

ROIC-DM: Robust Text Inference and Classification via Diffusion Model

Shilong Yuan¹, Wei Yuan², Tieke HE¹

¹Nanjing University

²The University of Queensland

shilongyuan@nju.edu.cn, w.yuan@uq.edu.au, hetieke@nju.edu.cn

Abstract

While language models have made many milestones in text inference and classification tasks, they remain susceptible to adversarial attacks that can lead to unforeseen outcomes. Existing works alleviate this problem by equipping language models with defense patches. However, these defense strategies often rely on impractical assumptions or entail substantial sacrifices in model performance. Consequently, enhancing the resilience of the target model using such defense mechanisms is a formidable challenge. This paper introduces an innovative model for robust text inference and classification, built upon diffusion models (ROIC-DM). Benefiting from its training involving denoising stages, ROIC-DM inherently exhibits greater robustness compared to conventional language models. Moreover, ROIC-DM can attain comparable, and in some cases, superior performance to language models, by effectively incorporating them as advisory components. Extensive experiments conducted with several strong textual adversarial attacks on three datasets demonstrate that (1) ROIC-DM outperforms traditional language models in robustness, even when the latter are fortified with advanced defense mechanisms; (2) ROIC-DM can achieve comparable and even better performance than traditional language models by using them as advisors.

1 Introduction

Text inference and classification are two fundamental and significant tasks in Natural Language Processing (NLP) (Li et al. 2022a). In recent years, large language models have made impressive advancements in these two tasks, however, these models are pointed out to be extremely vulnerable to textual adversarial attacks (Papernot et al. 2016), wherein adversaries can easily compromise the performance of these models by crafting deceptive inputs. To address this issue, many defense methods have been proposed to improve the adversarial robustness of these language models. These defense methods can generally be classified into three categories: adversarial detection (Mozes et al. 2021; Zhou et al. 2019a), adversarial training (Wang et al. 2021a; Zhu et al. 2020; Madry et al. 2018), and adversarial purification (Li, Song, and Qiu 2023a).

Although these defense approaches can slightly alleviate the threats of adversarial attacks to some extent, they ei-

ther rely on strong assumptions or sacrifice too much model performance, limiting their practical usage (Li, Song, and Qiu 2023a). Specifically, adversarial detection (Mozes et al. 2021; Mosca et al. 2022) and adversarial training methods (Zhu et al. 2020; Madry et al. 2018; Wang et al. 2021b) require prior knowledge of adversarial samples. However, in real-world scenarios, these adversarial samples are unavailable before potential attacks are launched. Even when attacks have been executed, differentiating and collecting the adversarial samples from input data remains challenging since the perturbations in these samples are imperceptible. Adversarial purification methods can avoid this dilemma as they can purify adversarial text without requiring knowledge of the specific attacks. Nevertheless, since purification operations have to be applied to all input data, these methods will deteriorate the model’s performance, as the original information of clean data will be inevitably modified during the purification process. Achieving the removal of adversarial perturbations while preserving the original meanings of the data remains a challenging task for these purification methods.

Considering the limitations of existing defense methods, it is not easy to improve the adversarial robustness of existing language models by equipping them with certain defending patches. In light of this, this paper proposes to explore a new kind of text inference and classification model that is robust enough against most adversarial attacks.

Diffusion models, as a new kind of generative models, have exhibited powerful learning ability in the computer vision community (Kong et al. 2021; Ramesh et al. 2022; Dhariwal and Nichol 2021). Naturally, some researchers attempt to transplant them into natural language processing areas (Li et al. 2022c; Gong et al. 2023; Savinov et al. 2022; Austin et al. 2021). But all these works are focusing on text generation tasks since diffusion models are generally used as generative models.

In this paper, we take the first step to investigate the usage of diffusion models in text inference and classification contexts. The basic motivation for replacing the traditional language models with diffusion models is that the diffusion models contain the diffusion-denoising steps so that they themselves would be more robust as they can more accurately estimate the whole data space (Chen et al. 2023). The left part of Figure 1 explains our motivation. During the re-

verse process, the model f_θ will “match” the text \mathbf{x} with many polluted \mathbf{y}_t until it removes all the noises. As a result, the data space of (x, y) pairs is large and f_θ would be robust to adversarial perturbations. However, applying diffusion models in text inference and classification tasks faces many challenges. Firstly, diffusion models are originally generative models, how to employ them for text classification and inference is non-trivial. Besides, the classical language models have been developed for a long duration in text classification and inference tasks with tons of researchers’ efforts, therefore, naively leveraging the new style of models in these fields may fail to achieve comparable performance. It is significant to incorporate the previous language models’ abilities in the diffusion models to gain better performance.

