Certified Robustness Against Natural Language Attacks by Causal Intervention

Haiteng Zhao [1 *] **Chang Ma** [1 *] **Xinshuai Dong** [2 *] **Anh Tuan Luu** [3 4] **Zhi-Hong Deng** [1] **Hanwang Zhang** [3]

Abstract

Deep learning models have achieved great success in many fields, yet they are vulnerable to adversarial examples. This paper follows a causal perspective to look into the adversarial vulnerability and proposes Causal Intervention by Semantic Smoothing (CISS), a novel framework towards robustness against natural language attacks. Instead of merely fitting observational data, CISS learns causal effects $p(y|do(x))$ by smoothing in the latent semantic space to make robust predictions, which scales to deep architectures and avoids tedious construction of noise customized for specific attacks. CISS is provably robust against word substitution attacks, as well as empirically robust even when perturbations are strengthened by unknown attack algorithms. For example, on YELP, CISS surpasses the runner-up by 6.8% in terms of certified robustness against word substitutions, and achieves 80.7% empirical robustness when syntactic attacks are integrated.

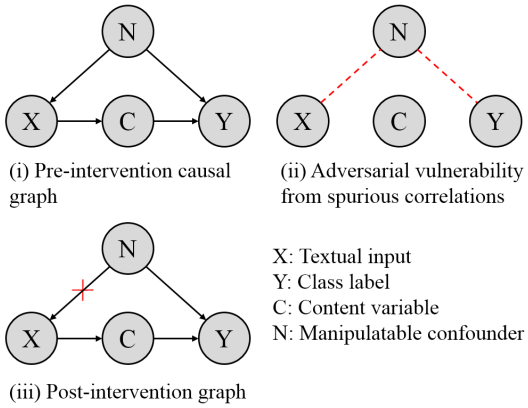

Figure 1. N is a set of variables that can be manipulated under attacks, e.g., style variables. (i): The proposed causal graph of data. (ii): The red dashed lines represent the spurious correlation between X and Y, and it is the source of vulnerability (if a model only fits the observational distribution $p(y|x)$). (iii): The causal graph after removing the arrow from N to X. The post-intervention conditional distribution $p^{N \nrightarrow X}(y|x)$ is robust as it does not rely on any spurious correlation.

1. Introduction

Deep learning models have achieved great success in many fields such as computer vision, natural language processing, and speech recognition (Goodfellow et al., 2016; Krizhevsky et al., 2012; Ren et al., 2015; Sutskever et al., 2014; Hinton et al., 2012). However, they are known to be vulnerable to adversarial examples (Szegedy et al., 2013; Goodfellow et al., 2015; Jia & Liang, 2017), *e.g.*, a BERT-based sentiment analysis model can be easily fooled by synonym substitution attacks (Alzantot et al., 2018), and thus raise severe security challenges to modern NLP systems.

In contrast to deep models, humans' cognitive systems are robust against adversarial perturbations (Szegedy et al., 2013; Goodfellow et al., 2015), as humans can perform causal reasoning (Pearl, 2009; Peters et al., 2017) and are more sensitive to causal relations than statistical correlations (Gopnik et al., 2004). To enable machine learning models with such abilities to predict robustly, it is therefore crucial to understand the adversarial vulnerability from a causal perspective (Zhang et al., 2021; Tang et al., 2021).

In this paper, we consider the source of adversarial vulnerability as the spurious correlations by the confounder effect. We give an illustration in Fig. 1, where X denotes the textual input, Y is the class label, C represents the content of X, and N is a set of confounder variables that can be manipulated under attacks. For example, on IMDB dataset (Maas et al., 2011), a professional reviewer may tend to use many jargons (N) in a movie review (X), while likely to be strict and give negative comments (Y). Such correlations are useful under the i.i.d. setting but harmful if the confounder is manipulatable under attacks; *e.g.*, adding more jargons to a positive movie review might fool a machine model.

From that causal perspective, we draw a link between causal

[*]Equal contribution [1]Peking University [2]Carnegie Mellon University [3]Nanyang Technological University [4]Corresponding Author. Correspondence to: Anh Tuan Luu <anhtuan.luu@ntu.edu.sg>.

1[st] Workshop on Formal Verification of Machine Learning, Baltimore, Maryland, USA. Colocated with ICML 2022. Copyright 2022 by the author(s).

intervention and randomizd smoothing (Lecuyer et al., 2019; Cohen et al., 2019). Randomized smoothing is a promising technique towards certified robustness. It adds random Gaussian noise to the input of a base classifier and predicts by taking the expectation over the noise; the resulting smoothed classifier is robust against l_2-bounded perturbations with certification. We found that a randomized classifier actually models the causal effect $p(y|do(x)) = \int p(y|x,n)p(n)\,dn$ (Pearl, 2009), where $p(y|x,n)$ is the base classifier and $p(n)$ follows a Gaussian distribution. This justifies randomized smoothing from a causal point of view and informs us how to design and improve randomized smoothing towards robustness in certain scenarios.

Following this line of thought, we propose a novel framework towards robustness against natural language attacks, Causal Intervention by Semantic Smoothing (CISS) (illustrated by Figure 2). CISS models interventional distribution $p(y|do(x))$ to predict robustly, by removing the confounder effect in the semantic space. CISS has the following merits: (i) It has clear causal interpretation and causality-guided learning objectives; (ii) It provides certified NLP robustness that scales to deep architectures like Transformers (Vaswani et al., 2017), while other robust certification methods like interval bound propagation (Jia et al., 2019) cannot; (iii) It smoothes in the latent semantic space, which frees us from tedious construction of noise distributions customized for specific attacks; (iv) In addition to certified robustness against seen attacks, CISS is empirically robust even when the perturbations are strengthened by unseen attacks.

We validate our merits by extensive experiments considering both seen word substitution attacks (Jia et al., 2019; Dong et al., 2021a) and unseen syntactic-trigger-based (Qi et al., 2021) and editing distance-based (Levenshtein et al., 1966; Liang et al., 2018) attacks. For example, on IMDB, CISS achieves 76.5% certified robust accuracy against adversarial word substitutions, surpassing the runner-up by 7.2%; on YELP, CISS achieves 83.1% empirical robustness against integrated attacks, surpassing the runner-up by 7.8%.

We summarize the contributions of this paper as follows:

- We propose a causal view to look into robustness: adversarial vulnerability comes from the confounding effect manipulatable by attacks, and randomized classifiers model the causal effect only and thus are robust to such manipulations.
- We propose a novel framework, CISS, to achieve robustness against natural language attacks. It learns causal effects $p(y|do(x))$ to predict robustly by smoothing in the latent semantic space.
- We validate that CISS is certifiably robust against known attacks and empirically robust against integrated attacks by experiments, where CISS consistently surpasses the runner-up with significant margins.

2. Preliminaries

2.1. Notations and Problem Setting

We suppose random variables $X, Y \sim p_{XY}(x, y)$, where $X \in \mathcal{X}$ represents the textual input, $Y \in \mathcal{Y}$ represents the class label, p_{XY} is the data distribution (we will use p in the rest of this paper for notation simplicity), and x, y are the observed values. We are interested in a classifier $q(y|x)$ that is robust against adversarial examples. Given a data point x, an adversarial example of it, $\hat{x} \in \mathbb{B}_{\text{adv}}(x)$, aims to fool a classifier, while $\mathbb{B}_{\text{adv}}(x)$ is defined as a neighbourhood near x to make sure that that $\hat{x}$ shares the same label with x from a human's perspective.

