# Robust Textual Embedding against Word-level Adversarial Attacks

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

38th Conference on Uncertainty in Artificial Intelligence
Eindhoven, Netherlands
August 1-5, 2022
uai2022

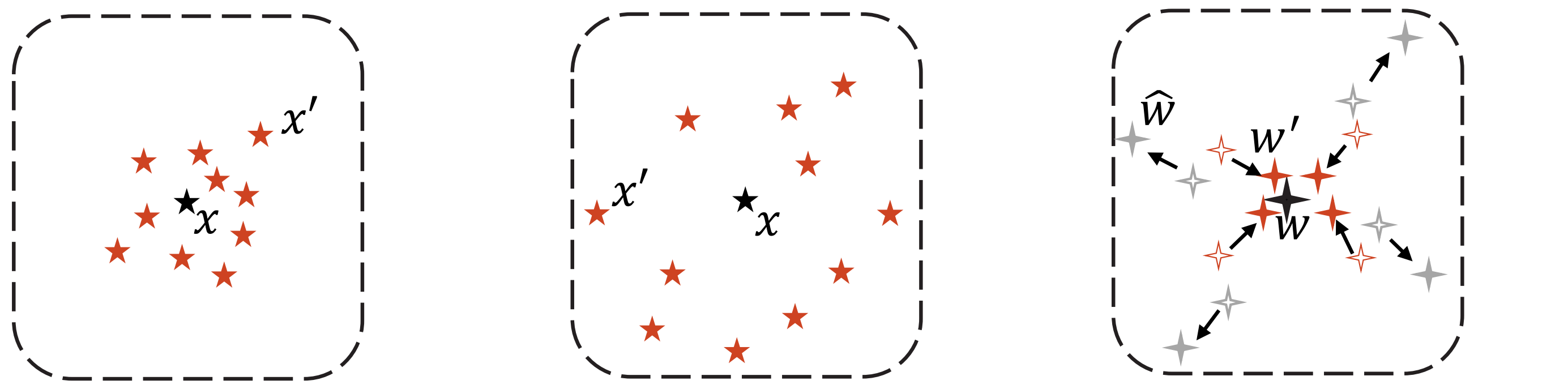
## Background

Deep models for natural language processing (NLP) are vulnerable to adversarial examples crafted by exerting synonym substitutions on the original input. To enhance the model robustness, three categories of adversarial defenses have been proposed.

- **Adversarial Training (AT)**, such as ATFL and ASCC, incorporates adversaries into the training set. But it is time-consuming due to the inefficiency of textual adversary generations.
- **Interval Bound Propagation (IBP)** minimizes the worst-case loss that any combination of the word substitutions can induce. But it is hard to extend to large datasets or large models due to the heavy computing overhead and strict constraints.
- **Synonym Encoding Method (SEM)** maps all synonyms to the same code to eliminate the adversarial examples. It is efficient, but its defense performance is inferior to that of AT.

We propose **an effective and efficient defense method called Fast Triplet Metric Learning (FTML) for robust word embedding.**

## Motivation



(a) Input space (b) Embedding space (c) Word embedding space

**Figure 1:** The motivation of the proposed FTML.

As shown in Fig. 1(a) and (b), we attribute the model's vulnerability to the fact that although the original sample $x$ and its similar samples $x'$ obtained by synonym substitutions are close in the input space, they have dissimilar embedding representations. Those similar samples ultimately leading to wrong predictions are adversarial examples. In contrast, **a robust model should have similar embedding representations when feeding with similar input samples.**

To this end, we adopt the triplet metric learning to force the embedding representations of the original sample and its similar samples to be close. However, the time complexity of exploring similar samples grows exponentially with the text length.

To address this issue, we turn to a word-level solution. As shown in Fig. 1(c), **we pull words $w$ closer to their synonyms $w'$ while pushing non-synonyms $\hat{w}$ further away in the embedding space**, then any input text will have similar embedding representations with its similar samples and distinguish their representations from that of other samples in the dataset.