To address the above challenges, we propose ROIC-DM (*Robost text inference and classification diffusion model*), which is the first text inference and classification diffusion model. ROIC-DM first modifies the original diffusion models to make them suitable for text classification and inference tasks. Then, to further improve the effectiveness and efficiency, ROIC-DM incorporates the traditional language models as advisors to provide advice during the denoising process. It is worth noting that ROIC-DM is much different from text purification (Li, Song, and Qiu 2023a), as the latter only uses diffusion models for data preprocessing while ROIC-DM directly utilizes diffusion models to solve the tasks. Extensive experiments on three real-world datasets demonstrate that (1) ROIC-DM is more robust to adversarial attacks than existing language models even when they are equipped with advanced defense methods. (2) ROIC-DM can achieve comparable and even better performance than traditional language models by using them as advisors. The main contributions of this paper are as follows.

- To the best of our knowledge, we are the first to introduce diffusion models in text classification and inference fields.
- We propose ROIC-DM which achieves better performance and robustness than language models.
- We conduct extensive experiments on three real-world datasets including AG NEWS, SST, and MRPC. The experimental results showcase the effectiveness and robustness of our proposed methods.

2 Related Work

2.1 Textual Adversarial Attacks

Although language models have made great achievements, they are revealed to be vulnerable to adversarial attacks, i.e., these language models can be easily manipulated by adversaries via minor revisions of normal samples. Generally, most existing textual adversarial attacks achieve their malicious goals by replacing specific words in input texts (Alzantot et al. 2018; Jin et al. 2020; Ren et al. 2019). These methods usually assume that the model is black-box but the logits of the output prediction are available. Then, they attempt to design some strategies to find appropriate words to replace. For example, (Jin et al. 2020; Mrkšić et al. 2016; Ren et al.

2019) replace words with synonyms, while (Jin et al. 2020; Li et al. 2020) uses greedy-search methods to get substitutes.

2.2 Textual Adversarial Defenses

To improve the robustness of these language models, many defenses are proposed. These defense approaches can be generally classified into adversarial training, adversarial detection, and adversarial purification. The line of adversarial training (Wang et al. 2021a, 2020; Zhu et al. 2020; Zhou et al. 2019b; Madry et al. 2018) is to incorporate perturbations during a model’s training process so that the model can be robust to the potential risks. The works of adversarial detection (Mozes et al. 2021; Zhou et al. 2019a) aim to filter out the adversarial samples. The purification methods (Samangouei, Kabkab, and Chellappa 2018; Li, Song, and Qiu 2023b) employ generative models to purify adversarial inputs before feeding them to a model. However, all these defense methods have certain limitations. To be specific, the adversarial training methods and detection approaches require prior knowledge of attacks which is infeasible in practice, while the adversarial purification methods cannot avoid the modification of normal inputs. As a result, using defense methods to improve language models’ robustness is challenging.

3 Preliminaries: Diffusion Models

In this section, we introduce the theory of diffusion model (Sohl-Dickstein et al. 2015; Ho, Jain, and Abbeel 2020a; Song, Meng, and Ermon 2020). Generally, a diffusion model processes data in two steps. First, it gradually transforms raw data into a Gaussian distribution through a diffusion process. Then, It learns the reverse procedure to reconstruct the original data from Gaussian white noise. (Sohl-Dickstein et al. 2015). The following paragraphs formally describe these two processes.

