACRED (Stoica et al., 2021) and Wiki80 (Han et al., 2019). For each dataset, we randomly choose 1,000 test samples to conduct experiments on. We use different attack methods including BAE (Garg and Ramakrishnan, 2020), TextFooler (Jin et al., 2020), TextBugger (Li et al., 2018) and Projected Gradient Descent (PGD) Attack (Madry et al., 2018). Here, PGD Attack is a white-box attack that utilizes the model's gradient, while the remaining three attacks are black-box attacks. We use Textattack[2] package and follow all the hyper-parameter settings in the original papers.

To evaluate how RE models perform under adversarial attacks, we follow the previous works (Li et al., 2021; Xu et al., 2022a) and report clean accuracy (the model accuracy on clean examples), accuracy under attack (the model accuracy on adversarial examples subjected to a specific attack), and the number of queries (the average number of queries the attacker required to perform successful attacks). The experiment results are shown in Table 2.

To access RE models' learning preferences, we analyze whether tokens in entities would be attacked more than them in context. If so, that means entities are more important than context in the model's perspective. For each dataset, We calculate how frequently the adversarial attacks involve the entity (Entity Freq) and the proportion of the perturbed entity in all perturbed tokens (Entity Ratios). We also report the average proportion of the entity length in the sample for comparison (Entity %). The experiment results are shown in Table 3.

3.2 Result Analysis

Here we analyze the attacking results of TextFooler on FineCL and put the remaining results with other attack methods and models into Section 5.3 and Appendix C. As shown in Table 2, FineCL suffers from a dramatic performance drop up to 91.2% in the Wiki80 dataset. In the other two datasets, there is also an obvious performance drop compared with using the clean test set, offering evidence that **current RE models are not very robust under adversarial attacks**.

As for the model's learning preference, from Table 3 we find Entity Freq is quite high in the three datasets, suggesting entities are frequently targeted for attacks. Also, Entity Ratio is much

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

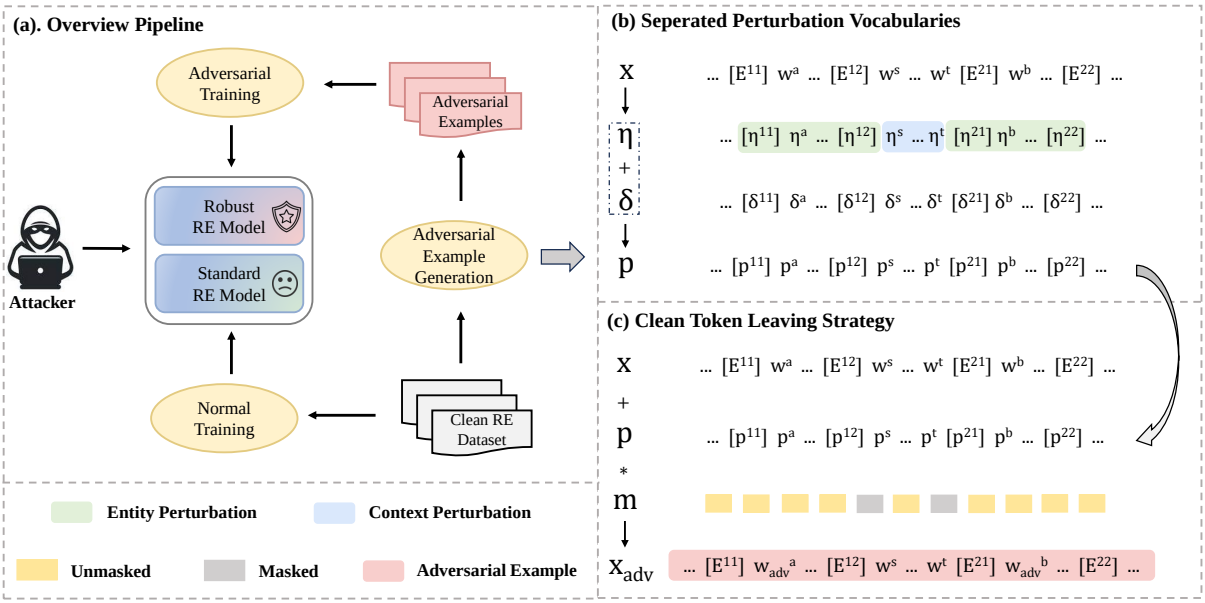


Figure 1: (a) Overview pipeline of our method which adopts adversarial methods to analyze and improve RE models. (b) Separated perturbation vocabularies (Section 4.2). (c) Clean token leaving strategy (Section 4.3). We use "$[E_{11}]/[E_{12}]$" and "$[E_{21}]/[E_{22}]$" to mark the head and tail entity respectively.

Dataset	Clean	AUA	Query
SemEval	92.7	18.1 (-80.5%)	73.83
ReTACRED	90.1	27.6 (-69.4%)	227.07
Wiki80	96.1	8.5 (-91.2%)	111.28

Table 2: TextFooler attack results on three RE datasets.

	Entity Freq	Entity Ratio	Entity %
SemEval	77.1	38.0	12.0
ReTACRED	52.6	12.7	9.2
Wiki80	90.7	36.4	17.4

Table 3: Analysis of the model's learning preference. We report how frequently the entity is attacked (Entity Freq), the proportion of the perturbed entity in all perturbed tokens (Entity Ratios), and the average proportion of the entity length in the sample (Entity %).