This paper focuses on robustness against natural language attacks, which can be categorized into char-level modifications (Belinkov & Bisk, 2018; Gao et al., 2018; Eger et al., 2019), word-level substitutions (Alzantot et al., 2018; Ren et al., 2019; Dong et al., 2021a), and sentence-level manipulations (Liang et al., 2018; Jia & Liang, 2017; Iyyer et al., 2018). *E.g.*, under adversarial word substitutions, $\mathbb{B}_{\text{adv}}(x)$ is defined as $\mathbb{B}_{\text{adv}}(x) = \{\hat{x} : \hat{x}^i \in \mathbb{S}_{\text{adv}}(x^i)\}$, where x^i is the i^{th} word of x and $\mathbb{S}_{\text{adv}}(x^i)$ is a pre-defined set consisting of semantically similar words of x^i. A classifier q is said to be empirically robust at a (x, y), if it predicts correctly given some $\hat{x}$ that are maliciously generated by some attack algorithms, while a classifier q is certifiably robust, if it can be theoretically guaranteed that q predicts correctly given any $\hat{x}$, *i.e.*, $\arg\max_{y'} q(y'|\hat{x}) = y,\ \forall\, \hat{x} \in \mathbb{B}_{\text{adv}}(x)$.

3. Methodology

3.1. Robustness from A Causal View

In causal inference (Pearl, 2009; Peters et al., 2017), observed data follow a generation process. This process is depicted by a set of structural equation models (SEMs) (Aldrich, 1989; Hoover, 2008; Pearl, 2009; Peters et al., 2017), with a corresponding causal graph. To understand the source of adversarial vulnerability, we propose a causal graph for text classification tasks, shown in Fig. 1 (i). As demonstrated, X and Y denote the textual input and the class label respectively, C represents the content of X, and N is a set of confounder variables that do not directly change the semantic content of the input, *e.g.*, styles.

Given the proposed causal graph, we are able to identify the vulnerability of a machine learning model: there exist spurious correlations $X \leftarrow N \rightarrow Y$, which are established by confounders N, and such non-causal correlations can be exploited by attacks to fool a model. First, these spurious correlations are useful under the i.i.d. setting. As shown by the red dashed lines in Fig. 1 (ii), the path $X \leftarrow N \rightarrow Y$ can be used to predict Y. *E.g.*, professional reviewers tend to use more jargons (N) in their review (X) and they tend to

give strict negative comments (Y), and thus chances are high that a review is negative if there exist many jargons. Second, a model which only fits the observational distribution $p(y|x)$ tends to learn such features for predictions, not only because such non-causal features are useful to minimize the training loss, but also because some of these non-causal features are easier to extract compared to causal relations (Ilyas et al., 2019; Geirhos et al., 2020; Tang et al., 2021). Unfortunately, when the confounder N is manipulatable by attack algorithms, such spurious correlations can become useless or even harmful. *E.g.*, if we employ word substitution attacks to substitute the words of a positive movie review by more jargons, a model which relies on jargons to predict is likely to give a wrong prediction.

To learn a robust model, we therefore need to learn the causal effect from X to Y for prediction, instead of fitting the observational distribution $p(y|x)$. This can be achieved by causal interventions (Pearl, 2009; Peters et al., 2017). Consider a world that follows the same SEMs as Fig. 1 (i), but the arrow from N to X is removed by intervention; the data from this post-intervention world follows $p^{N\nrightarrow X}(x, y, n, c)$, whose causal graph is shown in Fig. 1 (iii). According to Backdoor Adjustment (Pearl, 2009), we have the following theorem (the proof of which is in the Appendix), by which the causal effect $p(y|do(x))$ equals $p^{N\nrightarrow X}(y|x)$, and $p^{N\nrightarrow X}(y|x)$ can be calculated by observational information from $p(x, y, n, c)$.

Theorem 1. *(Backdoor Adjustment (Pearl, 2009)) Given $p(x, y, n, c)$ and $p^{N\nrightarrow X}(x, y, n, c)$, we have:*

$$p(y|do(x)) = p^{N\nrightarrow X}(y|x) = \int p(y|x, n)p(n)\, dn. \quad (1)$$

This theorem informs us that to learn a robust classifier against manipulations on N, we need to model $p(y|do(x))$ by modeling $p(y|x, n)$ and $p(n)$. We next show this objective aligns with randomized smoothing techniques (Lecuyer et al., 2019; Cohen et al., 2019) towards robustness.

3.2. Randomized Classifier Captures the Causal Effect

Let us implement $p(y|x, n)$ by a base classifier $f_y(x + n)$, and assume $p(n)$ follows $N(0, \sigma^2 I)$. For l_2-bounded adversarial perturbations, we have the following theorem:

Theorem 2. *(Soft Smoothed Classifier (Zhai et al., 2020)) Given a base classifier $f(x + n)$ and $p(n) \sim N(0, \sigma^2 I)$. If*

$$\mathop{\mathbb{E}}_{n\sim p(n)}[f_y(x + n)] \geq \max_{y'\neq y} \mathop{\mathbb{E}}_{n\sim p(n)}[f_{y'}(x + n)], \quad (2)$$

then classification by $\arg\max_{y'} \mathbb{E}_{n\sim p(n)}[f_{y'}(\hat{x} + n)]$ is robust for all $\hat{x}$, s.t., $\|x - \hat{x}\|_2 \leq R$, where $R = \frac{\sigma}{2}(\Phi^{-1}(\mathbb{E}_{n\sim p(n)}[f_y(x + n)]) - \Phi^{-1}(\max_{y'\neq y} \mathbb{E}_{n\sim p(n)}[f_{y'}(x + n)]))$, and Φ^{-1} is the inverse of standard Gaussian c.d.f.

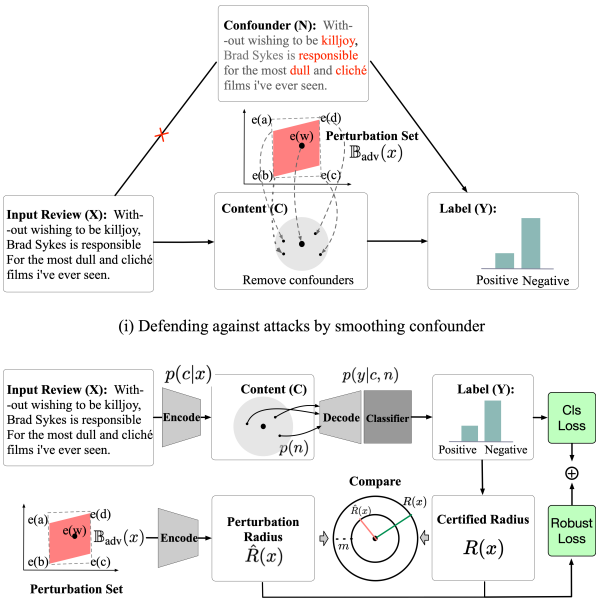

Figure 2. Overview of the proposed framework. (i) Noise added in the latent space can remove the confounder effects manipulated by potential attackers. (ii) An illustration of the training process of CISS.

As in Theorem 2, a randomized classifier adds Gaussian noise n to smooth the input of a base classifier f and predicts the final label by taking the expectation over n. If $f_y(x+n)$ mimics $p(y|x, n)$ well, then the resulting smoothed classifier $\mathbb{E}_{n\sim p(n)}[f_y(x + n)]$ is actually modeling the causal effect $= p(y|do(x)) = \int p(y|x, n)p(n)\, dn$ defined in Theorem 1.

