Using Random Perturbations to Mitigate Adversarial Attacks on NLP Models

Abigail Swenor
University of Colorado Colorado Springs
1420 Austin Bluffs Pkwy
Colorado Springs, CO 80918
aswenor@uccs.edu

Abstract

Deep learning models have excelled in solving many problems in Natural Language Processing, but are susceptible to extensive vulnerabilities. We offer a solution to this vulnerability by using random perturbations such as spelling correction, synonym substitution, or dropping the word. These perturbations are applied to random words in random sentences to defend NLP models against adversarial attacks. Our defense methods are successful in returning attacked models to their original accuracy within statistical significance.

Introduction

Deep learning models have excelled in solving difficult problems in many machine learning tasks, including Natural Language Processing (NLP) (Zhang, Zhao, and LeCun 2015; Kim 2014). However, research has discovered that inputs can be modified to cause trained deep learning models to produce incorrect results and predictions (Szegedy et al. 2014). These perturbations are caused by adversarial attacks, with some specific to NLP models (Gao et al. 2018). Although most NLP adversarial attacks are easily detectable, some new forms of adversarial attacks have become more difficult to detect (Wallace et al. 2021; Chen et al. 2020), revealing new vulnerabilities in NLP models. Considering the increasing difficulty in detecting attacks, a more prudent approach would be to work on neutralizing the effect of potential attacks rather than solely relying on detection. The work summarized here is a novel and highly effective defense solution that preprocesses inputs by random perturbations to mitigate potential hard-to-detect attacks.

Related Work

The work detailed in this paper relates to the attack on NLP models using the TextAttack library (Morris et al. 2020) and the use of randomness against adversarial attacks. The TextAttack library and associated GitHub repository represent current efforts to centralize attack and data augmentation methods for the NLP community. The library allows researchers to better understand the state-of-the-art attack models and to create new kinds of attacks to test the robustness of NLP models. The work summarized here utilizes the ready-to-use classification attacks from the TextAttack library (Morris et al. 2020) to test our defense methods.

Positive impact of randomness in classification tasks with featured datasets can be found in work using Random Forests (Breiman 2001). Random Forests have been useful in many domains to make predictions. Randomness has been deployed in computer vision defense methods against adversarial attacks. Levine and Feizi (2020) use random ablations to defend against adversarial attacks on computer vision classification models. Levine and Feizi defend against sparse adversarial attacks that perturb a small number of features in the input images. They found their random ablation defense method to produce certifiably robust results on the MNIST, CIFAR-10, and ImageNet datasets. We apply similar methods from this computer vision application to NLP models for our defense methods.

Research Questions

We attempt to answer the following questions with this work:

1. Is there a way to protect NLP models against adversarial attacks without relying on detection?
2. Is randomness an effective tool in the protection of NLP models against adversarial attacks?

Methods

The easy availability of successful adversarial attack methods necessitates defense methods that do not rely on detection and leverage intuitions gathered from popular attack methods to protect NLP models. In particular, we present a simple but highly effective defense for deep learning models that perform sentiment analysis. We created two algorithms that utilize randomness to neutralize the effects of adversarial attacks. We use three kinds of random perturbations in our algorithms: spell correction if necessary, substitution by random synonym, or simply dropping the word. These perturbations are applied to random words in a given input with the hope that they will negate the effect that attacking perturbations have on the overall sentiment analysis of an input.

Our first algorithm, Random Perturbations Defense (RPD), is based on the randomization of perturbations of the sentences of a review R followed by majority voting to decide the final prediction for sentiment analysis. We

consider each review R to be represented as a set $R = r_1, r_2, ..., r_i, ..., r_N$ of sentences. We consider each sentence r_i and create l replicates of the sentence $\hat{r}_{ij}$, and insert k perturbations in each replicate. Each perturbation is picked randomly. After lk replicates are made for each sentence r_i in R for a total of lkN perturbed sentences, we create an array of intermediate predictions after passing each replicate $\hat{r}_{ij}$ through a classifier $f()$. We perform majority voting on the intermediate predictions to come to a final prediction of the sentiment of R.

Our second algorithm, Increased Randomness Defense (IRD), contains three random processes. We randomly choose a sentence r_i from R and perform a random perturbation on this sentence to create a replicate $\hat{r}_j$ which is placed in a new set $\hat{R}$. We repeat this random selection of sentences, with replacement, until we reach K number of replicates $\hat{r}_j$ in $\hat{R}$. The rest of this algorithm is similar to our first algorithm. We pass the replicates through our classifier $f()$ to get intermediate predictions, which then go through majority voting to determine the final prediction for R.

