Natural Language Adversarial Defense through Synonym Encoding

Xiaosen Wang, Hao Jin, Yichen Yang and Kun He
School of Computer Science and Technology, Huazhong University of Science and Technology

uai2021

Introduction

Currently, synonym substitution based adversarial attacks are widely adopted for generating textual adversarial examples, such as GSA, PWWS and GA. In contrast, there are mainly two type of defense against synonym substitution based attacks:

- **Adversarial Training (AT)** incorporates adversarial examples into training set to enhance the model robustness, but it is time-consuming due to the inefficiency of existing adversary generations in text domain.
- **Interval Bound Propagation (IBP)** provides a provable guarantee that the model is robust to all word substitutions in one sample, but such defenses are hard to be scaled to large datasets and neural networks due to high complexity.

Goal: Proposing a simple yet **effective and efficient** defense method against synonym substitution based adversarial attacks.

Motivation

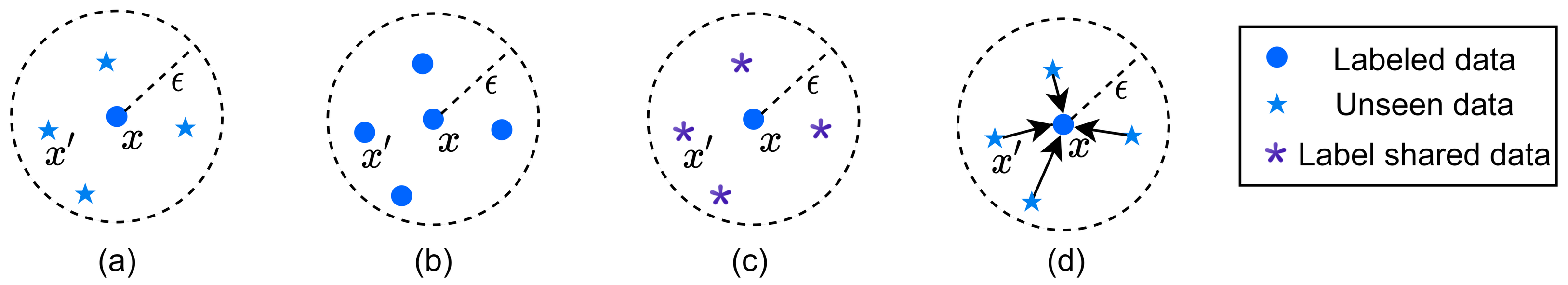


Figure 1: The neighborhood of a data point x in the input space. (a) Normal training: there exists some data point x' that the model has never seen before and yields wrong classification. (b) Adding infinite labeled data: this is an ideal case that the model has seen all possible data points to resist adversaries. (c) Sharing label: all the neighbors share the same label with x. (d) Mapping neighborhood data points: mapping all neighbors to center x so as to eliminate adversarial examples.

Let $\mathcal{X}$ denote the input space and $V_\epsilon(x)$ denote the ϵ-neighborhood of a data point $x \in \mathcal{X}$, where $V_\epsilon(x) = \{x' \in \mathcal{X} | \|x' - x\|_p < \epsilon\}$.

We postulate that the existence of adversarial examples is attributed to the weak generalization of the model. Specifically, for any data point $x \in \mathcal{X}$, $\exists x' \in V_\epsilon(x), f(x') \neq y'_{true}$ and x' is an adversarial example of x. Previous works have tried to adopt infinite labeled data or force the neighbors of a data point x to share the same label with x to improve the robustness but are either impractical or computational inefficient.

In this work, we propose a novel way to find an encoder $E: \mathcal{X} \to \mathcal{X}$ where $\forall x' \in V_\epsilon(x), E(x') = x$. In the context of text classification, the neighbors of x are its synonymous sentences and a reliable way to find synonymous sentences is to substitute words in the original sentence with their close synonyms.

Synonym Encoding Method

To effectively defend the synonym substitution based adversarial attacks, we propose a novel defense method **Synonym Encoding Method (SEM)** which **encodes the synonyms of each word to the same token** and embeds the encoder in front of the input layer of the neural network model using **normal training** to eliminate the word-level perturbations.