3.3. Modeling Causal Effect by Semantic Smoothing

As shown in Section 3.2, the randomized smoothing technique can be used to model the interventional distribution $p(y|do(x))$; *i.e.*, by assuming $p(n)$ as gaussian, the smoothed classifier $\mathbb{E}_{n\sim p(n)}[f_y(x + n)]$ captures the causal effect and is robust against l_2-bounded attacks. However, it cannot directly transfer to a NLP scenario, as NLP perturbations are neither continuous nor l_2 bounded. As such, we propose a novel framework, CISS, to model the causal effect by smoothing in the latent semantic space. Therefore the random noise can be modeled in a more flexible fashion.

Specifically, CISS first maps the input x into a semantic space by an encoder $s(\cdot)$ and then adds random gaussian noise to $s(x)$. The prediction is made by taking the expectation over n and therefore the final smoothed classifier is formulated as: $\mathbb{E}_{n\sim p(n)}[f_y(s(x) + n)]$. The smoothed classifier built by CISS has provable robustness against attacks with theoretical guarantees, which is summarized in the following theorem (the proof of which is in the Appendix).

Theorem 3. *(Robustness by Semantic Smoothing) Given a base classifier $f_y(s(x)+n)$, where $s(\cdot)$ denotes an encoder that maps input into a semantic space and $p(n) \sim N(0, \sigma^2 I)$. If*

$$\mathop{\mathbb{E}}_{n\sim p(n)}[f_y(s(x)+n)] \geq \max_{y'\neq y} \mathop{\mathbb{E}}_{n\sim p(n)}[f_{y'}(s(x)+n)], \quad (3)$$

and $s(\cdot)$ satisfies

$$\|s(x) - s(\hat{x})\|_2 \leq \hat{R}, \forall \hat{x} \in \mathbb{B}_{adv}(x), \quad (4)$$

then classification by $\arg\max_{y'} \mathbb{E}_{n\sim p(n)}[f_{y'}(s(\hat{x})+n)]$ is robust for all $\hat{x}$ if $\hat{R} \leq R$, where

$$\begin{aligned} R = &\frac{\sigma}{2}(\Phi^{-1}(\mathop{\mathbb{E}}_{n\sim p(n)}[f_y(s(x)+n)]) \\ &- \Phi^{-1}(\max_{y'\neq y} \mathop{\mathbb{E}}_{n\sim p(n)}[f_{y'}(s(x)+n)])) \end{aligned} \quad (5)$$

and Φ^{-1} is the inverse of standard Gaussian c.d.f.

Similar to Theorem 2, CISS also has clear interpretability from a causal perspective. The smoothed classifier $\mathbb{E}_{n\sim p(n)}[f_y(s(x)+n)]$ also models the interventional distribution $p(y|do(x))$ that captures the causal effect from X to Y, as: $p(y|do(x)) = p^{N\nrightarrow X}(y|x) = \int\int p(c|x)p(y|c,n)p(n)\,dcdn$, where $p(c|x)$ puts a point mass on $s(x)$, and $p(y|c,n)$ is implemented as $f_y(s(x)+n)$.

Moreover, CISS has the following three merits compared to previous randomized smoothing techniques: (1) we are able to use the flexible and tractable Gaussian noise to smooth out discrete natural language perturbations; (2) we are able to find the optimal trade-off between robustness and accuracy, without tuning the hyper-parameter σ; (3) in addition to certified robustness, we can provide some empirical robustness even when the perturbations are strengthened by unknown attacks algorithms. These merits will be detailed in the following sections and empirically validated by our experiments.

3.4. The Training Objective of CISS

In this section we introduce the training objective of CISS in detail. Theorem 3 informs us how to certify the robustness at a data point (x, y) by randomized smoothing in the latent semantic space, but Theorem 3 does not directly guide us about how the training objective should be formulated. To derive a training objective for CISS, we need to look into the interventional distribution $p(y|do(x)) = \int\int p(c|x)p(y|c,n)p(n)\,dcdn$, and formulate the training objective of CISS by the following three parts.

1. Training The Base Classifier. The conditional distribution $p(y|c,n)$ corresponds to the base classifier $f_y(s(x)+n)$ in Theorem 3. Therefore, we train $f_y(s(x)+n)$ by the classical cross-entropy loss, and take the expectation over the observational data distribution $p(x,y)$, as:

$$\mathcal{L}_{\text{cls}} = \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \mathop{\mathbb{E}}_{n\sim p(n)} -\log f_y(s(x)+n). \quad (6)$$

2. Semantic Smoothing. As for $p(c|x)$, because $p(c|x)$ is defined according to the functional relationship between c and x by putting a point mass on $s(x)$, there is no need to derive an explicit training objective for it.

However, as we assume $p(n)$ as a Gaussian distribution, we need to align it with the perturbations in the semantic space, such that the gaussian noise can smooth out the perturbations. This actually requires us to regularize the encoder $s(\cdot)$, to meet the certification condition $\hat{R} \leq R$ in Theorem 3. We formulate this objective as follows:

$$\mathcal{L}_{\text{robust}} = \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \max(0, \hat{R} - R + m), \quad (7)$$

where m is a hyper-parameter that controls the margin between $\hat{R}$ and R. As R also has gradients with respect to the parameters of the base classifier f, this loss also helps optimize the base classifier.

3. IBP Encoder. The remaining problem is how to get $\hat{R}$ with respect to x and $s(\cdot)$. This can be achieved by employing the Interval Bound Propagation (IBP) techniques (Weng et al., 2018; Jia et al., 2019; Huang et al., 2019). We here employ the method provided by Jia et al. (2019) to build an IBP encoder $s(\cdot)$, and we take word substitution attacks as an example. To be specific, given an IBP encoder $s(\cdot)$, input x, and $\hat{x} \in \mathbb{B}_{\text{adv}}(x) = \{\hat{x} : \hat{x}^i \in \mathbb{S}_{\text{adv}}(x^i)\}$, we have:

$$s_l^i(x) \leq s^i(x) \leq s_u^i(x), \quad (8)$$

where $s^i(x)$ denotes the scalar value of the i^{th} dimension of $s(x)$, and $s_l^i(x)$ and $s_u^i(x)$ are the lowerbound and the upperbound of $s^i(x)$ respectively.

Thanks to the IBP techniques, here both $s_l^i(x)$ and $s_u^i(x)$ have gradients with respect to x and ϕ, the parameters of $s(\cdot)$. Therefore, it is favorable to use $s_l^i(x)$ and $s_u^i(x)$ to calculate $\hat{R}(x)$, as follows:

$$\|s(x) - s(\hat{x})\|_2 \leq \hat{R}(x) \quad (9)$$

$$= (\sum_i \max(s_u^i(x) - s^i(x), s^i(x) - s_l^i(x))^2)^{\frac{1}{2}}, \quad (10)$$

where $\hat{R}(x)$ also has gradients with respect to x and ϕ.

Final Training Objective: Our final training objective is linear combination of Eqs. 6 and 7 with hyperparameter γ:

$$\min_{\theta,\phi} \mathcal{L}_{\text{cls}} + \gamma\mathcal{L}_{\text{robust}}, \quad (11)$$

where θ and ϕ are the parameters of the encoder $s(\cdot)$ and the base classifier $f(\cdot)$ respectively.

Algorithm 1 Training of CISS

Input: Data from $p(x, y)$, hyperparameters σ, m, γ, and the parameters of Adam.
Output: parameters θ and ϕ.
repeat
  for random mini-batch $\sim p(x, y)$ **do**
    Compute $\mathcal{L}_{\text{cls}}$ and $\mathcal{L}_{\text{robust}}$ by Eqs. 6 and 7.
    Update θ and ϕ by Adam to minimize Eq. 11;
  end for
until the training converges.

