Attack Named Entity Recognition by Entity Boundary Interference

Yifei Yang[1,2,†], **Hongqiu Wu**[1,2,†] **and Hai Zhao**[1,2,*]
[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
{yifeiyang, wuhongqiu}@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn

Abstract

Named Entity Recognition (NER) is a cornerstone natural language processing task while its robustness has been given little attention. This paper rethinks the principles of the conventional text attack, as they can easily violate the label consistency between the original and adversarial NER samples. This is due to the fine-grained nature of NER, as even minor word changes in the sentence can result in the emergence or mutation of any entity, producing invalid adversarial samples. To this end, we propose a novel one-word modification NER attack based on a key insight, NER models are always vulnerable to the boundary position of an entity to make their decision. We thus strategically insert a new boundary into the sentence and trigger the victim model to make a wrong recognition either on this boundary word or on other words in the sentence. We call this attack *Virtual Boundary Attack (ViBA)*, which is shown to be remarkably effective when attacking both English and Chinese models with a 70%-90% attack success rate on state-of-the-art language models, and also significantly faster than previous methods. We share the code in https://github.com/yangyifei729/ViBA.

Keywords: Named Entity Recognition, Explainability, Neural language representation models

1. Introduction

The goal of Named Entity Recognition (NER) is to find the predefined named entities, such as locations, persons, and organizations in a given sentence. It is a fundamental task in natural language processing (NLP) behind various downstream applications (Clark et al., 2018; Sil and Yates, 2013; Babych and Hartley, 2003; Nikoulina et al., 2012).

Language models have been shown to be vulnerable to cunningly crafted input data, producing misjudgments, thereby undermining their security and trustworthiness. Great attention has been paid to the robustness of natural language understanding (NLU), e.g., sentence classification (Jin et al., 2020; Garg and Ramakrishnan, 2020), question answering (Gan and Ng, 2019; Ribeiro et al., 2018), to unravel their vulnerabilities and deficiencies, for the sake of providing defense techniques. However, the study on the robustness of sequence labeling tasks like NER is still lacking.

Recently, Simoncini and Spanakis (2021) made an initial foray into the field of attacking NER models, taking inspiration from text attack methods designed for sentence classification and adapting them to NER tasks. In a parallel vein, Lin et al. (2021) introduce RockNER, an adversarial dataset generated through word substitution, a well-established technique commonly used to attack sentence classification models.

However, we find that the conventional princi-

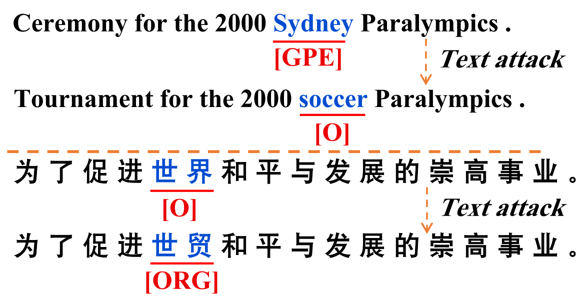

Figure 1: Label shift issue on English and Chinese adversarial samples.

ples of text attack on sentence classification can easily violate the label consistency between original and adversarial NER samples. Specifically, these attackers apply word insertion, swapping, or substitution to the sentence while maintaining its semantics to keep the sentence label unchanged as possible. Note that the labels of NLU tasks are greatly correlated to the semantics. As opposed to the sentence classification task, NER is often modelled as a fine-grained structure labeling task. In this context, any minor word changes like insertion, swapping, and substitution, can result in the emergence of new entities or mutation of original entities.

We denote this issue as *label shift*. We show two cases in Figure 1, where a GPE (geopolitical) entity *Sydney* in the original sentence is substituted to *soccer*, and *world* (“世界”) is substituted to *WTO* (“世贸”) by the attacker (Morris et al., 2020). However, *soccer* is obviously not a GPE entity and *WTO* should be an ORG (organization). As a result, these two are invalid adversarial samples. We find

* Corresponding author. † Equal contribution. This work was supported by Joint Research Project of Yangtze River Delta Science and Technology Innovation Community (No. 2022CSJGG1400).

LREC-COLING 2024, pages 1734–1744

such an issue widely exists in current attackers, which has a significant negative impact on NER adversarial samples.

The fine-grained nature of NER determines that one should make as few modifications as possible to the sentence in order not to incur label shift. Thus in this paper, we propose a novel NER attacker, which only modifies one word in the original sentence to maximumly alleviate label shift.

Our method is based on a key insight that the nowadays NER models concern more on the boundary tokens and tend to memory them for entity recognition. Specifically, when inserting a boundary token (i.e., the leftmost and rightmost token of the entity) into the sentence, the state-of-the-art NER models can be easily fooled and exhibit abnormal behaviors. We refer to this phenomenon as *Entity Boundary Interference* (EBI), and our attack is a natural extension of it.

The contributions of this paper are below:

- We first reveals the problem of Entity Boundary Interference (EBI). Based on it, we propose *Virtual Boundary Attack* (ViBA), a novel NER attacker which avoids the label shift problem that other attackers suffer from. We evaluate ViBA on several state-of-the-art pre-trained language models (PrLMs) on widely used English and Chinese benchmarks. Experiments show that ViBA has a high attack success rate and also maintains a high semantic and syntax similarity with the original sentences. Furthermore, it exhibits exceptional fluency and has a good efficiency advantage with almost a linear time complexity.
- We undertake a comprehensive analysis of the factors contributing to EBI and elucidate how ViBA's effectiveness is influenced.
- We propose two defense techniques to train robust NER models against EBI. Our defense strategy has also been demonstrated to withstand various word substitution NER attackers.