Algorithm 1 Synonym Encoding Algorithm

Input: $\mathcal{W}$: dictionary of words
Input: n: size of $\mathcal{W}$
Input: δ: distance for synonyms
Input: k: number of synonyms for each word
Output: E: encoding result

$E = \{w_1 : \text{None}, \dots, w_n : \text{None}\}$
Sort the words dictionary $\mathcal{W}$ by word frequency
for each word $w_i \in \mathcal{W}$ **do**
  if $E[w_i] = \text{NONE}$ **then**
    if $\exists \hat{w}_i^j \in Syn(w_i, \delta, k)$, $E[\hat{w}_i^j] \neq \text{NONE}$ **then**
      $\hat{w}_i^* \leftarrow$ the closest encoded synonym $\hat{w}_i^j \in Syn(w_i, \delta, k)$ to w_i
      $E[w_i] = E[\hat{w}_i^*]$
    else
      $E[w_i] = w_i$
    end if
    for each word $\hat{w}_i^j$ in $Syn(w_i, \delta, k)$ **do**
      if $E[\hat{w}_i^j] = \text{NONE}$ **then**
        $E[\hat{w}_i^j] = E[w_i]$
      end if
    end for
  end if
end for
return E

Experiments

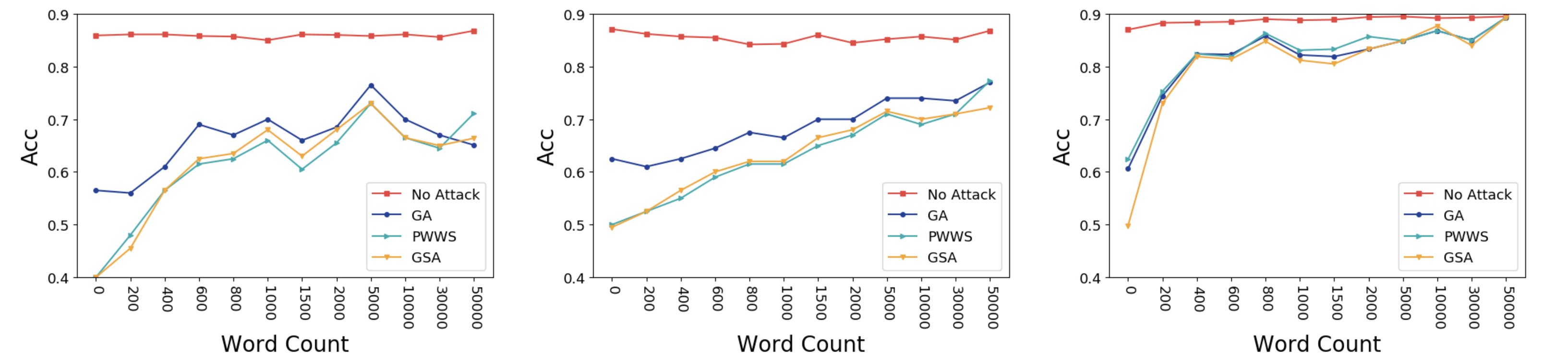
Dataset	Attack	Word-CNN				LSTM				Bi-LSTM				BERT		
		NT	AT	IBP	SEM	NT	AT	IBP	SEM	NT	AT	IBP	SEM	NT	AT	SEM
IMDB	No-attack	88.7	**89.1**	78.6	86.8	87.3	**89.6**	79.5	86.8	88.2	**90.3**	78.2	87.6	92.3	**92.5**	89.5
	GSA	13.3	16.9	**72.5**	66.4	8.3	21.1	70.0	**72.2**	7.9	20.8	**74.5**	73.1	24.5	34.4	**89.3**
	PWWS	4.4	5.3	**72.5**	71.1	2.2	3.6	70.0	**77.3**	1.8	3.2	74.0	**76.1**	40.7	52.2	**89.3**
	GA	7.1	10.7	71.5	**71.8**	2.6	9.0	69.0	**77.0**	1.8	7.2	**72.5**	71.6	40.7	57.4	**89.3**
AG's News	No-attack	**92.3**	92.2	89.4	89.7	92.6	**92.8**	86.3	90.9	**92.5**	**92.5**	89.1	91.4	94.6	**94.7**	94.1
	GSA	45.5	55.5	**86.0**	80.0	35.0	58.5	79.5	**85.5**	40.0	55.5	79.0	**87.5**	66.5	74.0	**88.5**
	PWWS	37.5	52.0	**86.0**	80.5	30.0	56.0	79.5	**86.5**	29.0	53.5	75.5	**87.5**	68.0	78.0	**88.5**
	GA	36.0	48.0	**85.0**	80.5	29.0	54.0	76.5	**85.0**	30.5	49.5	78.0	**87.0**	58.5	71.5	**88.5**
Yahoo! Answers	No-attack	68.4	**69.3**	64.2	65.8	71.6	**71.7**	51.2	69.0	72.3	**72.8**	59.0	70.2	**77.7**	76.5	76.2
	GSA	19.6	20.8	**61.0**	49.4	27.6	30.5	30.0	**48.6**	24.6	30.9	39.5	**53.4**	31.3	41.8	**66.8**
	PWWS	10.3	12.5	**61.0**	52.6	21.1	22.9	30.0	**54.9**	17.3	20.0	40.0	**57.2**	34.3	47.5	**66.8**
	GA	13.7	16.6	**61.0**	59.2	15.8	17.9	30.5	**66.2**	13.0	16.0	38.5	**63.2**	15.7	33.5	**66.4**