Remark 1. (About σ) In previous randomized smoothing techniques, it is crucial to tune the std σ of gaussian noise to an appropriate value: a small σ will make the final classifier not smoothed enough and thus do harm to an invariant prediction, while a too big σ can overly smooth the input and thus impede clean accuracy. In contrast, CISS directly minimizes the gap between R and $\hat{R}$, which avoids the hyper-parameter tuning of σ. The encoder $s(\cdot)$ will automatically adapt to σ during the minimization of $\mathcal{L}_{\text{robust}}$, for a better accuracy-robustness trade-off. We support this remark by ablation on σ in Section 4.6 and trade-off in Section 4.5.

Remark 2. (About IBP) IBP does not fit deep architectures, as the bounds can get looser exponentially with the depth of the model architecture. As such many advanced deep neural architectures like Transformers, do not fit IBP well (shown in Table 1). However, in our framework, we only employ a shallow IBP encoder to map input into a continuous semantic space, and thus we benefit from the good mathematical property of IBP while do not suffer from the looseness.

Remark 3. (About the linear combination in Eq.11) We note that $\mathcal{L}_{\text{cls}} + \gamma\mathcal{L}_{\text{robust}}$ is an upper bound on the expected certification error $1-\mathbb{E}_{x,y\sim p(x,y)}[\mathbb{1}(R-\hat{R}>0)]$, as long as $\gamma \geq \frac{1}{m}$ (the proof of which is in the Appendix). $\mathcal{L}_{\text{cls}}$ makes the training more smoothed and γ should be sufficiently large to effectively minimize the certification error (ablation study on γ and m can be found in Section 4.6).

We solve the optimization by Adam (Kingma & Ba, 2015). The overall training procedure is summarized in Algorithm 1 and illustrated in Fig. 2.

3.5. Prediction and Robustness Certification of CISS

Prediction. Similar to Cohen et al. (2019), we employ Monte Carlo algorithms for our prediction and certification process. Given data point (x, y), we build g, a hard prediction version of f, to simplify any hypothesis test:

$$g(x) = \arg\max_{\overline{y}} f_{\overline{y}}(s(x) + n), \tag{12}$$

with corresponding final prediction c defined as:

$$c = \arg\max_{y'} \mathbb{P}\left[g(x) = y'\right]. \tag{13}$$

Algorithm 2 Prediction and Certification of CISS

Input: data point (x, y), $\mathbb{B}_{\text{adv}}(x)$, σ, $f(\cdot)$, $s(\cdot)$, t, and α.
function Prediction$(g, \sigma, x, t; \alpha)$
  sample from $p(n)$ t times to get $\{n_i\}_1^t$
  For each $j \in \mathcal{Y}$, $\text{cnt}_j = \mathbb{E}_{n\sim\{n_i\}_1^t}\ \mathbb{1}(g(x) = j)$
  $\text{cls}_A, \text{cls}_B \leftarrow$ top two indices in cnt
  If PvalueBinom(cnt_A, cnt_A + cnt_B, 0.5) $\leq \alpha$,
  Return cls_A with confidence $1-\alpha$
  Else Abstain from return
end function
function Certification$(g, \sigma, x, \mathbb{B}_{\text{adv}}(x), t_1, t_2; \alpha)$
  sample from $p(n)$ $t_1 + t_2$ times to get $\{n_i\}_1^{t_1+t_2}$
  $\text{cls}_A = \arg\max_{j\in\mathcal{Y}} \mathbb{E}_{n\sim\{n_i\}_1^{t_1}}\ \mathbb{1}(g(x) = j)$
  $\text{cnt}_A = \mathbb{E}_{n\sim\{n_i\}_{t_1+1}^{t_1+t_2}}\ \mathbb{1}(g(x) = \text{cls}_A)$
  $\underline{p_A}$ = LowerConfBound($\text{cnt}_A, t_2, 1-\alpha$)
  If $\underline{p_A} > \frac{1}{2}$ and $\sigma\Phi^{-1}(\underline{p_A}) \geq \hat{R}$ ($\hat{R}$ by Eq. 10)
  Return $g(\hat{x}) = \text{cls}_A, \forall\hat{x} \in \mathbb{B}_{\text{adv}}(x)$ with conf $1-\alpha$
  Else Abstain from return
end function

To calculate c with confidence $1-\alpha$, we sample t times and employ a hypothesis test summarized in Algorithm 2.

Certification. For the sake of robust certification, we employ hypothesis test to ensure our prediction is certifiably robust. The justification is based on Theorem 4 in the Appendix, which consider g as the base classifier instead of f. The certification procedure is summarized in Algorithm 2, where function *PvalueBinom* returns the p-value of the two-sided hypothesis test and function *LowerConfBound* returns the lower bound of estimated Binomial parameter. Detials of *PvalueBinom* and *LowerConfBound* are in appendix E.

4. Experiments

4.1. Experimental Setting

Tasks and Datasets. Following previous state-of-the-arts (Jia et al., 2019; Ye et al., 2020), we examine the certified robustness by text classification tasks, and we choose the prevailing YELP(Shen et al., 2017) and IMDB(Maas et al., 2011) datasets. Also aligned with previous SOTA methods on robust certification, we focus on comparing certified robustness against natural language word substitution attacks, while we also test and compare the empirical robustness of each method for a more extensive comparison.

Baselines. We compare our method with (i) Vanilla BERT (Devlin et al., 2019), (ii) IBP method (Jia et al., 2019), and (iii) SAFER (Ye et al., 2020), a randomized smoothing method based on BERT. As the original IBP method in Jia et al. (2019) employs shallow text CNNs, we also implement a BERT version for it referred to as IBP-BERT for fair comparisons. We compare with IBP method to show our

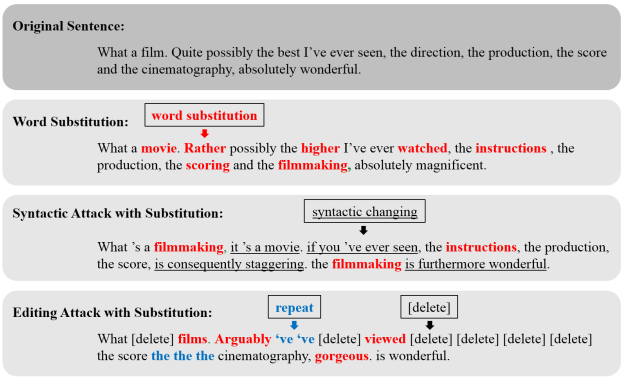

Figure 3. An illustration of the attacks we employed, using a random example from IMDB.

scalbility to deep architectures, and compare with SAFER to show the benefit of smoothing in the latent space.

Certification Setting. We are interested in examining the *certified robust accuracy*, which is defined as the fraction of the test set that is classified correctly with robust certifications. For all randomized smoothing based methods, we set $\alpha = 0.001$ to make sure that the certification result for each data point is correct with at least 99.9% confidence. For our method, sampling number is $t_1 = 50$ and $t_2 = 30000$ respectively. We use the same word substitution set as in Jia et al. (2019), which is constructed by the similarity of GloVe word embeddings (Pennington et al., 2014). The substitution table we use is from Jia et al. (2019) and is more complicated than that of SAFER (Ye et al., 2020). We do not use any language model constraint on the generated adversarial examples, and there is no limit on the number of substitutions per input.