2. Method

2.1. Entity Boundary Interference

Previous studies assume that an NER model is heavily reliant on the boundary of an entity when making decisions (Peng and Dredze, 2016; Tan et al., 2020a). Given an entity, the boundary refers to its leftmost or rightmost token. In light of this assumption, our vision is that NER models can be vulnerable if the attackers attempt to manipulate these boundary tokens. Figure 2 demonstrates two representative phenomenons where the model falls into mistakes when there is a new boundary inserted in the sentence at some positions.

- **S1:** Insertion of a semantically unrelated boundary may change the predictions of other entities. As shown in Figure 2 (S1), the model correctly recognizes *Paul Fischer* as a PER (Person) entity in (S1.a). When we insert the right boundary *Fischer* at the beginning of the sentence in (S1.b), surprisingly, the model no longer recognizes *Paul Fischer* as a PER, even if it still is. Apparently, humans will not make such a mistake.
- **S2:** The model may mistakenly assume a correlation between the inserted boundary and the original entity. In Figure 2 (S2), the model first wrongly recognizes the inserted *South* as a GPE in (S2.b). Paradoxically, it is no more after the original entity *South Korea* is masked in (S2.c). It indicates that the model pathologically assumes the co-occurring boundaries are relevant, which is different from the way humans perceive text and should be regarded as another non-robust phenomenon.

S1 and S2 show that there is a coupling effect between the model recognition of different entities in the sentence. In S1, the emergence of a new entity *Fischer* causes a flip in the prediction of *Paul Fischer*. In S2, the erasure (being masked) of an original entity *South Korea* causes a miss recall of another entity *South*. The underlying is that the prediction of *South* is coupled with the co-occurrence of *South Korea*. We notice that these entities are supposed not to have any connection. We denote the above phenomenon as *Entity Boundary Interference* (EBI) issue.

2.2. ViBA

We introduce *Virtual Boundary Attack* (ViBA), a novel attack algorithm for NER models based on our finding of EBI. ViBA attacks the model by inserting a boundary token of some entities into the sentence. The goal is to induce wrong predictions of the model, as in S1 and S2. We denote the inserted boundary as a "virtual boundary" for the reason that the inserted boundary is not a real entity. Algorithm 1 summarizes the procedure of ViBA:

(1) Prepare to Attack (line 1-3)

Given an input sentence $\mathcal{X} = \mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$, we first feed it to the victim model to obtain the original prediction $\mathcal{Y}$, which is a list of predicted named entity tags corresponding to $\mathcal{X}$. Each tag in $\mathcal{Y}$ is a predefined abbreviated label such as "PER" (Person), "LOC" (Location), etc. Following the convention, "O" refers to a non-entity token. Then we cache a set $\mathcal{E}$ of all the named entities as well as their corresponding positions $\mathcal{L}$ in the sentence.

(2) Restrict Safety Areas (line 4)

We introduce safety areas to keep the original entity tags unchanged. First, it is not allowed to insert a boundary inside an entity because it would undermine the entity and trigger label shift. Second, the entity tag is likely to mutate when its local context changes. For example, the inserted boundary may form a new entity with its surrounding tokens.

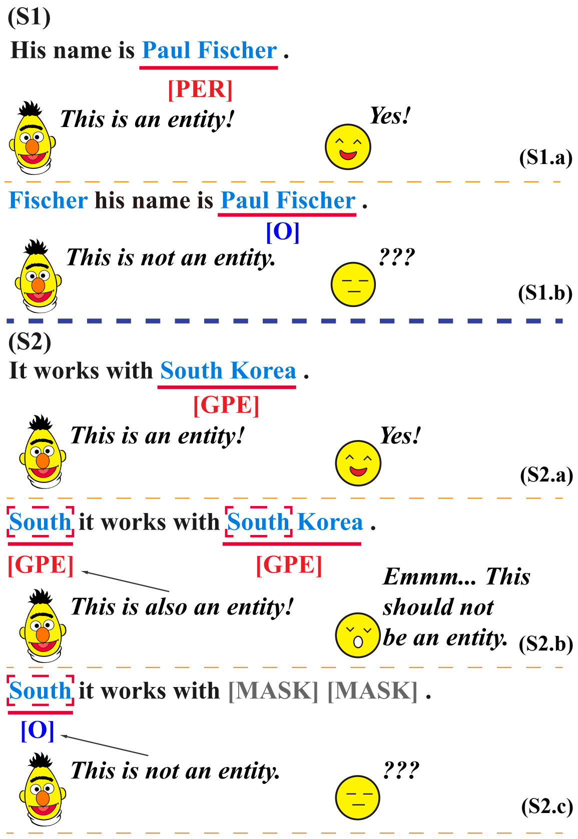

Figure 2: Demonstration of Entity Boundary Interference.

The safety areas are obtained by setting a safety distance w. A case is shown in Figure 3.

(3) Attack (line 5-9)

We next generate the candidate adversarial samples. We pick the leftmost and rightmost boundaries of all named entities e in $\mathcal{E}$. For each boundary b, we go through every position in the sentence outside the safety areas and insert the boundary to generate a candidate sample $\mathcal{X}'$.