	Entity AS	Context AS
SemEval	68.5	62.3
ReTACRED	44.2	33.9
Wiki80	84.2	75.5

Table 4: Attack success (AS) rate of entity and context. The AS for entity and context is calculated by dividing the total number of successfully attacked entities/contexts by the total number of attacked entities/contexts.

higher than Entity %, indicating that entities are more often considered important words according to the model's perspective. Based on these two findings we deduce that **Current RE models rely more on entities to make predictions.**

The aforementioned conclusion makes us wonder about the RE models' robustness and generalization toward entities. To evaluate it, we calculate the attack success (AS) rate of entity and context respectively. As Table 4 shows, we find the AS of entity is significantly higher than that of context, which means entities are more vulnerable to attacks. This provides evidence that **over-dependency on entities has led to a non-generalization within the model.**

4 Adversarial Training for RE

To improve the robustness and generalization of the RE models, READ employs an **Entity-Aware Virtual Adversarial Learning** method. In this section, we first give a brief illustration of the virtual adversarial training (VAT) process, then we will introduce our Entity-Aware VAT method in detail.

4.1 Virtual Adversarial Training

In virtual adversarial learning, we first need to find a small perturbation $\boldsymbol{\delta}$ that maximizes the misclassification risk of the model. Then, with the perturbation added to the original inputs $\boldsymbol{X}$, the goal of virtual adversarial learning is to optimize the model parameter θ to minimize the loss of those

adversarial samples. That Min-Max process can be summarized as follows:

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

where $\boldsymbol{X}$ is the embedding of the input sequence and y is the ground truth label. ϵ is the norm ball used to restrict the magnitude of $\boldsymbol{\delta}$.

Commonly, gradient ascent is used to do the perturbation search iteratively since the inner maximize function is non-concave. At step t:

$$\boldsymbol{\delta}_{t+1} = \prod_{||\boldsymbol{\delta}_t||_F < \epsilon} \frac{\boldsymbol{\delta}_t + \alpha g(\boldsymbol{\delta}_t)}{||g(\boldsymbol{\delta}_t)||_F} \quad (2)$$

$$g(\boldsymbol{\delta}_t) = \nabla_{\boldsymbol{\delta}} L(f_\theta(\mathbf{X}+\boldsymbol{\delta}_t), y) \quad (3)$$

where $\prod$ means the process of projecting the perturbation onto the norm ball. In the PGD algorithm, Frobenius norm F is used to constraint $\boldsymbol{\delta}$.

4.2 Separate Perturbation Vocabularies

Unlike images in the computer vision field where every pixel only carries limited information across instances, tokens in natural language processing are relatively independent semantic units and different tokens can vary in their importance for the sequence. Previous work (Li and Qiu, 2021) proposes a Token-Aware VAT method based on this thought and designs a global perturbation vocabulary to record each token's perturbation.

In our work, we borrow this insight and improve it for RE by using separate perturbation vocabularies. Intuitively, entity and context play quite different roles in the relation extraction process for models (Peng et al., 2020). Entities are the main components for the model to focus on while context can provide auxiliary information. To address this in adversarial training of RE, we keep two perturbation vocabularies for entities and context separately.

To be specific, we create the entity perturbation vocabulary $\boldsymbol{V}_e \in \mathbb{R}^{N \times D}$ and context perturbation vocabulary $\boldsymbol{V}_c \in \mathbb{R}^{N \times D}$ at the beginning of the adversarial training. Here N is the vocabulary size and D is the hidden size of the model's embedding. In each mini-batch, the i_{th} token in the sequence will be assigned an initialized perturbation from the corresponding vocabulary as the token-level perturbation $\boldsymbol{\eta}_0^i$:

$$\boldsymbol{\eta}_0^i = \begin{cases} \boldsymbol{V}_e[w_i], & w_i \in Entity, \\ \boldsymbol{V}_c[w_i], & w_i \in Context. \end{cases} \quad (4)$$

Then we follow Li and Qiu (2021) exactly to update the token-level perturbation. After the perturbation optimization, the two vocabularies are updated respectively with the token perturbation belonging to their category.

4.3 Probabilistic Clean Token Leaving

To address the importance of entities in adversarial training, we also adopt a probabilistic clean token leaving strategy for context. In each mini-batch, we randomly choose $n\%$ of tokens W_c in context and mask both their token- and sentence-level perturbation in every perturbation optimization step t:

$$W_c = RandomlySelect(Context, n) \quad (5)$$

$$\boldsymbol{X}_{adv}^i = \begin{cases} \boldsymbol{X}^i, & w_i \in W_c, \\ \boldsymbol{X}^i + \boldsymbol{\delta}_t + \boldsymbol{\eta}_t^i, & Otherwise \end{cases} \quad (6)$$

There are two benefits of using our probabilistic clean token leaving strategy. Firstly, the attack budget ϵ is constant for each sentence, which means reducing context perturbation is equivalent to increasing the attack budget for the entity. So it serves as an additional attack to further improve the model's robustness and generalization on entities. This is our main objective given the model's non-generalization and over-dependency on entities. Also, according to the previous works (Zhang et al., 2021; Mekala et al., 2022), deep neural networks are more willing to learn from clean components with less noise. So the strategy also gives the model more chances to leverage relational patterns present in the context (Peng et al., 2020) by learning from those clean tokens. We give a detailed process of our Entity-Aware VAT method in Figure 1.

5 Experiment

In this section, we design experiments to test our Entity-Aware VAT's performance on both clean and adversarial samples.