Empirical Setting Against Word Substitutions. In this setting we aim to examine the empirical robustness of each method against a real attack algorithm. This also serves as a sanity check of our certification process. We employ ASCC (Dong et al., 2021a), an empirical word substitution attack that finds the worst-case perturbation inside the convex hull composed of substitutions.

Empirical Setting Against Word Substitutions Combined With Unseen Attacks. In this setting we aim to examine whether the proposed framework is robust against some unseen natural language attacks, when we only see word substitution attacks during training. (1) HiddenKiller (Qi et al., 2021), a syntactic-based attack, It paraphrases the original input sentence by altering its parser tree; word substitution and irrelevant sentences are added to strengthen this attack. (2) Editing attack, which is inspired by (Liang et al., 2018), not only substitutes words but also change the token positions and sentence structures, by adding repetitive words, substituting words, and deleting random words. This two attacks are unseen during the training phrase of our model and we further enhanced them by combining them with word substitution attacks. We give some qualitative examples of these attacks in Fig. 3.

Method	YELP	IMDB
IBP (Jia et al., 2019)	83.81	68.60
IBP-BERT	N.A.	N.A.
SAFER (Ye et al., 2020)	80.63	69.20
CISS (Ours)	**90.58**	**76.45**

Table 1. Certified robust accuracy (%) against word substitutions on YELP and IMDB. All compared methods use the same word substitution table from (Jia et al., 2019) for a fair comparison.

Method	YELP	IMDB
IBP (Jia et al., 2019)	84.45	74.80
Vanilla BERT	50.09	5.68
SAFER (Ye et al., 2020)	80.63	69.20
CISS (Ours)	**91.72**	**78.29**

Table 2. Empirical robustness (%) of each method on the YELP and IMDB, under the ASCC word substitution attack (using the same substitution table as in Table 1).

Implementation Details. In our experiments, we employ IBP-based CNN as our encoder s. For the base classifier, we use BERT (base-uncased) (Devlin et al., 2019). As BERT takes natural language as input, we additionally add a CNN decoder before BERT. We note that our framework can be also applied to other NLP architectures and if the base classifier is not a pre-trained language model then the decoder might not be necessary. For hyper-parameters, we set $\sigma = 1$, $\gamma = 4.0$, and margin $m = 1.0$ (ablation on hyper-parameters in section 4.6). These parameters are tuned to achieve the best certified robustness as shown in 4.6. During training, we first use loss $\mathcal{L}_{\text{cls}}$ to optimize the model to convergence, and then add loss $\mathcal{L}_{\text{robust}}$ for training. Warm-up is used on γ during optimization. During training, we sample only 1 time from the Gaussian to perform smoothing. For ASCC attack, we run for 10 iterations to find the worst-case attack, and then discretize the attack into textual adversarial examples. In editing attack, we use a editing distance of 10 and 50 on YELP and IMDB, respectively. Our code is available at https://github.com/zhao-ht/ConvexCertify.

4.2. Main Results

In this subsection, we examine the certified robustness against word substitutions, and compare our method with state-of-the-arts. As shown in Table 1, our model outperforms baselines on both IMDB and YELP with significant margin. Specifically, we achieve 90.58% certified robustness on YELP, and 76.45% certified robustness on IMDB, surpassing the runner-up by 6.8% and 7.2% respectively. The significant margin strongly validates the certified robustness of the proposed CISS framework.

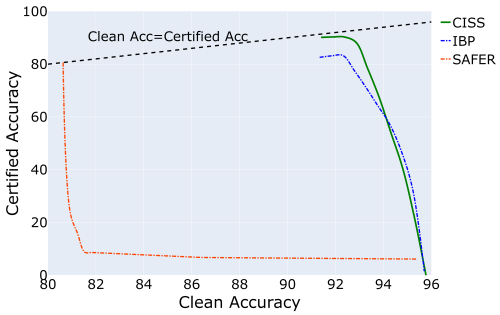

Figure 4. The trade-off curve between robustness and clean accuracy of compared methods. The black dashed line is the the upper bound of such a trade-off, as a model's robust accuracy is always lower than its clean accuracy.

Though IBP can achieve good certified robustness by using only shallow architectures like CNN, it hardly scales to deep models. For example, we test the certified robustness of IBP with BERT and the result is N.A.; this is because the interval bound grows exponentially with the depth of the architecture and finally becomes too loose to be used. SAFER smoothes in the input textual space but it relies on customized noise distribution over substitution words. To achieve robustness, its noise distribution over words forms a graph where many semantically unrelated words are forced to be connected, which impedes the discriminative power of a model and degrades its clean accuracy severely. See Fig. 4, where SAFER needs to sacrifice the clean accuracy from 96% down to 82% to increase some robustness. On the contrary, CISS smoothes in the latent semantic space and does not suffer from this problem.

4.3. Empirical Robustness Against Word Substitutions

In this section we employ a real word substitution attack , ASCC (Dong et al., 2021a), to examine the robustness of each compared method. The empirical robust accuracy also help us sanity check our certification procedure. The result is shown in Table 2. As we can see, the ASCC attack is quite strong: it degrades the accuracy of a vanilla BERT to 5.68% (from more than 92%), and the empirical robust accuracy of compared methods are just slightly higher than their corresponding certified robust accuracy. Nonetheless, CISS still outperforms all baselines with significant margins; *i.e.*, CISS surpasses the runner-up by 7.3% on YELP and 3.5% on IMDB, respectively.

4.4. Robustness Against Word Substitutions Combined With Unseen Natural Language Attacks

As proposed in our causal graph 1, attacks leverage the spurious correlations from $X \leftarrow N \rightarrow Y$ to fool a model, by manipulating N. There exist many different confounders manipulatable by different kind of attacks; *e.g.*, strict professional movie reviewer might prefer specific kind of syntactic structures. However, in training, we may not be able to enumerate all possible attacks to removing all possible manipulatable confounders. Here we conduct experiments and examine whether the proposed framework is still robust when there exist unseen attacks during testing. Specifically, we employ two sentence-level attacks that are both unseen during training, Syntactic-based attack (Qi et al., 2021) and Editing attack (Liang et al., 2018) , and further strength them with word substitution attacks, to examine whether the proposed method is still robust under such a challenge setting. As shown in Table 3, CISS still remains certain level of robustness, and still outperforms state-of-the-arts by around 7%. This owes to our smoothing in the semantic space, as it can remove some other latent confounder effects manipulatable by certain unseen attacks, and thus provide more generalized NLP robustness, while directly smoothing in the input space may not.

Method	YELP		IMDB	
Attack Method	Syntactic	Editing	Syntactic	Editing
IBP	73.7	75.3	67.7	67.1
SAFER	67.1	69.0	66.4	66.2
CISS (Ours)	**80.7**	**83.1**	**74.4**	**74.3**

Table 3. The certified accuracy against word substitution of our method and baselines on the YELP and IMDB datasets with sentence level confounders. Our model outperforms baselines under both syntactic and editing sentence level perturbations.

4.5. Accuracy-Robustness Trade-Off

Here we show the trade-off curves between certified robustness and clean accuracy of each method. For CISS, the trade-off is achieved by tuning γ. For SAFER, it is achieved by adjusting the scale of noise. For IBP, it is achieved by tuning the coefficient of training loss.

Figure 4 clearly shows that CISS achieves a better trade-off between robustness and clean accuracy compared to baselines. CISS achieves higher certified robust accuracy without sacrificing too much clean accuracy, and our curve is very closer to the theoretical upper bound, *i.e.*, the dashed black line representing clean accuracy = certified robust accuracy (as robust accuracy is always lower than the clean accuracy). This owes to that our encoder automatically adapts to our gaussian std σ towards the best trade-off curve of CISS, as we mentioned in Remark 1.