In **diffusion process**, the diffusion model incrementally corrupts the original representation \mathbf{x}_0 into a Gaussian noise \mathbf{x}_t via a Markov Chain with fixed parameters (Ho, Jain, and Abbeel 2020a) in T steps:

$$q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N} \left(\mathbf{x}_t; \sqrt{1 - \beta_t} \mathbf{x}_{t-1}, \beta_t \mathbf{I} \right) \quad (1)$$

where $\mathcal{N}(x; \mu, \sigma^2)$ represents x sampled from a Gaussian distribution with a mean μ and variance σ^2 . The value of β_t is determined by a pre-defined noise schedule β that regulates the amount of noise injected at each step. The common noise schedules encompass square-root (Li et al. 2022b), cosine (Ho, Jain, and Abbeel 2020a), and linear (Nichol and Dhariwal 2021) functions. According to (Ho, Jain, and Abbeel 2020b), \mathbf{x}_t can be directly computed conditioned on \mathbf{x}_0 with the following transformation:

$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N} \left(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I} \right) \quad (2)$$

$$\bar{\alpha}_t = \prod_{i=1}^t \alpha_i, \quad \alpha_i = 1 - \beta_i. \quad (3)$$

With the trick of re-parameter, we can get \mathbf{x}_t by adding Guassian noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ to \mathbf{x}_0 as follows:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon \quad (4)$$

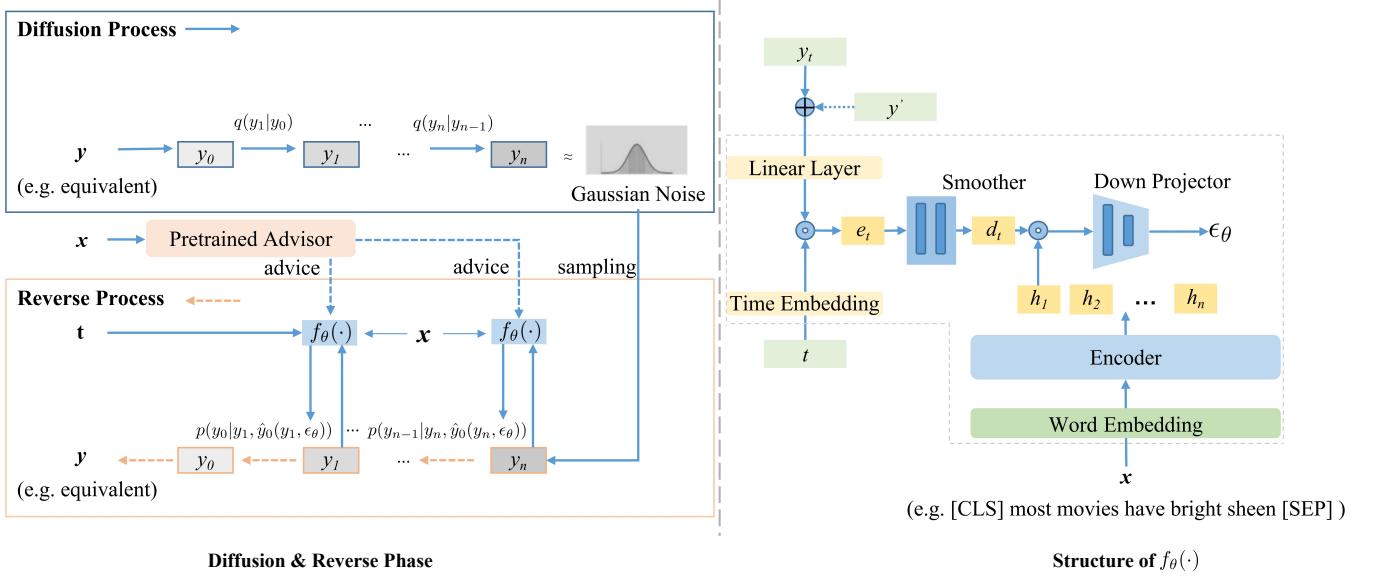


Figure 1: Illustration of our proposed Robust text Inference and Classification Diffusion Model (ROIC-DM). The figure on the left illustrates the diffusion phase and the reverse phase. The figure on the right illustrates the architecture of noise estimator f_θ . \mathbf{x} is the input text; \mathbf{y} is the categorical label; t is the time step number; ϵ_θ is the predicted noise. The \odot is the element-wise product. The \oplus indicates element-wise summation.