(4) Check Success (line 10-17)

We feed $\mathcal{X}'$ to the victim model and obtain its prediction $\mathcal{Y}'$. The following two criteria are applied to determine whether an attack is successful:

Criterion 1 (line 10-12) This criterion corresponds to the S1 case in Figure 2, that the inserted token should not affect the predictions of the original entities. Note that we also set a safety area for the inserted position during the comparison in order to avoid label shift in case the inserted boundary is an entity or forms a new entity with surrounding tokens. What we do is to check the consistency of $\mathcal{Y}$ and $\mathcal{Y}'$, and any inconsistency indicates the success of the attack.

Criterion 2 (line 13-17) This criterion corresponds to the S2 case in Figure 2, that the model prediction of the virtual boundary should not change after we mask its referential entity. We mask the named entity e in $\mathcal{X}'$ and get $\mathcal{X}'_m$. Then we feed it to the victim model and get $\mathcal{Y}'_m$. Any inconsistent prediction of b between $\mathcal{Y}'$ and $\mathcal{Y}'_m$ indicates the success of the attack.

It is worth noting that ViBA maximumly avoids the label shift issue by the following properties:

• The safety areas guarantee that the original entity tags will not be changed after the insertion of a new boundary.

• Criterion 2 is independent of labels since we only care about the consistency of the prediction of the virtual boundary.

Algorithm 1 Virtual Boundary Attack

Input: Victim model $\mathcal{F}$, input sample $\mathcal{X}$, safety distance w.
Output: Adversarial sample $\mathfrak{X}$.

1: $\mathcal{Y} \leftarrow \mathcal{F}(\mathcal{X})$
2: $\mathcal{E} \leftarrow$ Extract each entity in $\mathcal{X}$ following $\mathcal{Y}$
3: $\mathcal{L} \leftarrow$ Locate each entity in $\mathcal{X}$ following $\mathcal{Y}$
4: $\mathcal{S} \leftarrow$ Decide safety area following $\mathcal{L}$ and w
5: **for** e in $\mathcal{E}$ **do**
6:     **for** j in $\{1 \sim n\} \setminus \mathcal{S}$ **do**
7:         **for** b in $\{e^{left}, e^{right}\}$ **do**
8:             $\mathcal{X}' \leftarrow$ Insert b before $\mathcal{X}_{[j]}$
9:             $\mathcal{Y}' \leftarrow \mathcal{F}(\mathcal{X}')$
10:             **if** $\mathcal{Y}' \setminus \mathcal{Y}'_{[j-w:j+w+1]} \neq \mathcal{Y}$ **then**
11:                 **return** $\mathcal{X}'$
12:             **end if**
13:             $\mathcal{X}'_m \leftarrow$ Mask e in $\mathcal{X}'$
14:             $\mathcal{Y}'_m \leftarrow \mathcal{F}(\mathcal{X}'_m)$
15:             **if** $\mathcal{Y}'_{[j]} \neq \mathcal{Y}'_{m[j]}$ **then**
16:                 **return** $\mathcal{X}'$
17:             **end if**
18:         **end for**
19:     **end for**
20: **end for**
21: **return** None

Figure 3: A case of safety areas.

3. Experiments

3.1. Datasets

We explore the effectiveness of ViBA on three widely used benchmarks of Chinese and English:

• **OntoNotes5.0** (Weischedel et al., 2013) is a multilingual NER dataset of Chinese, English and Arabic. There are eighteen types of named entities, eleven of which are types like Person, Organization and seven are values such as Date and Percent. In this paper, we select the popular Chinese and English versions for our experiments.

Test set	WNUT	OntoNotes-en	MSRA	OntoNotes-ch
Samples	686 / 1287	4561 / 9479	2344 / 4365	2392 / 4472
Entities per sample	1.57	2.45	2.61	3.13
Tokens per sample	19.67	24.08	47.3	45.06

Table 1: Statistics for each used test set.

• **MSRA** (Levow, 2006) is one of the commonly used Chinese NER datasets which accommodates three named entity types and the data in MSRA are collected from the news domain.

• **WNUT2017** (Derczynski et al., 2017) is an English NER dataset which has six types. It focuses on identifying unusual, previously-unseen entities and is more challenging.

These benchmarks have standard train/dev/test split. Some statistical data of the test sets are shown in Table 1. The total number of sentences containing at least one entity / the sizes of datasets are shown in the Samples row. We also count the average amount of entities in each sentence and the average sentence length.

3.2. Metric

• **Attack Success Rate (ASR)** is the main measurement of the attacker's effectiveness towards a victim model, which is the ratio of the achieved adversarial samples over all samples. A higher ASR suggests a more effective attacker.

• **Semantic Similarity (SS)** measures semantic distance between two sentences. We leverage *text2vec* for evaluation (Xu, 2022). A greater SS suggests the semantics of the adversarial sample are close to the original one.

• **Entity-Level Attack Success Rate (EASR)** is a ViBA-specific metric which is the proportion of entities that can successfully trigger Entity Boundary Interference out of all entities. EASR1 and EASR2 imply how frequently S1 and S2 occur.

• **Edit Distance (ED)** reflects the syntax similarity between two sentences. We expect to generate an adversarial sample with a high overlap with the original one.