4.6. Ablation Study and Hyper-parameter Sensitivity

In this section, we do the ablation study by training CISS using different values of γ, m, and σ, and see how each hyper-parameter affects the performance. In this section, for the sake of efficiency, we only sample $t_2 = 300$ times in the second step of the certification, and set α to 0.05 so

When $m = 1$	Robustness	When $\gamma = 4$	Robustness
$\gamma = 0.25$	73.38	$m = -0.5$	81.70
$\gamma = 2$	89.72	$m = 0$	89.65
$\gamma = 4$	**90.47**	$m = 1$	**90.47**
$\gamma = 8$	90.15	$m = 2$	90.33

Table 4. Certified robust accuracy (%) of our method on YELP using different values of γ and m.

When $m = 1$	Robustness	When $\gamma = 4$	Robustness
$\gamma = 0.25$	48.13	$m = -0.5$	66.40
$\gamma = 2$	71.20	$m = 0$	70.26
$\gamma = 4$	**75.25**	$m = 1$	**75.25**
$\gamma = 8$	74.37	$m = 2$	75.20

Table 5. Certified robust accuracy (%) of our method on IMDB using different values of γ and m.

that we can obtain results comparable with Tabel 1. See the Appendix F for details.

First, Table 4 and Table 5 illustrate the certified robustness of CISS on YELP and IMDB using different values of γ and m. We observe that low γ like 0.25 and m like -0.5 can degrade robustness; When γ and m are sufficiently large, the final performance is not sensitive to small changes of γ and m. The above result supports the claim we made in Remark 3: our training objective $\mathcal{L}_{\text{cls}} + \gamma\mathcal{L}_{\text{robust}}$ is an upper bound on the certification error $1 - \mathbb{E}_{x,y\sim p(x,y)}[\mathbb{1}(R - \hat{R} > 0)]$, when $\gamma * m \geq 1$. Too small γ and m cannot ensure that our training objective is an upper bound and thus cannot support effective minimization of the certification error; when $\gamma * m \geq 1$ is sufficiently large, our training objective is already an upper bound on the certification error and thus the performance is not sensitive to small changes of γ or m.

We also conduct experiments to examine how the Gaussian std σ affect the performance of CISS, in order to support our Remark 1. Table 6 shows how different values of σ affect the performance of CISS. As demonstrated, changing the value of σ only affect the final performance to an ignorable extent; all the certified robustness showed in Table 6 are strong and surpass state-of-the-arts. Therefore, it can be argued that CISS does not rely on tedious tuning of σ to achieve a good performance, while previous randomized smoothing methods rely on the tuning of σ heavily.

Gaussian Std σ	0.5	1	2	4
Certified Robustness	90.30	90.47	90.43	90.14

Table 6. Certified robust accuracy (%) of our method on YELP using different values of Gaussian std σ.

5. Related work

Causality and Adversarial Robustness. Graphical causal inference (Pearl, 2009; Peters et al., 2017) aims at discovering the causal structure, calculating the causal effect of interventions, and answering counterfactual questions. 's interest in causality has significantly increased in recent years In recent years, it has drawn increasing attention from the machine learning community (Schölkopf, 2019; Schölkopf et al., 2021), *e.g.*, in few-shot learning (Teshima et al., 2020)(Yue et al., 2020), long-tail classification (Tang et al., 2020), and generative modeling (Sauer & Geiger, 2021). Causal inference has an instinct for modeling distribution change and adversarial robustness, *e.g.*, Zhang et al. (2020) Yang et al. (2019) Mitrovic et al. (2020) Tang et al. (2021). This work differs in that we propose a causal perspective to understand randomized smoothing, and achieve certified robustness by modeling causal effects.

NLP Attacks and Empirical Defenses. Adversarial attacks (Szegedy et al., 2013; Goodfellow et al., 2015; Papernot et al., 2016; Kurakin et al., 2016), are maliciously generated to fool DNNs while keeping innocuous to humans. In NLP, attacks algorithms include char-level modifications (Hosseini et al., 2017; Ebrahimi et al., 2018; Belinkov & Bisk, 2018; Gao et al., 2018; Eger et al., 2019; Pruthi et al., 2019), sequence-level manipulations (Iyyer et al., 2018; Ribeiro et al., 2018; Jia & Liang, 2017; Zhao et al., 2018; Qi et al., 2021), and adversarial word substitutions (Alzantot et al., 2018; Ren et al., 2019; Jin et al., 2020; Dong et al., 2021a; Zang et al., 2020). Adversarial training (Madry et al., 2018; Athalye et al., 2018; Miyato et al., 2017; Ebrahimi et al., 2018; Dong et al., 2021a;b) augments training by adversarial examples and is currently the most effective empirical defense in NLP defense.

Certified Robustness and Randomized Smoothing. This line of work aims at provable robustness with theoretical guarantees. This field can be sorted into deterministic methods and randomized smoothing methods. Deterministic certification includes Dual Approach (Dvijotham et al., 2018a;b), Interval Bound Propagation (IBP) (Wong & Kolter, 2018; Jia et al., 2019; Huang et al., 2019) and Linear Relaxation methods (Zhang et al., 2018; 2019; Shi et al., 2020). IBP methods certify the robustness by propagating interval bounds to bound the output, but are often too computationally expensive. In contrast, randomized smoothing (Lecuyer et al., 2019; Cohen et al., 2019) are more applicable to large models (Zhai et al., 2020). Due to the discrete NLP input space, Zeng et al. (2021) masks tokens randomly, and SAFER (Ye et al., 2020) constructs stochastic ensembles by random substitutions. Our method differs in that we smooth in the latent space, which makes our framework applicable to more general NLP attacks and frees us from constructing complicated attack-customized noise distributions.

6. Discussion and Conclusion

In this paper, we provide a novel causal perspective to understand model vulnerability, and propose a novel framework CISS to achieve certified NLP robustness. CISS consistently surpasses state-of-the-arts with significant margins. In future work, we plan to explore a more generalized framework towards robustness in both CV and NLP.

Acknowledgements

This work is supported by the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 1 (RS21/20) and Tier 2.

References

Aldrich, J. Autonomy. *Oxford Economic Papers*, 41(1): 15–34, 1989.

Alzantot, M., Sharma, Y., Elgohary, A., Ho, B.-J., Srivastava, M., and Chang, K.-W. Generating natural language adversarial examples. In *EMNLP*, 2018.

Athalye, A., Carlini, N., and Wagner, D. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018.

Belinkov, Y. and Bisk, Y. Synthetic and natural noise both break neural machine translation. In *ICLR*, 2018.

Cohen, J., Rosenfeld, E., and Kolter, Z. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*. PMLR, 2019.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

Dong, X., Luu, A. T., Ji, R., and Liu, H. Towards robustness against natural language word substitutions. In *ICLR*, 2021a.

Dong, X., Luu, A. T., Lin, M., Yan, S., and Zhang, H. How should pre-trained language models be fine-tuned towards adversarial robustness? In *NeurIPS*, 2021b.

Dvijotham, K., Gowal, S., Stanforth, R., Arandjelovic, R., O'Donoghue, B., Uesato, J., and Kohli, P. Training verified learners with learned verifiers. *arXiv preprint arXiv:1805.10265*, 2018a.

Dvijotham, K., Stanforth, R., Gowal, S., Mann, T. A., and Kohli, P. A dual approach to scalable verification of deep networks. *ArXiv*, abs/1803.06567, 2018b.