The **reverse process** is a Markov Chain with a learnable θ to denoise \mathbf{x}_T to \mathbf{x}_0 . Specifically, when provided with the current representation \mathbf{x}_s , the subsequent representation \mathbf{x}_{s-1} after denoising is calculated as follows:

$$p(\mathbf{x}_{s-1}|\mathbf{x}_s, \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_{s-1}; \tilde{\mu}_s(\mathbf{x}_s, \mathbf{x}_0), \tilde{\beta}_s \mathbf{I}\right) \quad (5)$$

$$\tilde{\mu}_s(\mathbf{x}_s, \mathbf{x}_0) = \frac{\sqrt{\alpha_{s-1}}\beta_s}{1-\bar{\alpha}_s} \mathbf{x}_0 + \frac{\sqrt{\alpha_s}(1-\bar{\alpha}_{s-1})}{1-\bar{\alpha}_s} \mathbf{x}_s \quad (6)$$

$$\tilde{\beta}_s = \frac{1-\bar{\alpha}_{s-1}}{1-\bar{\alpha}_s} \beta_s \quad (7)$$

Since \mathbf{x}_0 is unknown during the reverse phase, the diffusion model utilizes a noise estimator $f_\theta(\mathbf{x}_t, t)$, which is typically modelled by a deep neural network, such as Transformer (Vaswani et al. 2017) or U-Net (Ronneberger, Fischer, and Brox 2015). We optimize the reverse phase, by optimizing the variational lower bound(VLB)

$$\begin{aligned} \mathcal{L}_{vlb} = & \underbrace{\mathbb{E}_q [D_{KL}(q(\mathbf{x}_t|\mathbf{x}_0)||p(\mathbf{x}_t))]_{L_t}} \\ & + \underbrace{\mathbb{E}_q \left[\sum_{s=2}^t D_{KL}(q(\mathbf{x}_{s-1}|\mathbf{x}_s, \mathbf{x}_0)||p_\theta(\mathbf{x}_{s-1}|\mathbf{x}_s)) \right]_{L_{t-1}}}_{L_0} \\ & - \underbrace{\log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0} \end{aligned} \quad (8)$$

Finally, after simplifying (Kingma et al. 2021), the diffusion loss is as follows:

$$\mathcal{L}_{simple} = \mathbb{E}_{t, \mathbf{x}_0, \epsilon} [\|\epsilon - f_\theta(\mathbf{x}_t, t)\|^2] \quad (9)$$

More details of the general diffusion model can be found in previous works (Ho, Jain, and Abbeel 2020a; Song, Meng, and Ermon 2020).

4 Methodology

In this section, we first introduce how to apply diffusion models in text classification and inference, i.e., ROIC-DM. Then, we describe how to incorporate the knowledge from pre-trained models to ROIC-DM to improve the model performance. Finally, we present the detailed model architecture of ROIC-DM. The overview of our ROIC-DM is illustrated in Figure 1.

4.1 Diffusion Model for Text Classification and Inference

The skeleton of ROIC-DM is the diffusion model. As diffusion models exhibit strong learning ability in Computer Vision, many works attempt to employ them to solve NLP tasks (Li et al. 2022b; Gong et al. 2022). However, all of these works mainly utilize diffusion models to solve natural language generation problems since diffusion models are a kind of generative model. This paper takes the first step to modifying diffusion models as robust classifiers to solve text classification and inference problems. The technical details are as follows.

In the diffusion process, ROIC-DM gradually adds noise to the label \mathbf{y}_0 . Note that since the label \mathbf{y}_0 usually can be denoted as a one-hot vector, ROIC-DM can directly add noise to it (Hoogeboom et al. 2021; Han, Zheng, and Zhou 2022). Therefore, the only difference in the diffusion process between ROIC-DM and other standard diffusion models is that

our input is a label while others are images or sentences. As a result, the diffused \mathbf{y} in each timestep can be obtained as follows:

$$\mathbf{y}_t = \sqrt{\bar{\alpha}_t} \mathbf{y}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (10)$$