Table 1: The classification accuracy (%) of three defense methods under various attacks.

Experiments

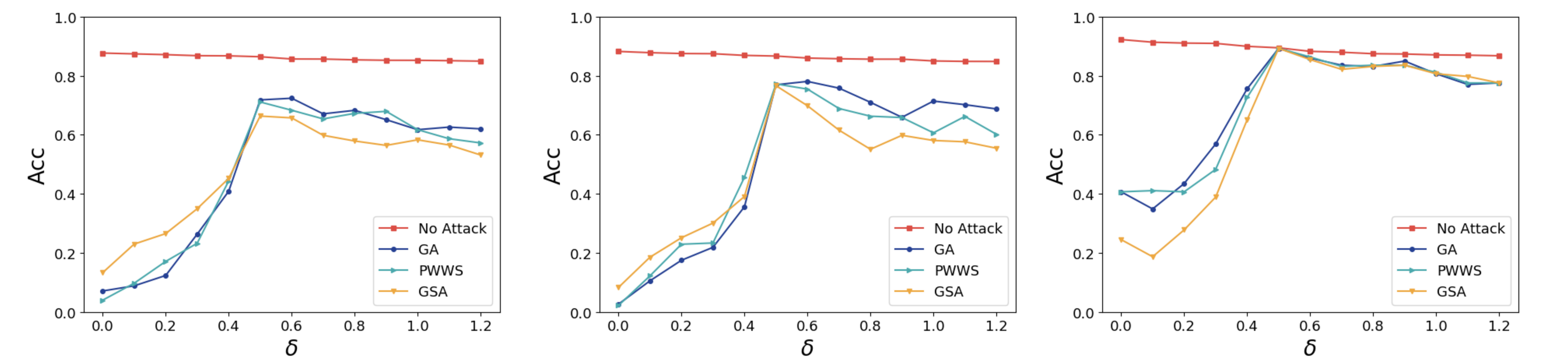
Attack	Word-CNN				LSTM				Bi-LSTM				BERT		
	NT	AT	IBP	SEM	NT	AT	IBP	SEM	NT	AT	IBP	SEM	NT	AT	SEM
GSA	45.5*	86.0	**87.0**	**87.0**	80.0	89.0	83.0	**90.5**	80.0	87.0	87.5	**91.0**	92.5	**94.5**	90.5
PWWS	37.5*	86.5	**87.0**	**87.0**	70.5	87.5	83.0	**90.5**	70.0	87.0	86.5	**90.5**	90.5	**95.0**	90.5
GA	36.0*	85.5	**87.0**	**87.0**	75.5	88.0	83.5	**90.5**	76.0	86.5	86.0	**91.0**	91.5	**95.0**	90.5
GSA	84.5	89.0	**87.5**	87.0	35.0*	87.0	83.5	**90.5**	73.0	85.0	86.5	**91.0**	93.0	**95.5**	90.5
PWWS	83.0	89.0	**87.5**	87.0	30.0*	86.0	85.0	**90.5**	67.5	85.5	86.5	**90.5**	93.0	**95.0**	90.5
GA	84.0	89.5	**87.5**	87.0	29.0*	88.0	83.5	**90.5**	70.5	87.5	87.0	**91.0**	92.5	**95.5**	90.5