Ebrahimi, J., Rao, A., Lowd, D., and Dou, D. Hotflip: White-box adversarial examples for text classification. In *ACL*, 2018.

Eger, S., Şahin, G. G., Rücklé, A., Lee, J.-U., Schulz, C., Mesgar, M., Swarnkar, K., Simpson, E., and Gurevych, I. Text processing like humans do: Visually attacking and shielding nlp systems. In *NAACL*, 2019.

Gao, J., Lanchantin, J., Soffa, M. L., and Qi, Y. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *SPW*. IEEE, 2018.

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. *Deep learning*, volume 1. MIT Press, 2016.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Gopnik, A., Glymour, C., Sobel, D. M., Schulz, L. E., Kushnir, T., and Danks, D. A theory of causal learning in children: causal maps and bayes nets. *Psychological review*, 111(1):3, 2004.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 2012.

Hoover, K. D. Causality in economics and econometrics. *The new Palgrave dictionary of economics*, 2, 2008.

Hosseini, H., Kannan, S., Zhang, B., and Poovendran, R. Deceiving google's perspective api built for detecting toxic comments. *arXiv preprint arXiv:1702.08138*, 2017.

Huang, P.-S., Stanforth, R., Welbl, J., Dyer, C., Yogatama, D., Gowal, S., Dvijotham, K., and Kohli, P. Achieving verified robustness to symbol substitutions via interval bound propagation. In *EMNLP*, 2019.

Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*, 2019.

Iyyer, M., Wieting, J., Gimpel, K., and Zettlemoyer, L. Adversarial example generation with syntactically controlled paraphrase networks. In *NAACL*, 2018.

Jia, R. and Liang, P. Adversarial examples for evaluating reading comprehension systems. In *EMNLP*, 2017.

Jia, R., Raghunathan, A., Göksel, K., and Liang, P. Certified robustness to adversarial word substitutions. In *EMNLP*, 2019.

Jin, D., Jin, Z., Zhou, J. T., and Szolovits, P. Is bert really robust? natural language attack on text classification and entailment. *AAAI*, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial examples in the physical world, 2016.

Lecuyer, M., Atlidakis, V., Geambasu, R., Hsu, D., and Jana, S. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 656–672. IEEE, 2019.

Levenshtein, V. I. et al. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*. Soviet Union, 1966.

Liang, B., Li, H., Su, M., Bian, P., Li, X., and Shi, W. Deep text classification can be fooled. In *IJCAI*, 2018.

Maas, A., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 142–150, 2011.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Mitrovic, J., McWilliams, B., Walker, J., Buesing, L., and Blundell, C. Representation learning via invariant causal mechanisms. *arXiv preprint arXiv:2010.07922*, 2020.

Miyato, T., Dai, A. M., and Goodfellow, I. Adversarial training methods for semi-supervised text classification. In *ICLR*, 2017.

Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. The limitations of deep learning in adversarial settings. In *EuroS&P*. IEEE, 2016.

Pearl, J. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146, 2009.

Pennington, J., Socher, R., and Manning, C. D. Glove: Global vectors for word representation. In *EMNLP*, pp. 1532–1543, 2014.

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Pruthi, D., Dhingra, B., and Lipton, Z. C. Combating adversarial misspellings with robust word recognition. In *ACL*, 2019.

Qi, F., Li, M., Chen, Y., Zhang, Z., Liu, Z., Wang, Y., and Sun, M. Hidden killer: Invisible textual backdoor attacks with syntactic trigger. In *ACL*, 2021.

Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

Ren, S., Deng, Y., He, K., and Che, W. Generating natural language adversarial examples through probability weighted word saliency. In *ACL*, 2019.

Ribeiro, M. T., Singh, S., and Guestrin, C. Semantically equivalent adversarial rules for debugging nlp models. In *ACL*, 2018.

Sauer, A. and Geiger, A. Counterfactual generative networks. *arXiv preprint arXiv:2101.06046*, 2021.

Schölkopf, B. Causality for machine learning. *arXiv preprint arXiv:1911.10500*, 2019.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. Toward causal representation learning. *Proceedings of the IEEE*, 109(5): 612–634, 2021.

Shen, T., Lei, T., Barzilay, R., and Jaakkola, T. Style transfer from non-parallel text by cross-alignment. *arXiv preprint arXiv:1705.09655*, 2017.

Shi, Z., Zhang, H., Chang, K.-W., Huang, M., and Hsieh, C.-J. Robustness verification for transformers, 2020.

Sutskever, I., Vinyals, O., and Le, Q. V. Sequence to sequence learning with neural networks. In *NeurIPS*, 2014.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *ICLR*, 2013.

Tang, K., Huang, J., and Zhang, H. Long-tailed classification by keeping the good and removing the bad momentum causal effect. In *NeurIPS*, 2020.

Tang, K., Tao, M., and Zhang, H. Adversarial visual robustness by causal intervention. *arXiv preprint arXiv:2106.09534*, 2021.

Teshima, T., Sato, I., and Sugiyama, M. Few-shot domain adaptation by causal mechanism transfer. In *ICML*, 2020.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

Weng, L., Zhang, H., Chen, H., Song, Z., Hsieh, C.-J., Daniel, L., Boning, D., and Dhillon, I. Towards fast computation of certified robustness for relu networks. In *ICML*, 2018.

Wong, E. and Kolter, Z. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *ICML*, 2018.

Yang, C.-H. H., Liu, Y.-C., Chen, P.-Y., Ma, X., and Tsai, Y.-C. J. When causal intervention meets adversarial examples and image masking for deep neural networks. In *ICIP*. IEEE, 2019.

Ye, M., Gong, C., and Liu, Q. Safer: A structure-free approach for certified robustness to adversarial word substitutions. In *ACL*, 2020.

Yue, Z., Zhang, H., Sun, Q., and Hua, X.-S. Interventional few-shot learning. *arXiv preprint arXiv:2009.13000*, 2020.

Zang, Y., Qi, F., Yang, C., Liu, Z., Zhang, M., Liu, Q., and Sun, M. Word-level textual adversarial attacking as combinatorial optimization. In *ACL*, 2020.

Zeng, J., Zheng, X., Xu, J., Li, L., Yuan, L., and Huang, X. Certified robustness to text adversarial attacks by randomized [mask], 2021.

Zhai, R., Dan, C., He, D., Zhang, H., Gong, B., Ravikumar, P., Hsieh, C.-J., and Wang, L. Macer: Attack-free and scalable robust training via maximizing certified radius. In *ICLR*, 2020.

Zhang, C., Zhang, K., and Li, Y. A causal view on robustness of neural networks. In *NeurIPS*, 2020.

Zhang, H., Weng, T.-W., Chen, P.-Y., Hsieh, C.-J., and Daniel, L. Efficient neural network robustness certification with general activation functions. In *Advances in Neural Information Processing Systems (NuerIPS)*, dec 2018.

Zhang, H., Chen, H., Xiao, C., Gowal, S., Stanforth, R., Li, B., Boning, D., and Hsieh, C.-J. Towards stable and efficient training of verifiably robust neural networks, 2019.

Zhang, Y., Gong, M., Liu, T., Niu, G., Tian, X., Han, B., Schölkopf, B., and Zhang, K. Adversarial robustness through the lens of causality. *arXiv preprint arXiv:2106.06196*, 2021.

Zhao, Z., Dua, D., and Singh, S. Generating natural adversarial examples. In *ICLR*, 2018.