• **Fluency (FLU)** reflects the smoothness and naturalness of generated adversarial samples. We prompt the state-of-the-art AI model, ChatGPT, to play the role of a professional linguist and provide fluency scores between 0 and 100 for the sentences, using their average as the metric.

3.3. Settings

We evaluate ViBA on the extractive models, specifically the BERT-base (Devlin et al., 2019), RoBERTa-large (Liu et al., 2019) models of Chinese and English versions. In addition, DeBERTa-large (He et al., 2020) is leveraged for the evaluation of the English datasets. MacBERT-large (Cui et al., 2020) is used for the Chinese datasets. We first fine-tune the models with multilayer perceptron (MLP) as the classification heads on the training sets for 6 epochs and select the best-trained checkpoints by dev sets. Then we apply ViBA to attack them on the test sets. We have heuristically set the safety distance $w = 2$. We conduct experiments on a single NVIDIA RTX 3090 GPU. It is worth noting that according to the latest researches (Wang et al., 2023; Xie et al., 2023), BERT-like models still remain the state-of-the-art models for NER. Hence, in this paper, we refrain from including generative models such as ChatGPT[1] for this purpose.

	English			
	WNUT		**OntoNotes**	
	ASR	*SS*	*ASR*	*SS*
BERT$_{\text{base}}$	57.1	98.0	73.2	98.1
	59.6	95.4	**75.1**	96.5
RoBERTa$_{\text{large}}$	67.1	97.9	70.0	98.1
	67.8	95.5	73.0	96.4
DeBERT$_{\text{large}}$	56.1	98.0	70.7	98.1
	62.5	95.7	74.7	96.4
	Chinese			
	MSRA		**OntoNotes**	
	ASR	*SS*	*ASR*	*SS*
BERT$_{\text{base}}$	91.2	98.8	85.5	98.7
	91.4	98.4	86.4	98.2
RoBERTa$_{\text{large}}$	91.7	98.8	86.9	98.1
	92.3	98.3	89.1	98.2
MacBERT$_{\text{large}}$	**93.2**	98.8	89.4	98.6
	92.0	98.3	**89.8**	98.1

Table 2: Attack success rate (ASR) and semantic similarity (SS) across various NER datasets. For a victim model, the top row corresponds to ViBA, and the bottom corresponds to ViBA-rep.

[1]https://chat.openai.com/

	English			
	WNUT		**OntoNotes**	
	EASR1	*EASR2*	*EASR1*	*EASR2*
BERT$_{\mathrm{base}}$	54.9	75.7	22.3	55.4
RoBERTa$_{\mathrm{large}}$	41.0	67.9	17.1	52.6
DeBERTa$_{\mathrm{large}}$	42.6	73.6	14.5	51.0
	Chinese			
	MSRA		**OntoNotes**	
	EASR1	*EASR2*	*EASR1*	*EASR2*
BERT$_{\mathrm{base}}$	42.6	75.8	55.4	61.8
RoBERTa$_{\mathrm{large}}$	37.1	76.8	56.6	65.7
MacBERT$_{\mathrm{large}}$	46.5	77.4	60.5	71.8

Table 3: EASR for ViBA on different datasets.

3.4. Main Results

We evaluate ViBA for multiple models on different Chinese and English datasets, and the results are shown in Table 2. Considering that the insertion will change the length of the sentence and cause too obvious a distinction, we also change the "insert" operation in ViBA to the "replace" operation for comparison, named ViBA-rep. Overall, ViBA achieves high ASR when attacking both Chinese and English datasets. The ASR on the Chinese datasets is as high as 85% - 93%. Although relatively lower on the English datasets, the ASR ranges from 55% to 73%, which is still an ideal performance. It is noteworthy that the English datasets generally have shorter sentences and fewer entities. Their smaller search spaces will lead to relatively lower ASR. Comprehensively, ViBA is an ideal attacker on both English and Chinese.

In Table 2, the average SS between the adversarial and original samples of ViBA on all datasets exceeds 97.9, which guarantees that (1) the semantics of the adversarial samples are extremely close to the original ones; (2) the adversarial samples are natural and look similar to the original ones.

Generally, ViBA-rep exhibits higher ASR than vanilla ViBA. But replacement fails to retain all the tokens and generates samples with a greater semantic difference, as its lower SS. Considering ASR and SS comprehensively, we conduct follow-up experiments all on vanilla ViBA.

To explore the occurrence frequency of S1 and S2, we present in Table 3 the EASR1 and EASR2. Since many entities can induce both S1 and S2, their sum may exceed 1.0. We find that S1 and S2 are both frequent non-robust phenomena, as high EASR1 and EASR2 suggest, which shows the NER models are fragile to the boundary tokens. Furthermore, a consistently higher EASR2 indicates that the model possesses a comparatively weaker capability in resisting S2 compared to S1.

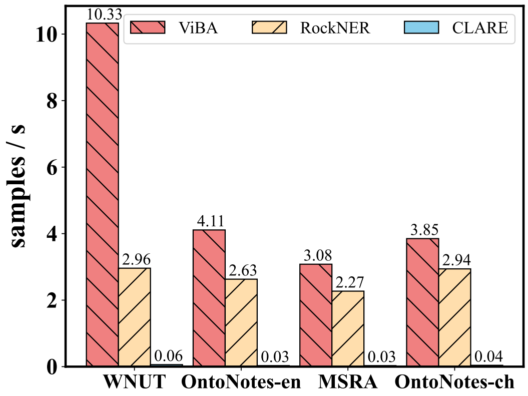

Figure 4: Comparison of attackers' efficiency.