We performed experiments on these two defense methods using seven attacks from the TextAttack library (Morris et al. 2020). We use the IMDB dataset (Maas et al. 2011) for our input. Each attack was used to create perturbed reviews from the dataset with a mix of positive and negative sentiments. As our classifier $f()$, we used the HuggingFace library transformer pipeline for sentiment-analysis (Wolf et al. 2020). These attacks were chosen from the 14 classification model attacks available on the library because they represent different kinds of attack methods, including misspelling, synonym substitution, and antonym substitution.

The original accuracy of the HuggingFace transformer pipeline used as our classifier $f()$ was 80% without defense and without being under attack. The model under the seven different attacks ranged in accuracy from 0% to 44%. After applying each of our defense methods individually, both of them were able to return the model within statistical significance of the original accuracy for six of the attacks. The full results can be seen in Table 1.

Attack	w/o Defense	w/ RPD	w/ IRD
BAE	33%	80.80%±1.47	78.40%±3.14
DeepWordBug	34%	76.60%±1.85	76.80%±2.64
FasterGeneticAlgo	44%	82.20%±1.72	82.80%±2.48
Kuleshov*	0%	60.00%±2.24	66.23%±4.65
PWWS	0%	81.80%±1.17	79.20%±1.72
TextBugger	6%	79.20%±2.32	77.00%±2.97
TextFooler	1%	83.20%±2.48	80.20%±2.48

Table 1: Accuracy for each of the attack methods under attack, and under attack with our two defense methods deployed. The accuracy prior to attack is 80%.

The programs and data used in this paper can be found at: https://github.com/aswenor/rand-perturbations-defense.

Significance

The work detailed in this paper is significant to the usage of NLP models that continue to have vulnerabilities exposed and exploited by adversarial attacks. Our defense method offers a unique solution by offering protection to NLP models without relying on detection of an attack, thus increasingly relevant as hard-to-detect attacks are created and used. Our method also is novel because it neutralizes the effects of an attack by using random perturbations. Our method is also generalized to multiple attack methods.

Future Work

While this work was highly effective in mitigating NLP models against adversarial attacks, the theoretical and mathematical approach taken in this work can be further developed to establish provability as well as efficiency of the proposed defenses. Our work also relates specifically to the sentiment analysis task which is a form of binary classification. We would like to expand this work to multi-class classification. We have just begun realizing the effectiveness of the use of randomness in the defense of NLP models against adversarial attacks. We would also like to explore the appropriateness of randomness-based defenses in distributed and federated learning.

References

Breiman, L. 2001. Random forests. *Machine Learning*, 45(1): 5–32.

Chen, X.; Salem, A.; Backes, M.; Ma, S.; and Zhang, Y. 2020. Badnl: Backdoor attacks against NLP models. *arXiv preprint arXiv:2006.01043*.

Gao, J.; Lanchantin, J.; Soffa, M. L.; and Qi, Y. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*, 50–56.

Kim, Y. 2014. Convolutional Neural Networks for Sentence Classification. In *EMNLP*, 1746–1751.

Levine, A.; and Feizi, S. 2020. Robustness certificates for sparse adversarial attacks by randomized ablation. In *AAAI*, 4585–4593.

Maas, A. L.; Daly, R. E.; Pham, P. T.; Huang, D.; Ng, A. Y.; and Potts, C. 2011. Learning Word Vectors for Sentiment Analysis. In *NAACL-HLT*, 142–150.

Morris, J.; Lifland, E.; Yoo, J. Y.; Grigsby, J.; Jin, D.; and Qi, Y. 2020. TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP. In *EMNLP: System Demonstrations*, 119–126.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2014. Intriguing properties of neural networks. In *ICLR 2014*.

Wallace, E.; Zhao, T.; Feng, S.; and Singh, S. 2021. Concealed Data Poisoning Attacks on NLP Models. In *NAACL-HLT*, 139–150.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; et al. 2020. Transformers: State-of-the-Art Natural Language Processing. In *EMNLP: System Demonstrations*, 38–45.

Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level convolutional networks for text classification. *NIPS*, 28: 649–657.