## Fast Triplet Metric Learning

**Word Distance.** For two words $w_a$ and $w_b$, we use the $\ell_p$-norm distance of their word vectors in the embedding space as their distance:

$$d(w_a, w_b) = \|v(w_a) - v(w_b)\|_p. \qquad (1)$$

**Word-level Triplet Loss.** For a word $w$ in the input text, given its synonym set $\mathcal{S}(w)$ and the non-synonym set $\mathcal{N}$ containing words randomly sampled from the dictionary, we design the word-level triplet loss as:

$$\mathcal{L}_{tr}(w, \mathcal{S}(w), \mathcal{N}) = \frac{1}{|\mathcal{S}(w)|}\sum_{w' \in \mathcal{S}(w)} d(w, w') - \frac{1}{|\mathcal{N}|}\sum_{\tilde{w} \in \mathcal{N}} \min(d(w, \tilde{w}), \alpha) + \alpha, \qquad (2)$$

where the hyper-parameter $\alpha$ is to prevent the distance of negative pairs from keep increasing indefinitely.

**Overall Training Objective.** Given a text $x = \langle w_1, w_2, \cdots, w_n \rangle$ with the ground-truth class label $y$, we incorporate the triplet loss with standard training to train a robust model. The overall training objective is as follows:

$$\mathcal{L}(x, y) = \mathcal{L}_{ce}(f(x), y) + \beta \cdot \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_{tr}(w_i, \mathcal{S}(w_i), \mathcal{N}_i), \qquad (3)$$

where $\mathcal{L}_{ce}(\cdot, \cdot)$ denotes the cross-entropy loss, and $\beta$ is a hyper-parameter to control the weight of the triplet loss.