Since ViBA is an attack that operates at the word level, for the sake of a fair comparison, we select other latest and state-of-the-art word-level attackers as baselines. We reproduce the context-level RockNER (Lin et al., 2021) and CLARE (Li et al., 2021) adapted for NER on the four datasets, using RoBERTa-large as the victim model. It is worth noting that our ViBA only replaces context words instead of the entities to avoid the label shift, making it a fair comparison with strong context-level RockNER. When adapting the previous attackers, we keep their algorithms but change the success judgment to whether the predicted tag sequences have changed.

We compare the ASR/SS/ED/FLU of different attackers in Table 4. For ASR, it is displayed that ViBA effectively outperforms the previous attackers. Considering that transplant text attackers may trigger the label shift problem, their actual ASR should be even lower than the reported value. Better SS proves that ViBA preserves more semantic similarity. It is worth mentioning that ViBA is a one-word modification attacker and always maintains the ED to 1.0, which shows that it keeps better syntax than all the other attackers. Both superior SS and ED indicate the ViBA adversarial samples are more imperceptible. The consistently higher fluency also underscores that ViBA's generated adversarial samples are more fluent and natural compared to other attack methods. To further validate that the proposed ViBA can generate more natural and fluent adversarial samples, we also conduct manual evaluation, as demonstrated in appendix A. Overall, ViBA maintains a significant advantage over existing strong baselines.

3.5. Time Analysis

The time complexity for ViBA to attack a sentence of length n is $O(m \times n)$, where m is the amount of the named entities in this sentence. Usually, m is much smaller than n. Thus, the time complexity is

	WNUT	OntoNotes-en	MSRA	OntoNotes-ch
RockNER	64.3/90.3/2.2/53.1	17.1/84.1/2.1/54.7	53.6/94.8/2.3/53.1	72.2/95.0/2.3/47.3
CLARE	55.5/95.4/1.2/54.9	55.0/95.9/1.2/54.1	56.4/94.9/5.9/54.9	40.7/96.3/2.6/63.1
ViBA	**67.1/97.9/1.0/59.8**	**70.0/98.1/1.0/66.6**	**91.7/98.8/1.0/59.8**	**86.9/98.1/1.0/66.2**

Table 4: ASR↑/SS↑/ED↓/FLU↑ comparisons of ViBA and state-of-the-art attackers.

w	1	2	3	4
ASR	74.6	70.0	62.9	55.6

Table 5: The trend of ASR as safety distance w varies, where w=2 is set for all other experiments.

almost linear with n, which makes ViBA efficient. To verify it, we evaluate the number of samples that can be processed by ViBA, RockNER and CLARE within one second on the four datasets, as shown in Figure 4. The victim model is RoBERTa-large.

3.6. Effect of Safety Distance

To investigate the impact of the safety distance w towards ViBA, we conduct comparative experiments by varying it. The experiments are conducted on the OntoNotes-en dataset, with RoBERTa-large chosen as the victim model, as shown in Table 5.

As w increases, the ASR decreases. However, when $w = 1$, there remains a slight label shift issue. But with $w = 2$, this issue is largely mitigated. Additionally, setting $w = 2$ maintains a high ASR, which is why we have chosen this value in our paper.

4. Discussion

This section discusses the effectiveness of ViBA and our motivation through empirical experiments.

4.1. Boundary as Trigger

As mentioned in (Lin et al., 2021), the NER models tend to memorize the entity patterns instead of reasoning them by context, which hints us to explore which tokens play the key role for such entity patterns (i.e., boundary tokens or non-boundary tokens). Thus, we mask out the boundary or non-boundary tokens respectively of an entity to expose which one is more important for entity recognition.

Specifically, we fine-tune two RoBERTa-large models on MSRA and OntoNotes-en datasets. Then we examine the models' dependence on the boundary and inner tokens: (1) For each sentence $\mathcal{X} = \mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$, one of its entities $e = \mathbf{x}_i, \mathbf{x}_{i+1}, \cdots, \mathbf{x}_{i+m}$ is first recognized as type t with the highest probability p_t among all the types. (2) We mask out the boundary tokens $\mathbf{x}_i$ and $\mathbf{x}_{i+m}$ of e in $\mathcal{X}$ respectively to obtain two sentences and

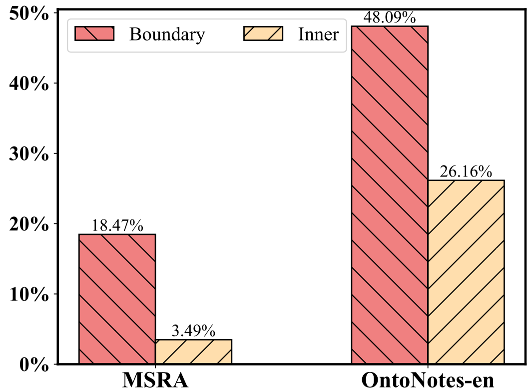

Figure 5: The probability drops caused by masking out boundary and inner tokens.