5.1 Setup

To evaluate our method's performance, we report performance on three RE datasets, SemEval-2010 Task 8 (Hendrickx et al., 2019), ReTACRED (Stoica et al., 2021) and Wiki80 (Han et al., 2019). We follow the previous work and use 1%, 10%

Dataset	Method	Clean	PGD		TextBugger		BEA		TextFooler	
			AUA↑	Query↑	AUA↑	Query↑	AUA↑	Query↑	AUA↑	Query↑
SemEval	Normal-Train	92.7	42.2	6.55	39.2	39.03	30.5	75.27	15.9	73.83
	FreeLB	**93.3**	45.4	6.80	41.5	39.41	31.6	75.79	15.6	73.35
	TA-VAT	93.1	45.2	6.75	41.6	39.22	31.6	**78.93**	16.5	71.97
	Ours	93.1	**51.5**	**7.0**	**42.6**	**41.18**	**32.5**	76.7	**18.8**	**74.77**
ReTACRED	Normal-Train	90.1	56.4	7.52	31.7	89.25	41.4	126.27	27.6	227.07
	FreeLB	90.0	64.2	7.87	29.8	85.83	40.1	127.16	28.6	228.54
	TA-VAT	**91.3**	68.6	8.11	28.9	83.38	41.8	128.10	30.0	230.88
	Ours	**91.3**	**76.2**	**8.43**	**34.0**	**89.30**	**49.6**	**140.98**	**38.9**	**252.63**
Wiki80	Normal-Train	96.1	58.7	8.34	26.3	52.93	37.8	46.32	8.5	111.28
	FreeLB	95.9	65.3	8.57	27.2	53.13	39.0	49.1	9.0	111.18
	TA-VAT	96.5	74.0	8.82	**29.2**	**54.56**	39.3	**49.55**	8.3	107.21
	Ours	**96.7**	**76.3**	**8.99**	28.8	53.40	**40.0**	48.64	**10.7**	**112.08**

Table 5: Experiment results on the three datasets under adversarial attacks. The best results in each dataset are in bold. For each experiment, we run three times and the average scores are reported.

and 100% data in the training set to train the model respectively. For the baseline RE model, we choose BERT (Kenton and Toutanova, 2019), RoBERTa (Liu et al., 2019), ERICA (Qin et al., 2021) and FineCL (Hogan et al., 2022). We choose the two best baseline models, FineCL and ERICA, to apply the adversarial learning methods. Here we report FineCL's result and put the results of ERICA in Appendix E. We compare our proposed method with FreeLB(Zhu et al., 2019) and TA-VAT(Li and Qiu, 2021). They are widely used virtual adversarial learning methods against textual attacks. For standard accuracy metrics, we follow the previous works and report the F1 score for SemEval and ReTACRED, and the accuracy score for Wiki80. We also test our method in the document-level RE scenario and put the result in Appeneix F.

We also test our proposed method's performance under adversarial attacks. All the adversarial attack methods and robustness metrics we use are mentioned in Section 3.1

5.2 Implementation Details

We build our method based on PyTorch-1.8.1[3] deep learning framework and Transformers-2.5.0[4] library. We follow the hyper-parameter settings in the original paper to reproduce each baseline's result. To improve the experiments' reliability, we report the average results of the top three adversarial hyper-parameter configurations based on their scores in the development set. Refer to Appendix G for more detailed settings of our experiments.

[3]https://pytorch.org/

[4]https://huggingface.co/docs/transformers/index

5.3 Results on Adversarial Samples

We employed FineCL as the baseline and assessed the performance of each adversarial method against different attacks. To provide a baseline comparison, we designated the standard model without any adversarial training as "Normal-Train", which is included in the first row of Table 5. From the scores reported, we can observe some readily apparent trends: (1). Our method consistently outperforms other adversarial training methods under various attack methods on the three datasets. (2) For the ReTACRED dataset, both FreeLB and TA-VAT exhibit a decrease in performance under the TextBugger attack. In contrast, our method demonstrates robust improvements in both accuracy and query number, showing the resilience of our proposed approach. (3) TextFooler achieves the best attack success rate (AS) result on all three datasets, indicating that current RE models are particularly sensitive to the synonym replacement attack employed by TextFooler.

5.4 Results on Clean Samples

Table 6 presents the results evaluated using the clean samples of each dataset. It is evident that the utilization of adversarial training methods yields a significant improvement in the performance of the best baseline model (FineCL). Among the three employed adversarial training methods, our Entity-Aware VAT method stands out by reaching the best score across almost every dataset and availability setting. That indicates our improved adversarial training method also benefits the RE model in clean test samples.

Moreover, we have observed that adversarial learning exhibits a more pronounced impact in low-resource settings. For example, the improvement

Dataset	SemEval			ReTACRED			Wiki80		
Size	1%	10%	100%	1%	10%	100%	1%	10%	100%
BERT	40.8	78.7	86.4	52.4	73.3	83.2	57.1	81.0	90.7
Roberta	50.0	81.6	85.8	58.2	82.5	88.7	60.7	85.4	91.3
ERICA	50.2	82.0	88.5	64.1	83.4	87.8	71.3	86.8	91.6
FineCL	50.8	82.7	88.6	62.8	83.2	87.1	72.7	86.9	91.6
FineCL + FreeLB	52.0	83.2	88.8	63.1	84.0	88.4	72.6	87.1	91.8
FineCL + TA-VAT	52.5	83.1	89.0	64.1	84.3	88.5	73.0	**87.5**	91.8
FineCL + Ours	**53.2**$_{+4.7\%}$	**83.3**$_{+0.7\%}$	**89.2**$_{+0.7\%}$	**64.4**$_{+2.5\%}$	**85.0**$_{+2.2\%}$	**88.7**$_{+1.8\%}$	**73.3**$_{+0.8\%}$	87.3$_{+0.5\%}$	**92.0**$_{+0.4\%}$

Table 6: Experiment results on clean samples of each dataset. We follow the previous works (Hogan et al., 2022; Qin et al., 2021) and report the F1 score for SemEval and ReTACRED, and the accuracy score for Wiki80. We also add the quantitative comparison results between our method and the FineCL baseline. For each experiment, we run three times and report the average score.

brought by our Entity-Aware VAT method on three datasets with 100% training data is 0.7%, 0.8% and 0.4%. However, it achieves a remarkable 4.7% of performance improvement on SemEval with 1% of training data. This notable improvement highlights the immense potential of adversarial training methods for RE in scenarios with limited resources.