A. Proof of Theorem 1

We have:

$$p(y|do(x)) = p^{N\nrightarrow X}(y|x) \tag{14}$$
$$= \int p^{N\nrightarrow X}(y,n|x)\,dn \tag{15}$$
$$= \int p^{N\nrightarrow X}(y|x,n)p^{N\nrightarrow X}(n|x)\,dn \tag{16}$$
$$= \int p^{N\nrightarrow X}(y|x,n)p^{N\nrightarrow X}(n)\,dn \tag{17}$$
$$= \int p(y|x,n)p(n)\,dn. \tag{18}$$

B. Proof of Theorem 3

Define

$$B_h = \{h|\arg\max_{y'} \mathop{\mathbb{E}}_{n\sim p(n)}[f_{y'}(h+n)] = \arg\max_{y'} \mathop{\mathbb{E}}_{n\sim p(n)}[f_{y'}(s(x)+n)]\}$$

, and

$$B_R = \{h|\|s(x)-h\|_2 \le R\}$$

, where $R = \frac{\sigma}{2}(\Phi^{-1}(\mathbb{E}_{n\sim p(n)}[f_y(s(x)+n)]) - \Phi^{-1}(\max_{y'\neq y}\mathbb{E}_{n\sim p(n)}[f_{y'}(s(x)+n)])$, and Φ^{-1} is the inverse of standard Gaussian c.d.f. By applying Theorem 2, we have classification by $\arg\max_{y'}\mathbb{E}_{n\sim p(n)}[f_{y'}(h+n)]$ is robust for all h, s.t., $\|s(x)-h\|_2 \le R$, which means $B_R \subset B_h$.

According to the condition

$$\|s(x)-s(\hat{x})\|_2 \le \hat{R}, \forall \hat{x} \in \mathbb{B}_{\text{adv}}(x), \tag{19}$$

and $\hat{R} \le R$, we have

$$s(\mathbb{B}_{\text{adv}}(x)) \subset B_R$$

, which means

$$s(\mathbb{B}_{\text{adv}}(x)) \subset B_h$$

, i.e. $\arg\max_{y'}\mathbb{E}_{n\sim p(n)}[f_{y'}(s(x)+n)] = \arg\max_{y'}\mathbb{E}_{n\sim p(n)}[f_{y'}(s(\hat{x})+n)], \forall \hat{x} \in \mathbb{B}_{\text{adv}}(x)$. Thus we can get the conclusion that classification by $\arg\max_{y'}\mathbb{E}_{n\sim p(n)}[f_{y'}(s(\hat{x})+n)]$ is robust for all $\hat{x} \in \mathbb{B}_{\text{adv}}(x)$.

C. Theorem for Randomized Smoothing via a Hard Classifier

Note that our certification process will employ the hard prediction of f, *i.e.*, $\arg\max_{\overline{y}} f_{\overline{y}}(s(x)+n)$, to make hypothesis test easier to implement. The certification process is based on the following theorem (which is achieved by fitting Cohen et al. (2019) to the semantic smoothing setting).

Theorem 4. *(Robustness by Semantic Smoothing (Hard Classifier Certification)) Given x, y and a base classifier $f(s(x)+n)$, where $s(\cdot)$ denotes an encoder that maps input into a semantic space and $p(n) \sim N(0,\sigma^2 I)$. Define hard classifier $g(\cdot)$ as:*

$$g(x) = arg\max_{\overline{y}} f_{\overline{y}}(s(x)+n). \tag{20}$$

If

$$\mathbb{P}\left[g(x)=y\right] \ge \underline{p_A}, \underline{p_A} \in (\frac{1}{2}, 1], \tag{21}$$

and $s(\cdot)$ satisfies

$$\|s(x)-s(\hat{x})\|_2 \le \hat{R}, \forall \hat{x} \in \mathbb{B}_{adv}(x), \tag{22}$$

then classification by $\arg\max_{y'}\mathbb{P}\left[g(x)=y'\right]$ returns y for all $\hat{x}$, if $\hat{R} \le R = \sigma(\Phi^{-1}(\underline{p_A}))$ and Φ^{-1} is the inverse of standard Gaussian c.d.f.

D. Proof of Remark 3

Following the proof technique in Zhai et al. (2020), if $\gamma \geq \frac{1}{m}$, we have:

$$\mathcal{L}_{\text{cls}} + \gamma \mathcal{L}_{\text{robust}} = \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \mathop{\mathbb{E}}_{n\sim p(n)} -\log f_y(s(x)+n) + \gamma * \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \max(0, \hat{R} - R + m) \tag{23}$$

$$\geq \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \mathop{\mathbb{E}}_{n\sim p(n)} -\log f_y(s(x)+n) \tag{24}$$

$$+ \gamma * \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \max(0, \hat{R} - R + m) * \mathbb{1}(arg\max_{c\in Y} \mathop{\mathbb{E}}_{n\sim p(n)} f_y(s(x)+n) = y) \tag{25}$$

$$\geq \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \mathbb{1}(arg\max_{c\in Y} \mathop{\mathbb{E}}_{n\sim p(n)} f_y(s(x)+n) \neq y) \tag{26}$$

$$+ \frac{1}{m} * \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \max(0, \hat{R} - R + m) * \mathbb{1}(arg\max_{c\in Y} \mathop{\mathbb{E}}_{n\sim p(n)} f_y(s(x)+n) = y) \tag{27}$$

$$\geq \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \mathbb{1}(arg\max_{c\in Y} \mathop{\mathbb{E}}_{n\sim p(n)} f_y(s(x)+n) \neq y) \tag{28}$$

$$+ \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \mathbb{1}(arg\max_{c\in Y} \mathop{\mathbb{E}}_{n\sim p(n)} f_y(s(x)+n) = y, R - \hat{R} \leq 0) \tag{29}$$

$$\geq \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \mathbb{1}(R - \hat{R} \leq 0) \tag{30}$$

$$= 1 - \mathop{\mathbb{E}}_{x,y\sim p(x,y)} \mathbb{1}(R - \hat{R} > 0). \tag{31}$$

E. Details of *PvalueBinom* and *LowerConfBound*

We use the same statistic testing method as the (Cohen et al., 2019).

PvalueBinom(cnt_A, cnt_A + cnt_B,p) calculates the p-value of the two-sided hypothesis test that $\text{cnt}_A \sim$ Binomial(cnt_A + cnt_B, p), which is implied as *scipy.stats.binom test*(cnt_A, cnt_A + cnt_B,p).

LowerConfBound($\text{cnt}_A, t_2, 1-\alpha$) calculates the one-sided $(1-\alpha)$ lower confidence interval for the Binomial parameter p given that $\text{cnt}_A \sim$ Binomial(t_2, p), and is implied as *statsmodels.stats.proportion.proportion_confint*(cnt_A, t_2, alpha=$2*\alpha$, method="beta")[0].

F. Trade off between significance criterion α and sampling number t_2

The p-value of a statistical test is strongly correlated with the sample number. Our main results use the sampling number $t_2 = 30000$ in the second step of certification, which can get p-value lower than the significance criterion of $\alpha = 0.001$. However, this consumes around 12 hours to complete the certification using a Tesla V100 for IMDB test set of size 25000. Therefore, in the ablation experiment, in order to improve the efficiency of certification, we use $t_2 = 300$, and adjust α to 0.05. The comparison table 7 below shows that such a setup can achieve similar certification robustness to the main results.

Method	YELP	IMDB
CISS (30000,0.001)	**90.58**	**76.45**
CISS (300,0.05)	90.47	75.25

Table 7. Comparison between different certification settings.