## Experiments

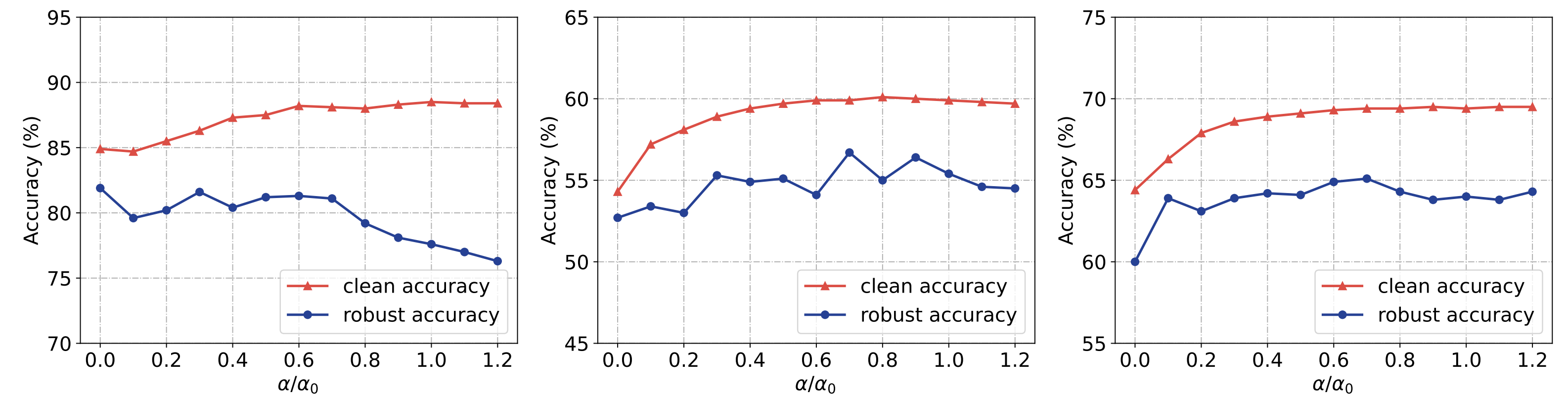
| Dataset | Defense | CNN | | | | | LSTM | | | | | BERT | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Clean | PWWS | GA | PSO | HLA | Clean | PWWS | GA | PSO | HLA | Clean | PWWS | GA | PSO | HLA |
| *IMDB* | Standard | 89.7 | 0.6 | 2.6 | 1.4 | 17.7 | 89.1 | 0.2 | 1.6 | 0.3 | 8.7 | 92.4 | 16.6 | 8.1 | 1.9 | 8.2 |
| | IBP | 81.7 | 75.9 | 76.0 | 75.9 | 76.6 | 77.6 | 67.5 | 67.8 | 67.6 | 68.2 | - | - | - | - | - |
| | ATFL | 85.0 | 63.6 | 66.8 | 64.7 | 72.8 | 85.1 | 72.2 | 75.5 | 74.0 | 77.7 | - | - | - | - | - |
| | SEM | 87.6 | 62.2 | 63.5 | 61.5 | 70.5 | 86.8 | 61.9 | 63.7 | 62.2 | 70.8 | 89.9 | 72.3 | 70.5 | 69.2 | 75.2 |
| | ASCC | 84.8 | 74.0 | 75.5 | 74.5 | 77.6 | 84.3 | 74.2 | 76.8 | 75.5 | 79.5 | 81.3 | 65.1 | 65.4 | 63.1 | 69.5 |
| | FTML | 88.1 | **81.1** | **81.4** | **81.1** | **82.4** | 87.2 | **79.0** | **79.2** | **78.8** | **79.7** | 91.3 | **81.2** | **81.5** | **80.0** | **83.1** |
| | | | ↑5.2 | ↑5.4 | ↑5.2 | ↑4.8 | | ↑4.8 | ↑2.4 | ↑3.3 | ↑0.2 | | ↑8.9 | ↑11.0 | ↑10.8 | ↑7.9 |
| *Yelp-5* | Standard | 62.7 | 1.1 | 1.3 | 0.8 | 1.1 | 64.8 | 0.5 | 0.9 | 0.4 | 0.5 | 65.7 | 2.4 | 1.1 | 0.7 | 1.3 |
| | IBP | 52.1 | 47.8 | 47.8 | 47.7 | 47.6 | 42.6 | 42.1 | 40.7 | 40.6 | 40.2 | - | - | - | - | - |
| | ATFL | 61.4 | 50.0 | 51.7 | 50.2 | 53.9 | 62.4 | 48.0 | 48.8 | 46.9 | 51.9 | - | - | - | - | - |
| | SEM | 60.1 | 34.9 | 33.8 | 32.4 | 37.2 | 61.9 | 35.5 | 34.3 | 33.7 | 37.6 | 63.7 | 39.9 | 37.0 | 36.9 | 39.4 |
| | ASCC | 58.9 | 47.3 | 49.3 | 47.4 | 50.6 | 59.9 | 48.5 | 50.5 | 49.6 | 52.5 | 63.4 | 50.0 | 50.8 | 49.5 | 54.8 |
| | FTML | 59.9 | **56.7** | **56.7** | **56.6** | **56.5** | 61.9 | **57.5** | **57.6** | **57.5** | **57.6** | 63.0 | **55.4** | **55.1** | **55.0** | **55.2** |
| | | | ↑6.7 | ↑5.0 | ↑6.4 | ↑2.6 | | ↑9.0 | ↑7.1 | ↑7.9 | ↑5.1 | | ↑5.4 | ↑4.3 | ↑5.5 | ↑0.4 |
| *Yahoo! Answers* | Standard | 72.6 | 6.8 | 7.2 | 4.9 | 7.0 | 74.7 | 12.2 | 9.6 | 6.5 | 10.4 | 77.0 | 20.7 | 10.3 | 7.3 | 10.0 |
| | IBP | 63.1 | 54.9 | 54.9 | 54.8 | 55.0 | 54.3 | 47.3 | 47.6 | 47.0 | 47.3 | - | - | - | - | - |
| | ATFL | 72.5 | 62.5 | 63.1 | 62.5 | **65.0** | 73.6 | 61.7 | 60.8 | 60.3 | 63.1 | - | - | - | - | - |
| | SEM | 70.1 | 53.8 | 52.4 | 51.9 | 54.6 | 72.3 | 57.0 | 56.1 | 55.4 | 56.8 | 75.6 | 64.6 | 62.0 | 61.9 | 63.6 |
| | ASCC | 69.0 | 58.4 | 59.6 | 58.5 | 59.9 | 70.7 | 61.7 | 62.3 | 61.9 | 63.2 | 75.2 | 66.4 | 67.5 | 66.6 | 68.0 |
| | FTML | 69.4 | **65.1** | **65.1** | **65.0** | 64.9 | 71.4 | **67.8** | **67.8** | **67.8** | **67.9** | 74.8 | **70.0** | **70.0** | **70.0** | **70.0** |
| | | | ↑2.6 | ↑2.0 | ↑2.5 | ↓0.1 | | ↑6.1 | ↑5.5 | ↑5.9 | ↑4.7 | | ↑3.6 | ↑2.5 | ↑3.4 | ↑2.0 |

**Table 1:** The classification accuracy (%) of models with FTML and defense baselines against various adversarial attacks on three datasets for CNN, LSTM and BERT models.