6 Further Analysis

In this section, we conduct further experiments to give in-depth analyses of the mechanism of our proposed method.

	1%	10%	100%		
Metrics	F1	F1	F1	AUA	Query
TA-VAT	52.5	83.1	89.0	16.5	71.97
Ours w/o SPV	52.8	83.2	89.1	**18.8**	73.63
Ours w/o CTL	53.1	83.0	**89.2**	16.3	72.51
Ours	**53.2**	**83.3**	**89.2**	**18.8**	**74.77**

Table 7: Ablation study on separate perturbation vocabulary (SPV) and clean token leaving (CTL) strategy using SemEval. The attacker used in 100% training data availability is TextFooler. We include TA-VAT since it is identical to our method when both SPV and CTL are removed.

6.1 Ablation Study

The separate perturbation vocabulary (SPV) and clean token leaving (CTL) strategy are the two main methods we propose for adversarial training in RE. In this section, we conduct an ablation study on them to figure out each method's effectiveness in improving the robustness and accuracy of the model. We conduct experiments on SemEval with 1%, 10% and 100% training data availability. We report F1 in all three availability settings and AUA and Query in 100% training data availability.

Table 7 shows the result of our ablation study. We also report the model's performance with TA-VAT because our method degrades to be TA-VAT without the two methods we propose. We find both separate perturbation vocabulary and clean token leaving are effective in improving the model's accuracy in clean samples. And clean token leaving brings a significant improvement in robustness to the model while the model with separate perturbation vocabulary only does not. That indicates the improvement in robustness of our method is mainly from clean token leaving in the context.

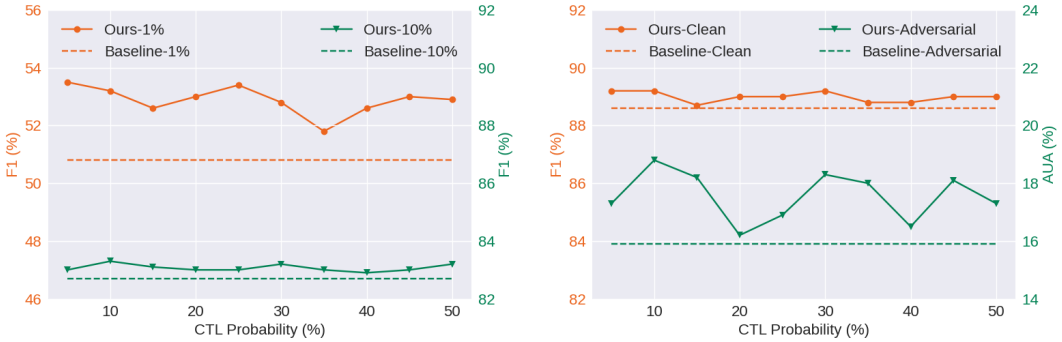
Attack Method	Method	Entity Freq	Entity Ratio	Entity AS
BAE	Normal-Train	89.0	51.1	38.2
	FreeLB	91.0	53.2	36.1
	TA-VAT	89.0	51.4	36.7
	Ours	**87.7**	**50.4**	**34.5**
TextFooler	Normal-Train	90.7	36.4	84.2
	FreeLB	**89.0**	36.7	85.2
	TA-VAT	89.7	36.5	86.9
	Ours	89.7	**35.4**	**80.0**

Table 8: Adversarial attack results of the entity on Wiki80. BAE and TextFooler are used as attackers.

6.2 Improvement on Robustness of Entity

Our Entity-Aware VAT method is first introduced to improve the robustness of entities against adversarial attacks. To investigate its effectiveness in improving entity robustness, we report Entity Freq, Entity Ratio, and Entity AS as we defined in Section 3. We choose to conduct experiments on the Wiki80 dataset here since it suffers the most from entity attacks, as indicated by the results of our pilot experiments in Section 3.

According to the results presented in Table 8, our method consistently reduces both the frequency of entity attacks and the ratio of perturbed entities compared to the normal-trained baseline and other VAT methods. This indicates that our method successfully reduces the model's reliance on entities for making predictions. Also, our method achieves a better performance in terms of entity AS, high-



(a) 1% and 10% training data (b) 100% training data

Figure 2: Different clean token leaving probability settings in SemEval. For 1% and 10% of the training data, we report the F1 score. For the 100% training data, we report both the F1 score and AUA score

lighting its effectiveness in improving the model's robustness toward entities.

6.3 Impact of Clean Token Leaving Probability

As we demonstrate in Section 6, the clean token leaving strategy is a very important design for improving model performance in both clean and adversarial samples. In this section, we train models with different clean token leaving probabilities to observe their influence on the model performance. We conduct the analysis on SemEval.

As Figure 2 shows, we add the model without any adversarial training as "Baseline" to have a comparison. It is notable that models with different clean token leaving probabilities consistently outperform baselines. Additionally, we notice models with different data availability usually achieve the best performance with a relatively small clean token leaving probability (0.05 – 0.15).