Table 2: The classification accuracy (%) of various models for adversarial examples generated through CNN or LSTM model on *AG's News* for evaluating the transferability.



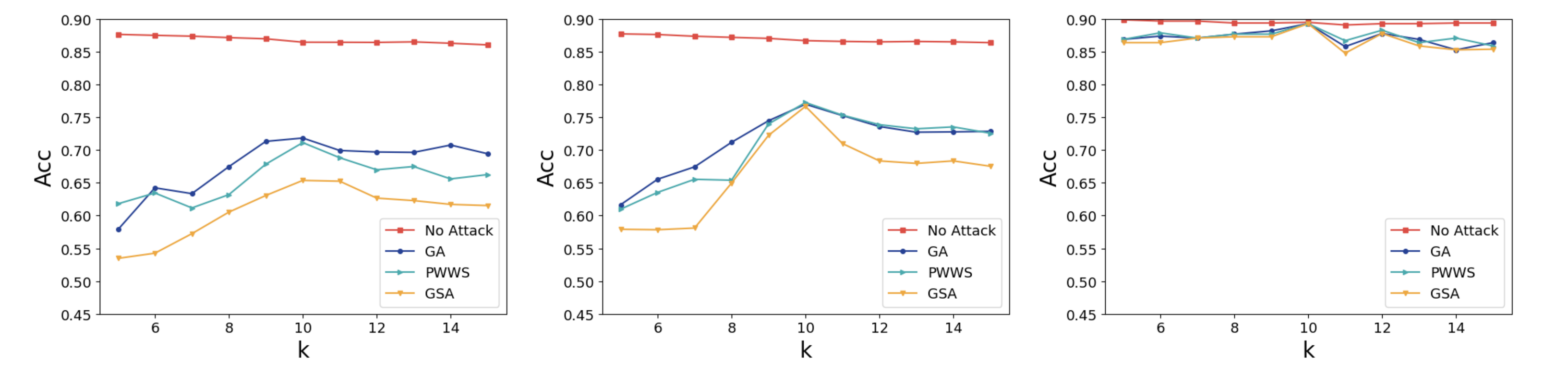
(a) Word-CNN under attacks (b) LSTM under attacks (c) BERT under attacks

Figure 2: The impact of word frequency on the performance of SEM for four models on *IMDB*. We report the classification accuracy (%) of each model with various number of words ordered by word frequency.



(a) Word-CNN under attacks (b) LSTM under attacks (c) BERT under attacks

Figure 3: Classification accuracy (%) of SEM on various δ ranging from 0 to 1.2 on *IMDB*.



(a) Word-CNN under attacks (b) LSTM under attacks (c) BERT under attacks

Figure 4: Classification accuracy (%) of SEM on various k ranging from 5 to 15 on *IMDB*.

Conclusion

We propose a novel defense SEM against synonym substitution based adversarial attacks in the context of text classification.

- **Effective**. Compared with AT and IBP, SEM can remarkably improve model robustness and block the transferability of adversarial examples, while maintaining good classification accuracy on the benign data.
- **Efficient**. Training with SEM is even faster than normal training due to the reduction of encoding space. SEM is also easy to apply to large models and big datasets due to its simplicity.