	OntoNotes-en	OntoNotes-ch
Boundary Tokens	0.95	0.93
Inner Tokens	0.96	0.95

Table 6: The cosine similarity of the hidden-states.

feed them into the model again. The model separately estimates the probabilities $p_t^{'}$ and $p_t^{''}$ that the masked entities remain type t. Since $p_t^{'}$ and $p_t^{''}$ are always less than p_t, we leverage the mean value of two probability drops $p_t - p_t^{'}$, $p_t - p_t^{''}$ to reflect the dependence of the model on boundary. (3) Similarly, we mask out the inner tokens of e and calculate the mean value of probability drops, as shown in Figure 5.

On the Chinese MSRA dataset, the probability drop caused by masking out boundary tokens is more than five times that of masking out inner tokens. On English OntoNotes, masking out boundary tokens even causes a probability drop of nearly 50%. It can be concluded that compared to masking out inner tokens, masking out the boundary tokens will significantly hurt the probability that the model maintains the original prediction, which indicates that the models are more reliance on the boundary of an entity for making final prediction and provides evidence for our intuition to insert boundary which triggers the misclassification.

4.2. Robustness of Encoder and Decoder

The BERT-style NER models can be summarized into an encoder-decoder structure. The encoder

	OntoNotes-en		OntoNotes-ch	
	ASR	*F_1*	*ASR*	*F_1*
FreeLB	70.5	**89.5**	86.0	85.2
ASA	72.2	89.3	86.8	**85.3**
Mixed	68.6	75.4	77.7	84.1
p	*ASR*	*F_1*	*ASR*	*F_1*
0	73.2	89.2	85.5	85.0
0.3	**63.7**	88.8	87.1	84.7
0.5	67.7	88.3	85.4	83.6
0.8	69.8	83.1	**71.5**	63.0

Table 7: Results of masking out the boundary tokens.

usually leverages a strong PrLM, which encodes the input into contextual hidden-states. The decoder is usually an MLP classifier, a conditional random field (CRF), etc and classifies each token into a pre-defined tag based on its hidden-states.

Since the hidden-states are the only medium between the encoder and decoder, we analyze their robustness from the stability of the hidden-states to further interpret ViBA. For each generated adversarial sample $\mathcal{X}$, it is fed into the encoder to obtain its hidden-states $\mathcal{H}$. Then we mask out the original entity in $\mathcal{X}$ to get $\mathcal{X}_m$ and input it into the encoder to obtain hidden-states $\mathcal{H}_m$. We select the representations of the inserted boundary from the $\mathcal{H}, \mathcal{H}_m$ and calculate the cosine similarity between them. Similarly, we also calculate the cosine similarity for all the other tokens in the sentence. We conduct experiments with BERT-base on the OntoNotes dataset. The average values of the cosine similarities are displayed in Table 6.

We figure out that for the inserted boundary tokens, the cosine similarity of the hidden-states between the $\mathcal{H}$ and $\mathcal{H}_m$ exceeds 0.93 in two datasets. It is worth noting that the hidden-states of the BERT-base are as high as 768 dimensions, and the cosine similarity so close to 1 shows that the inserted boundary does not cause a significant deviation in the encoder output. Similar to this phenomenon, other tokens also obtain an average similarity of 0.95 in two datasets, which further verifies that the encoder is relatively stable against $\mathcal{X}$ and $\mathcal{X}_m$. It implies that even if the slight changes of the hidden-states output by the encoder in the position of the inserted boundary can confuse the decoder.

To summary up, (1) The NER models tend to recognize the entities depending on the boundary and perhaps memorize the boundary pattern. (2) The decoder is not robust enough to resist slight perturbation on hidden-states.

	OntoNotes-en		OntoNotes-ch	
	ASR	*F_1*	*ASR*	*F_1*
WP	70.4	88.4	88.4	84.7
p	*ASR*	*F_1*	*ASR*	*F_1*
0	73.2	**89.2**	85.5	85.0
0.3	**70.2**	88.8	85.7	**85.1**
0.5	70.8	88.7	84.7	85.0
0.8	75.1	87.6	**80.4**	84.3

Table 8: Results of boundary dropout to the hidden-states for the decoder and weight perturbation baseline.

5. Defense Strategy: Boundary Cut

This section presents a Boundary Cut strategy that enhances NER robustness against ViBA.

5.1. Decouple Boundary and Inner Words

Since the NER model recognizes the entity relying more on the boundary pattern, a very straightforward idea is to decouple the boundary words and inner words, encouraging the model to capture the pattern of inner words. We achieve this goal by masking out the boundary words at the input. In detail, we randomly mask out the left and right boundary tokens of an entity with a probability p during the fine-tuning phase. In addition, to explore whether masking out the boundary words during training will influence the model on entity recognition, we also report the F_1 on the clean test set, where a higher F_1 indicates a higher recognition performance. We apply BERT-base to conduct experiments on OntoNotes in Table 7.

Compared to the case without masking ($p = 0$), almost all ASR has a significant decrease after masking out the boundary words, suggesting that masking out boundary words is beneficial for resisting ViBA. An exception happens when $p = 0.3$ on OntoNotes-ch. Our explanation for this anomaly is that masking out boundary words can be a trade-off. On the one hand, it reduces the model sensitivity to boundaries, thus decreasing ASR. On the other hand, it will also bring noise, which may lead to insufficient training and make the model vulnerable. In some cases, the latter may outweigh the former. When observing the recognition performance, the F_1 of all experiments slightly decreases as $p = 0.3, 0.5$, which indicates that the noise introduced by masking out the boundary does not cause much performance reduction. It is also not surprising that there is a large drop in F_1 with such big noise when $p = 0.8$. Overall, when p is within a reasonable range, masking out boundary can effectively resist ViBA without significantly reducing the recognition performance. Based on our experiments, $p = 0.5$ works best.