Method	SemEval	ReTACRED	Wiki80
Normal-Train	51.6	62.8	72.7
w/ DA	54.1	63.1	72.0
w/ Ours	53.9	**64.3**	73.3
w/ DA + Ours	**55.0**	64.0	**73.5**

Table 9: Experiment results with data augmentation on 1% training data of three datasets. For a fair comparison, we show the result of the optimal model from the development set of our approach.

6.4 Comparison and Compatibility with Data Augmentation

An important finding observed in Section 5.4 is adversarial training is especially effective in RE when the training data is limited. Data augmentation (Teru, 2023; Hu et al., 2023) is another widely used technique in low-resource RE. In this section, we conduct experiments using data augmentation to have a comparison and explore our method's compatibility with data augmentation. Currently, large language models (LLMs) show promising performance in generating diverse and high-quality content. To benchmark current LLMs' ability in augmenting RE samples, we prompt ChatGPT[5] to do data augmentation. We put details about the data augmentation method in Appendix H.

Table 9 shows the experiment results with 1% training data. While data augmentation brings improvement to SemEval and ReTACRED, it also leads to a non-trivial performance drop on Wiki80. Compared with that, our method consistently improves the model's performance in the three datasets. Also, combining data augmentation with our method achieves two best results over three datasets, showing our method's compatibility with data augmentation methods.

7 Conclusion

In this work, we present READ, a novel method that leverages an adversarial perspective for analyzing and enhancing RE models. Our adversarial attacks experiment on current SOTA RE models reveals their excessive reliance on entities for relation prediction. Through our analysis, this over-dependency is the underlying cause of the models' non-robustness to adversarial attacks and can limit the model's generalization. To tackle this issue, we propose an Entity-Aware Virtual Adversarial Training method. Experiment results show our method's effectiveness in improving the performance in both adversarial and clean samples.

[5]https://platform.openai.com/docs/mode

8 Limitations

This work introduces an Entity-Aware Virtual Adversarial Training method. Similar to other virtual adversarial training algorithms, our method incorporates search perturbation in each mini-batch, leading to a relatively longer training time compared to other normal-trained models. Due to limitations in computing resources, we evaluate our method on four RE datasets, while disregarding scenarios such as continual relation extraction (Han et al., 2020b) and few-shot relation extraction (Gao et al., 2019). In future research, we plan to investigate the effectiveness of our method in border scenarios.

References

Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. Ieee.

Tao Chen, Haochen Shi, Liyuan Liu, Siliang Tang, Jian Shao, Zhigang Chen, and Yueting Zhuang. 2021. Empower distantly supervised relation extraction with collaborative adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12675–12682.

Yong Cheng, Lu Jiang, and Wolfgang Macherey. 2019. Robust neural machine translation with doubly adversarial inputs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4324–4333.

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 (Volume 2: Short Papers)*, 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 (SPW)*, pages 50–56. IEEE.

Tianyu Gao, Xu Han, Hao Zhu, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2019. Fewrel 2.0: Towards more challenging few-shot relation classification. 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 6250–6255.

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.

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

Wenjuan Han, Liwen Zhang, Yong Jiang, and Kewei Tu. 2020a. Adversarial attack and defense of structured prediction models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2327–2338.

Xu Han, Yi Dai, Tianyu Gao, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2020b. Continual relation learning via episodic memory activation and reconsolidation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6429–6440.

Xu Han, Tianyu Gao, Yankai Lin, Hao Peng, Yaoliang Yang, Chaojun Xiao, Zhiyuan Liu, Peng Li, Jie Zhou, and Maosong Sun. 2020c. More data, more relations, more context and more openness: A review and outlook for relation extraction. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 745–758.

Xu Han, Tianyu Gao, Yuan Yao, Deming Ye, Zhiyuan Liu, and Maosong Sun. 2019. Opennre: An open and extensible toolkit for neural relation extraction. 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): System Demonstrations*, pages 169–174.

Kailong Hao, Botao Yu, and Wei Hu. 2021. Knowing false negatives: An adversarial training method for distantly supervised relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9661–9672.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid O Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2019. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. *arXiv preprint arXiv:1911.10422*.

William Hogan, Jiacheng Li, and Jingbo Shang. 2022. Fine-grained contrastive learning for relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1083–1095.

Xuming Hu, Aiwei Liu, Zeqi Tan, Xin Zhang, Chenwei Zhang, Irwin King, and Philip S Yu. 2023. Gda: Generative data augmentation techniques for relation extraction tasks. *arXiv preprint arXiv:2305.16663*.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885.

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 *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025.

Dongyeop Kang, Tushar Khot, Ashish Sabharwal, and Eduard Hovy. 2018. Adventure: Adversarial training for textual entailment with knowledge-guided examples. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2418–2428.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Emanuele La Malfa and Marta Kwiatkowska. 2022. The king is naked: on the notion of robustness for natural language processing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11047–11057.

Yibin Lei, Yu Cao, Dianqi Li, Tianyi Zhou, Meng Fang, and Mykola Pechenizkiy. 2022. Phrase-level textual adversarial attack with label preservation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1095–1112.

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

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

Linyang Li and Xipeng Qiu. 2021. Token-aware virtual adversarial training in natural language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8410–8418.

Linyang Li, Demin Song, and Xipeng Qiu. 2023a. Text adversarial purification as defense against adversarial attacks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 338–350, Toronto, Canada. Association for Computational Linguistics.

Wanli Li, Tieyun Qian, Xuhui Li, and Lixin Zou. 2023b. Adversarial multi-teacher distillation for semi-supervised relation extraction. *IEEE Transactions on Neural Networks and Learning Systems*.