## Experiments

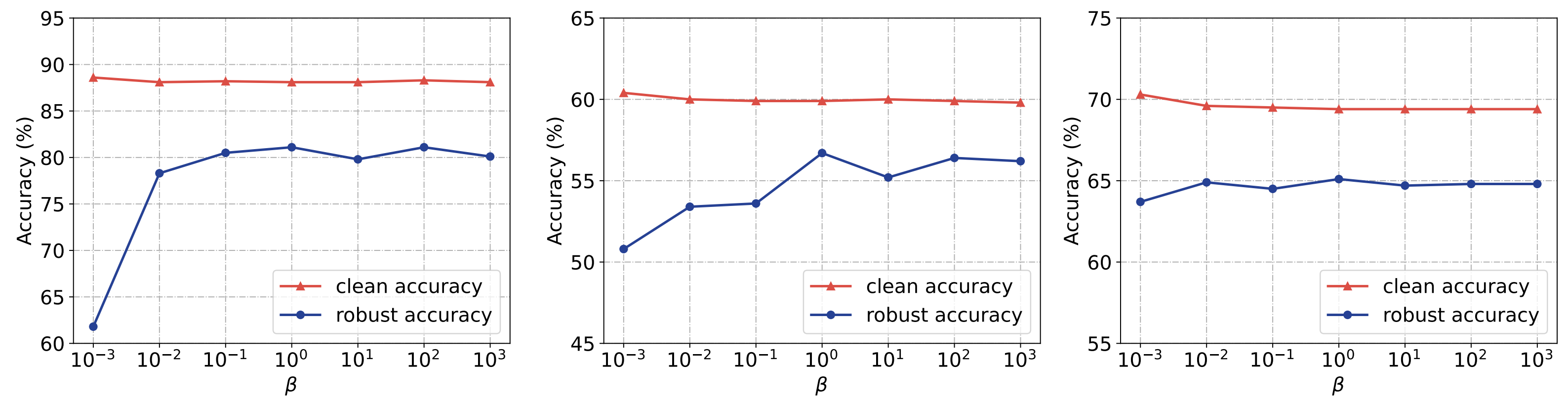
| Defense | *IMDB* | | | *Yelp-5* | | | *Yahoo! Answers* | | |
|---|---|---|---|---|---|---|---|---|---|
| | CNN | LSTM | BERT | CNN | LSTM | BERT | CNN | LSTM | BERT |
| Standard | 1 | 1 | 11 | 4 | 6 | 178 | 9 | 13 | 371 |
| IBP | 1 | 48 | - | 12 | 610 | - | 26 | 953 | - |
| ATFL | 15 | 22 | - | 203 | 290 | - | 444 | 573 | - |
| SEM | 1 | 1 | 11 | 4 | 6 | 178 | 9 | 16 | 371 |
| ASCC | 2 | 6 | 100 | 32 | 71 | 1638 | 55 | 125 | 2523 |
| FTML | 1 | 1 | 12 | 12 | 16 | 183 | 22 | 29 | 384 |

**Table 2:** The training time per epoch (in minutes) for the models with various defenses.



(a) *IMDB* (b) *Yelp-5* (c) *Yahoo! Answers*

**Figure 2:** The impact of hyper-parameter $\alpha$ on the performance of FTML on CNN models against PWWS attack.



(a) *IMDB* (b) *Yelp-5* (c) *Yahoo! Answers*

**Figure 3:** The impact of hyper-parameter $\beta$ on the performance of FTML on CNN models against PWWS attack.

## Conclusion

We propose a new textual adversarial defense approach that focuses on robust word embedding.

- **Generality.** FTML conveys a general idea to learn a robust word embedding by pulling words closer to their synonyms while pushing non-synonyms further away in the embedding space, which is generic to any NLP models and languages.
- **Effectiveness.** Compared to defense baselines, **FTML could significantly promote the model robustness against various advanced adversarial attacks while keeping high clean accuracy.**
- **Efficiency. FTML introduces only a little overhead to the standard training for adjusting the word embedding, facilitating its application to large-scale datasets and complex models.**
- **Impact.** FTML reveals a new perspective of word embedding on enhancing the NLP model robustness, highlighting the difference of model robustness on texts and images.