Adversarial Training (AT) is the commonly used method to improve the model's robustness. We select FreeLB (Zhu et al., 2020) and ASA (Wu and Zhao, 2022) as our baselines. Compared to them, though F_1 is relatively lower, our method achieves a significantly advantageous ASR. Also, we re-train the model on the mixture of adversarial and original samples (Mixed), where we set the label of the inserted boundary to "O" in an adversarial sample, and the rest of the tokens are consistent with the original sample. To our surprise, Mixed significantly reduces ASR and does not damage F_1 substantially, especially for the Chinese dataset, which indicates the distinction between generated adversarial samples and the original samples is really slight.

5.2. Dropout Hidden-States

Since the decoder is relatively non-robust to the hidden-states and ViBA mainly fools it, improving its robustness is also a direct idea. We propose to apply dropout (Hinton et al., 2012) on the hidden-states for enhancement. While also considering that the NER model is sensitive to boundary words, we randomly dropout the boundary of an entity on top of the hidden-states with a probability p. We conduct experiments on the OntoNotes dataset. The victim model is BERT-base with a vanilla MLP decoder. We take a classic weight perturbation (WP) method (Wen et al., 2018), which can improve model robustness as the baseline.

In Table 8, ASR drops significantly when $p = 0.5$. Meanwhile, the F_1 on the test set is almost unaffected. ViBA also outperforms WP with a lower ASR and higher F_1. We can conclude that such a concise dropout can help the victim model resist ViBA without affecting its recognition performance. Also, the model is fragile due to the undertraining problem, and it is understandable to have poor ASR and F_1 when $p = 0.8$ on OntoNotes-en.

5.3. Defense Against General Attacks

Since previous experiments have demonstrated that the Boundary Cut strategy can help mitigate Entity Boundary Interference, thus enhancing the model's robustness against ViBA adversarial samples, it prompts us to explore whether this strategy can be extended to other word substitution attacks (Lin et al., 2021; Li et al., 2021). To this end, we verify whether Boundary Cut can assist the model in defending against RockNER as a representative. Specifically, we train the BERT-base models with the defense strategies: Mask out Boundary Words (M), Dropout Hidden-States (D) both with $p = 0.5$. And then we evaluate their F_1 scores on the adversarial samples generated by RockNER.

Model	BERT-base	+M	+D
F_1	65.3	68.7(↑ 3.4)	68.0(↑ 2.7)

Table 9: F_1 scores on adversarial samples generated by RockNER, with +M representing the Masking out of Boundary Words during training and +D indicating the addition of Dropout Hidden-States.

As illustrated in Table 9, our implementation of the Boundary Cut strategy results in a significant enhancement of the victim model's proficiency in accurately identifying entities within the RockNER adversarial samples. This observation underlines the broad adaptability of our Boundary Cut approach when it comes to defending against a variety of different attack techniques.

6. Related Work

In recent years, adversarial samples (Goodfellow et al., 2015) generation has been a popular research area in NLP, mainly focusing on evaluating the robustness of NLP models.

Current studies on robustness concentrate on text classification, question answering (QA), etc. For instance, Gao et al. (2018) propose the DeepWordBug, which effectively fools the models in a black-box scenario. SCPNs (Iyyer et al., 2018) employ syntactic information to generate adversarial samples specifically for text classification tasks. The widely recognized TextFooler (Jin et al., 2020) attacks the BERT-style models and has gained prominence due to its remarkable effectiveness and efficiency. BAE (Garg and Ramakrishnan, 2020) is designed to perform adversarial attacks on text classification tasks and generates adversarial samples through contextual perturbations, making it particularly effective in black-box scenarios. CLARE (Li et al., 2021) is known for its ability to create adversarial samples that exhibit fluency and grammatical coherence by employing a mask-then-infill procedure. Gan and Ng (2019) attacks the question paraphrasing in the QA dataset. Tan et al. (2020b) perturb the inflectional morphology of words to generate plausible and semantically similar adversarial samples. However, despite the numerous works on generating adversarial samples for NLP tasks, they have all overlooked NER.

Recently, some researchers have begun to focus on the robustness of NER models. Mayhew et al. (2020) investigate the influence of capitalization on NER models. Das and Paik (2022) delve into the examination of how perturbations in the surrounding context impact entities. But none of them propose an efficient NER attacker. Nowadays, there are only a few studies that propose attackers for NER systems. While Seqattack (Simoncini and Spanakis, 2021) does adapt some of the previously mentioned attack methods from text classification to NER, it does not introduce a novel approach, and the success rates of these methods are in need of improvement. While there are some rare NER attackers like RockNER (Lin et al., 2021) and Breaking BERT (Dirkson et al., 2021), they essentially introduce a label shift issue and face challenges related to low efficiency and a poor success rate.

7. Conclusion

This paper studies the robustness of current dominant NER models. Due to the label shift problem, existing attackers easily generate invalid adversarial samples. We first reveal a noteworthy problem, the Entity Boundary Interference that is particularly prevalent in NER models. Subsequently, we propose a novel one-word modification attacker *ViBA* that alleviates label shift. Moreover, we interpret the effectiveness of it and further propose a boundary cut strategy that enhances the model's robustness against a variety of word substitution attackers.