Zongyi Li, Jianhan Xu, Jiehang Zeng, Linyang Li, Xiaoqing Zheng, Qi Zhang, and Cho-Jui Hsieh. 2021. Searching for an effective defender: Benchmarking defense against adversarial word substitution. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

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. *arXiv preprint arXiv:1907.11692*.

Chonggang Lu, Richong Zhang, Kai Sun, Jaein Kim, Cunwang Zhang, and Yongyi Mao. 2023. Anaphor assisted document-level relation extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15453–15464.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232.

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

Dheeraj Mekala, Chengyu Dong, and Jingbo Shang. 2022. Lops: Learning order inspired pseudo-label selection for weakly supervised text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4894–4908.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1105–1116.

Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2016. Adversarial training methods for semi-supervised text classification. In *International Conference on Learning Representations*.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. 2018. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993.

R Mooney. 1999. Relational learning of pattern-match rules for information extraction. In *Proceedings of the sixteenth national conference on artificial intelligence*, volume 328, page 334.

Hao Peng, Tianyu Gao, Xu Han, Yankai Lin, Peng Li, Zhiyuan Liu, Maosong Sun, and Jie Zhou. 2020. Learning from context or names? an empirical study on neural relation extraction. In *Proceedings of the*

2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 3661–3672.

Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen-tau Yih. 2017. Cross-sentence n-ary relation extraction with graph lstms. *Transactions of the Association for Computational Linguistics*, 5:101–115.

Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Dsgan: Generative adversarial training for distant supervision relation extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 496–505.

Yujia Qin, Yankai Lin, Ryuichi Takanobu, Zhiyuan Liu, Peng Li, Heng Ji, Minlie Huang, Maosong Sun, and Jie Zhou. 2021. Erica: Improving entity and relation understanding for pre-trained language models via contrastive learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3350–3363.

Chris Quirk and Hoifung Poon. 2017. Distant supervision for relation extraction beyond the sentence boundary. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1171–1182.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 1085–1097.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part III 21*, pages 148–163. Springer.

Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255*.

Livio Baldini Soares, Nicholas Fitzgerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905.

George Stoica, Emmanouil Antonios Platanios, and Barnabás Póczos. 2021. Re-tacred: Addressing shortcomings of the tacred dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13843–13850.

Haitian Sun, Patrick Verga, Bhuwan Dhingra, Ruslan Salakhutdinov, and William W Cohen. 2021. Reasoning over virtual knowledge bases with open predicate relations. In *International Conference on Machine Learning*, pages 9966–9977. PMLR.

Qingyu Tan, Lu Xu, Lidong Bing, Hwee Tou Ng, and Sharifah Mahani Aljunied. 2022. Revisiting docred-addressing the false negative problem in relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8472–8487.

Komal Teru. 2023. Semi-supervised relation extraction via data augmentation and consistency-training. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1104–1116.

Xiaosen Wang, Jin Hao, Yichen Yang, and Kun He. 2021. Natural language adversarial defense through synonym encoding. In *Uncertainty in Artificial Intelligence*, pages 823–833. PMLR.

Yiwei Wang, Muhao Chen, Wenxuan Zhou, Yujun Cai, Yuxuan Liang, Dayiheng Liu, Baosong Yang, Juncheng Liu, and Bryan Hooi. 2022. Should we rely on entity mentions for relation extraction? debiasing relation extraction with counterfactual analysis. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3071–3081.

Yi Wu, David Bamman, and Stuart Russell. 2017. Adversarial training for relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1778–1783.

Jianhan Xu, Linyang Li, Jiping Zhang, Xiaoqing Zheng, Kai-Wei Chang, Cho-Jui Hsieh, and Xuan-Jing Huang. 2022a. Weight perturbation as defense against adversarial word substitutions. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7054–7063.

Jianhan Xu, Cenyuan Zhang, Xiaoqing Zheng, Linyang Li, Cho-Jui Hsieh, Kai-Wei Chang, and Xuan-Jing Huang. 2022b. Towards adversarially robust text classifiers by learning to reweight clean examples. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1694–1707.

Yichen Yang, Xiaosen Wang, and Kun He. 2022. Robust textual embedding against word-level adversarial attacks. In *Uncertainty in Artificial Intelligence*, pages 2214–2224. PMLR.

Jin Yong Yoo and Yanjun Qi. 2021. Towards improving adversarial training of nlp models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 945–956.

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.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2021. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115.

Dongxu Zhang and Dong Wang. 2015. Relation classification via recurrent neural network. *arXiv preprint arXiv:1508.01006*.

Mi Zhang, Tieyun Qian, Ting Zhang, and Xin Miao. 2023. Towards model robustness: Generating contextual counterfactuals for entities in relation extraction. In *Proceedings of the ACM Web Conference 2023*, pages 1832–1842.

Ningyu Zhang, Shumin Deng, Zhanlin Sun, Jiaoyan Chen, Wei Zhang, and Huajun Chen. 2020. Relation adversarial network for low resource knowledge graph completion. In *Proceedings of the web conference 2020*, pages 1–12.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2019. Freelb: Enhanced adversarial training for natural language understanding. In *International Conference on Learning Representations*.

A Text Substitution Method

In this section, we conduct an experiment using the text substitution method. Specifically, we follow (Li et al., 2020) and utilize a BERT model to replace the critical token which can mislead the model most to produce the adversarial samples. We conduct evaluation using FineCL, on SemEval and ReTACRED with 1% and 10% training data. As Table 10 shows, while BERT-Attack improves the model's performance on SemEval, it also leads to a non-trivial performance drop on ReTACRED. This finding aligns with some previous works that point out the traditional text substitution method could cause a performance drop in the clean test set (Yoo and Qi, 2021; Xu et al., 2022b).

	SemEval		ReTACRED	
	1%	10%	1%	10%
FineCL	50.8	82.7	**62.8**	**83.2**
FineCL +BERT-Attack	**53.1**	**83.6**	62.7	82.7

Table 10: Experiment result using BERT-Attack (Li et al., 2020) on FineCL.

B A Detailed Survey of Adversarial Attack & Training

In the computer vision field, adversarial attacks (Goodfellow et al., 2014; Carlini and Wagner, 2017) have been widely explored since it is easy to implement over the continual space of images. Based on the gradient-based adversarial attacks, various adversarial training (Goodfellow et al., 2014; Madry et al., 2018) are proposed. They add the adversarial sample for the training set to make the model more robust under adversarial attacks. One major problem of directly applying this gradient-based adversarial training method in NLP is the discrete text prevents the gradient from propagating.

To introduce adversarial training into NLP, some works adopt text substitution as an alternative method to generate adversarial samples (Li et al., 2018; Jin et al., 2020; Garg and Ramakrishnan, 2020). This method always involves replacing the original word with its synonym based on certain criteria like word embedding similarity (Zang et al., 2020; Ren et al., 2019; Jin et al., 2020) or model infilling (Garg and Ramakrishnan, 2020; Li et al., 2020). Another commonly used approach to produce adversarial samples is to generate them with a sequence-to-sequence model (Kang et al., 2018; Han et al., 2020a; La Malfa and Kwiatkowska, 2022).

In contrast, virtual adversarial training (VAT) methods generate adversarial samples by applying perturbations to the embedding space (Miyato et al., 2018). That helps VAT become more efficient than traditional text substitution methods. VAT makes the model more robust under adversarial attacks while also improving the model's performance in clean test samples (Miyato et al., 2016; Cheng et al., 2019). To make VAT more effective, Zhu et al. (2019) accumulate perturbation in multiple searching steps to craft adversarial examples. Li and Qiu (2021) devise a Token-Aware VAT (TA-VAT) method to allocate more attack budget to the important tokens in the sequence. Following them, Xu et al. (2022a) combines weight perturbation with embedding perturbation in training to make the model more robust against text adversarial attacks. While there are some works that apply virtual adversarial training methods to RE for different purpose (Wu et al., 2017; Chen et al., 2021), we propose an Entity-Aware VAT method explicitly designed for RE to mitigate over-dependency and non-generalization on entities.

Beyond (virtual) adversarial training, there are also many other techniques proposed as defense mechanisms to adversarial attacks. For example, some works focus on detecting the adversarial samples and correcting them before inputting them into the language model (Wang et al., 2021; Yang et al., 2022; Li et al., 2023a). However, our goal in this paper is to improve the RE models' robustness during training. Such plug-in methods outside the models are not within the scope of our consideration.

C Attack Result on ERICA

We also conduct adversarial attacks on ERICA and put the results in Table 11. ERICA exhibited a significant decrease in performance across all attack methods, particularly with TextFooler. Our analysis of learning preference and entity generalization in ERICA is presented in Table 12 and Table 13. The

high frequency of successful attacks and their success rate on entities indicates that over-dependency and poor-generalization on entities are ubiquitous in RE models.

Dataset	Clean	PGD		TextBugger		BAE		TextFooler	
		AUA	Query	AUA	Query	AUA	Query	AUA	Query
SemEval	93.3	46.1	7.44	38.1	40.47	25.3	99.74	9.1	83.19
Retacred	89.5	56.8	7.87	27.0	83.90	37.2	124.21	25.2	221.05
Wiki80	96.1	68.0	8.56	27.7	53.95	15.7	74.61	12.1	118.96

Table 11: Adversarial attack results with ERICA. The attack settings and metrics align with the ones used in 3.1.

	Entity Freq	Entity Ratio	Entity %
SemEval	72.7	30.8	12.0
ReTACRED	55.7	13.8	9.2
Wiki80	85.3	31.6	17.4

Table 12: Analysis of ERICA's learning preference with TextFooler.

	Entity-AS	Context-AS
SemEval	86.0	81.8
ReTACRED	56.0	45.5
Wiki80	79.5	71.6

Table 13: Attack success (AS) rate of entity and context on ERICA with TextFooler.

D Details of Entity-aware Virtual Adversarial Training

We give a detailed algorithm for our Entity-aware Virtual Adversarial Training in Algorithm 1.

E Our Method on ERICA

The performance of our method with ERICA is presented in Table 14. It is evident that with our method, ERICA also demonstrates a non-trivial improvement in each data availability across three RE datasets.

Dataset	SemEval			ReTACRED			Wiki80		
Size	1%	10%	100%	1%	10%	100%	1%	10%	100%
ERICA	50.2	82.0	88.5	64.1	83.4	87.8	71.3	86.8	91.6
ERICA +Ours	**51.8**	**82.6**	**89.1**	**64.6**	**84.8**	**88.8**	**71.6**	**87.0**	**91.8**

Table 14: Experiment results of ERICA on clean samples of each dataset.

F Our method in Document-level RE

To demonstrate the compatibility of our proposed entity-aware VAT method across various RE scenarios, we conduct an experiment in a document-level RE dataset, Re-DocRED (Tan et al., 2022) and report the results in Table 15.