Limitations

Typically, the Chinese boundary token is a single character and the English boundary token is a meaningful word. We do not explore how much this distinction affects our attack in depth.

8. Bibliographical References

Bogdan Babych and Anthony Hartley. 2003. Improving machine translation quality with automatic named entity recognition. In *Proceedings of the 7th International EAMT workshop on MT and other language technology tools, Improving MT through other language technology tools, Resource and tools for building MT at EACL 2003*.

Elizabeth Clark, Yangfeng Ji, and Noah A. Smith. 2018. Neural text generation in stories using entity representations as context. 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 2250–2260, New Orleans, Louisiana. Association for Computational Linguistics.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for Chinese natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 657–668, Online. Association for Computational Linguistics.

Sudeshna Das and Jiaul Paik. 2022. Resilience of named entity recognition models under adversarial attack. In *Proceedings of the First Workshop on Dynamic Adversarial Data Collection*, pages 1–6.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Anne Dirkson, Suzan Verberne, and Wessel Kraaij. 2021. Breaking bert: Understanding its vulnerabilities for biomedical named entity recognition through adversarial attack. *ArXiv preprint*, abs/2109.11308.

Wee Chung Gan and Hwee Tou Ng. 2019. Improving the robustness of question answering systems to question paraphrasing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6065–6075, Florence, Italy. Association for Computational Linguistics.

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

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

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *ArXiv preprint*, abs/2006.03654.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *ArXiv preprint*, abs/1207.0580.

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, New Orleans, Louisiana. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8018–8025. AAAI Press.

Gina-Anne Levow. 2006. The third international Chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 108–117, Sydney, Australia. Association for Computational Linguistics.

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2021. Contextualized perturbation for textual adversarial attack. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5053–5069, Online. Association for Computational Linguistics.

Bill Yuchen Lin, Wenyang Gao, Jun Yan, Ryan Moreno, and Xiang Ren. 2021. RockNER: A simple method to create adversarial examples for evaluating the robustness of named entity recognition models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3728–3737, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv preprint*, abs/1907.11692.

Stephen Mayhew, Nitish Gupta, and Dan Roth. 2020. Robust named entity recognition with truecasing pretraining. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8480–8487. AAAI Press.

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

Vassilina Nikoulina, Agnes Sandor, and Marc Dymetman. 2012. Hybrid adaptation of named entity recognition for statistical machine translation. In *Proceedings of the Second Workshop on Applying Machine Learning Techniques to Optimise the Division of Labour in Hybrid MT*, pages 1–16, Mumbai, India. The COLING 2012 Organizing Committee.

Nanyun Peng and Mark Dredze. 2016. Improving named entity recognition for Chinese social media with word segmentation representation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 149–155, Berlin, Germany. Association for Computational Linguistics.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865, Melbourne, Australia. Association for Computational Linguistics.

Avirup Sil and Alexander Yates. 2013. Re-ranking for joint named-entity recognition and linking. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, pages 2369–2374. ACM.

Walter Simoncini and Gerasimos Spanakis. 2021. Seqattack: On adversarial attacks for named entity recognition. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 308–318.

Chuanqi Tan, Wei Qiu, Mosha Chen, Rui Wang, and Fei Huang. 2020a. Boundary enhanced neural span classification for nested named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9016–9023.

Samson Tan, Shafiq Joty, Min-Yen Kan, and Richard Socher. 2020b. It's morphin' time! Combating linguistic discrimination with inflectional perturbations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2920–2935, Online. Association for Computational Linguistics.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. Gpt-ner: Named entity recognition via large language models. *arXiv preprint arXiv:2304.10428*.

R Weischedel, M Palmer, M Marcus, E Hovy, S Pradhan, L Ramshaw, N Xue, A Taylor, J Kaufman, M Franchini, et al. 2013. Ontonotes release 5.0 ldc2013t19. linguistic data consortium, philadelphia, pa (2013).

Yeming Wen, Paul Vicol, Jimmy Ba, Dustin Tran, and Roger B. Grosse. 2018. Flipout: Efficient pseudo-independent weight perturbations on mini-batches. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Hongqiu Wu and Hai Zhao. 2022. Adversarial self-attention for language understanding. *ArXiv preprint*, abs/2206.12608.

Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2023. Empirical study of zero-shot ner with chatgpt. *arXiv preprint arXiv:2310.10035*.

Ming Xu. 2022. Text2vec: Text to vector toolkit. `https://github.com/shibing624/text2vec`.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2020. Freelb: Enhanced adversarial training for natural language understanding. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

A. Manual Evaluation

We employ five participants with a computer science background and five with a humanities background to compare the fluency and naturalness of the adversarial samples generated by different attackers.

Specifically, we select 30 sentences each from OntoNote-en and OntoNote-ch to generate a total of 60 sets of adversarial samples using ViBA, RockNER, and CLARE. Participants are required to score the naturalness and fluency of each set of adversarial samples, where the most fluent and natural samples are rated as 3, followed by 2, and the poorest as 1. The average scores are shown in Table 10.

	OntoNotes-en	OntoNotes-ch
RockNER	1.93	1.83
CLARE	1.83	1.97
ViBA	2.23	2.20

Table 10: The average scores assigned by participants to samples generated by different attackers.

From Table 10, it can be observed that participants are inclined to perceive that ViBA generates more fluent and natural adversarial samples, which aligns with the results presented in Table 4.