G Training Details

In our method, we have set the clean token leaving probability to 10% for SemEval and 15% for ReTACRED and Wiki80 datasets. Following the approach of Hogan et al. (2022), the compared models employ the following settings: a batch size of 64, a maximum sequence length of 100, a learning rate of 5e-5, an Adam epsilon of 1e-8, a weight decay of 1e-5, a maximum gradient norm of 1.0, 500 warm-up steps, and a hidden size of 768. To account for different data availability scenarios, we utilize dropout rates of 0.2/0.1/0.35 and set the maximum number of training epochs to 80/20/8 for training proportions of 0.01/0.1/1.0, respectively.

Algorithm 1 Detailed process of our Entity-Aware Virtual Adversarial Training. We use // to highlight the important steps.

Require: Training Samples $S = (X = [w_0, ...w_i, ...], y)$, perturbation bound ϵ, initialize bound σ, adversarial steps K, adversarial step size α, model parameter θ, clean token leaving probability n

$\boldsymbol{V}_e \in \mathbb{R}^{N \times D} \leftarrow \frac{1}{\sqrt{D}} U(-\sigma, \sigma)$, $\boldsymbol{V}_c \in \mathbb{R}^{N \times D} \leftarrow \frac{1}{\sqrt{D}} U(-\sigma, \sigma)$ // Separate Vocabulary Initialization

for epoch = 1, ..., **do**

  for batch $B \in S$ **do**

    $\boldsymbol{\eta}_0^i = \begin{cases} \boldsymbol{V}_e[w_i], & w_i \in Entity \\ \boldsymbol{V}_c[w_i], & w_i \in Context \end{cases}$ // Separate Token-level Perturbation Initialization

    $\boldsymbol{\delta}_0 \leftarrow \frac{1}{\sqrt{D}} U(-\sigma, \sigma)$, $\boldsymbol{g}_0 \leftarrow 0$

    $W_c = RandomlySelect(Context, n)$ // Clean Token Leaving in Context

    for t = 1, ..., K **do**

      $\boldsymbol{X}_{adv}^i = \begin{cases} \boldsymbol{X}^i, & w_i \in W_c, \\ \boldsymbol{X}^i + \boldsymbol{\delta}_t + \boldsymbol{\eta}_t^i, & Otherwise \end{cases}$

      $\boldsymbol{g}_t \leftarrow \boldsymbol{g}_{t-1} + \frac{1}{K} \mathbb{E}_{(X,y) \in B} \left[\nabla_\theta L(f_\theta(\boldsymbol{X}_{adv}), y) \right]$

      $\boldsymbol{g}_\eta^i \leftarrow \nabla_{\eta^i} L(f_\theta(\boldsymbol{X}_{adv}), y)$

      $\boldsymbol{\eta}_t^i \leftarrow n_i * (\boldsymbol{\eta}_{t-1}^i + \alpha \cdot \boldsymbol{g}_\eta^i) / ||\boldsymbol{g}_\eta^i||_F)$

      $\boldsymbol{\eta}_t \leftarrow \prod_{||\boldsymbol{\eta}||_F < \epsilon} (\boldsymbol{\eta}_t)$

      $\boldsymbol{g}_\delta \leftarrow \nabla_\delta L(f_\theta(\boldsymbol{X}_{adv}), y)$

      $\boldsymbol{\delta}_t \leftarrow \prod_{||\boldsymbol{\delta}||_F < \epsilon} (\boldsymbol{\delta}_{t-1} + \alpha \cdot \boldsymbol{g}_\delta) / ||\boldsymbol{g}_\delta||_F)$

    end for

    $\boldsymbol{V}_e[w_i] \leftarrow \eta_K^i, w_i \in Entity$ // Entity Vocabulary Update

    $\boldsymbol{V}_c[w_i] \leftarrow \eta_K^i, w_i \in Context$ // Context Vocabulary Update

    $\theta \leftarrow \theta - g_K$

  end for

end for

For all the adversarial training methods, we search adversarial learning rate in [2e-2, 5e-2, 1e-1], attack budget in [2e-1, 4e-1, 6e-1], and perturbation searching steps in [1,2,3]. For each experiment, we employ grid search[6] to discover the above hyperparameters, and we report the average results of the top three configurations based on their scores in the development set.

We train all models on a single A6000 GPU with CUDA version 11.1. The training time for a RE model ranges from approximately 20 to 60 minutes, depending on the specific dataset and availability settings.

H Data Augmentation with ChatGPT

We use the model 'GPT-3.5-turbo-0301' to generate augmented data for 1% training data availability of each dataset. For each sample, we randomly choose other two samples with the same relation labels and input them into the model as demonstrations. After getting output from ChatGPT, we verify that the sentence includes both entities mentioned. If not, we discard the generated output. We provide an example of the prompt we use in Table 16.

[6] https://wandb.ai/

	Ign-F1	F1
ATLOP*	76.94	77.73
DocuNet*	77.27	77.92
KD-DocRE*	77.63	78.35
DREEAM*	79.66	80.73
PEMSCL*	79.01	79.86
AA	80.39	81.34
AA + Ours	**81.21**	**82.22**

Table 15: Experimental results on Re-DocRED dataset. We apply our entity-aware VAT method on AA (Lu et al., 2023) and * denote the results we take from Lu et al. (2023).

Prompt	Read the following examples of the relation 'Component-Whole(e2,e1)' between the head and tail and write another new example following the same format. Note that the sentence must contain both head and tail: head: kangaroo, tail: legs, sentence: the kangaroo moves by hopping on its hind legs using its tail for steering and balancing while hopping at speed up to 40mph/60kmh. head: cottage, tail: kitchen, sentence: the cottage kitchen is on the first floor and is fully fitted with fridge, dishwasher, microwave and all the standard self catering facilities. head: armature, tail: coil, sentence: the armature has a coil of wire wrapped around an iron core.

Table 16: An example of the prompt we use to generate augmented samples.