Algorithm 1: Training procedure of ROIC-DM

Input: text x , learning epochs E , maximum diffusion steps T , linear schedule β , model $f_\theta(\cdot)$, advisor prediction \mathbf{y}' (optional), \dots
Output: well trained $f_\theta(\cdot)$;

```

1: while  $j < E$  do
2:    $t \sim \text{uniform}(\{0, \dots, T\})$ ;
3:    $\mathbf{y}_t, \epsilon \leftarrow \text{E.q. 10}$ 
4:   if use advisor then
5:      $\epsilon_\theta = f_\theta(\mathbf{x}, \mathbf{y}_t, t, \mathbf{y}')$ 
6:   else
7:      $\epsilon_\theta = f_\theta(\mathbf{x}, \mathbf{y}_t, t)$ 
8:   end if
9:   update  $f_\theta(\cdot)$  using  $\nabla_{\theta} \|\epsilon - \epsilon_\theta\|^2$ 
10:   $j = j + 1$ ;
11: end while
```

The critical difference between ROIC-DM and the generative diffusion models is the reverse process. Specifically, for traditional diffusion models, during their reverse process, they utilize a noise estimator $f_\theta(\mathbf{x}_t, t)$ to predict the noise (note that the \mathbf{x}_t in this case is equal to \mathbf{y}_t in ROIC-DM). However, in ROIC-DM, directly denoising \mathbf{y}_t without any conditions is meaningless since it is just a certain category label (Hoogeboom et al. 2021; Han, Zheng, and Zhou 2022). Thus, ROIC-DM constructs a trainable model $f_\theta(\mathbf{x}, \mathbf{y}_t, t)$ whose goal is to generate a noise ϵ_θ to recover \mathbf{y}_t to \mathbf{y}_{t-1} considering the corresponding text context \mathbf{x} .

Algorithm 1 shows how to train the model $f_\theta(\mathbf{x}, \mathbf{y}_t, t)$. To be specific, ROIC-DM randomly selects a pair of data (\mathbf{x}, \mathbf{y}) and calculates \mathbf{y}_t using E.q. 10. Then, it predicts the noise using $f_\theta(\mathbf{x}, \mathbf{y}_t, t)$ and calculates the loss with E.q. 11. Above steps iteratively continue until model convergence.

$$\mathcal{L} = \mathbb{E}_{t, \mathbf{x}_0, \epsilon} [\|\epsilon - f_\theta(\mathbf{x}, \mathbf{y}_t, t)\|^2] \quad (11)$$

The inference procedure of ROIC-DM is displayed in Algorithm 2. In the inference phase, ROIC-DM randomly samples a Gaussian noise \mathbf{y}_T as the start point. Then, ROIC-DM recovers \mathbf{y}_T to \mathbf{y}_0 by repeating the following equations:

$$\begin{aligned}
\mathbf{y}_{t-1} &= \tilde{\mu}(\hat{\mathbf{y}}_0, \mathbf{y}_t) + \tilde{\beta}_t * \zeta \\
\tilde{\mu}_t(\mathbf{y}_t, \hat{\mathbf{y}}_0) &= \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \hat{\mathbf{y}}_0 + \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{y}_t \\
\tilde{\beta}_t &= \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t \\
\hat{\mathbf{y}}_0 &= \frac{1}{\sqrt{\bar{\alpha}_t}} (\mathbf{y}_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta)
\end{aligned} \quad (12)$$

where $\zeta \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, ϵ_θ is calculated by well-trained $f_\theta(\cdot)$. After obtaining \mathbf{y}_0 , ROIC-DM selects the position with the maximum value as the final predicted class for \mathbf{x} .

Algorithm 2: Inference procedure of ROIC-DM

Input: text x , total reverse steps T , linear schedule β , model $f_\theta(\cdot)$, advisor prediction $\mathbf{y}' \dots$
Output: label

```

1: sample  $\mathbf{y}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
2:  $t = T$ 
3: while  $t > 0$  do
4:   sample  $\zeta \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
5:   if use advisor then
6:      $\epsilon_\theta = f_\theta(\mathbf{x}, \mathbf{y}_t, t, \mathbf{y}')$ 
7:   else
8:      $\epsilon_\theta = f_\theta(\mathbf{x}, \mathbf{y}_t, t)$ 
9:   end if
10:   $\mathbf{y}_{t-1} \leftarrow \text{calculate E.q. 12 with } \zeta$ 
11:   $t = t - 1$ 
12: end while
13: label  $\leftarrow$  find the position with max value in  $\mathbf{y}_0$ 
```

4.2 Pre-trained Advisor Improved ROIC-DM

Although language models have been revealed to be vulnerable to adversarial attacks, they have developed for a long time and have achieved remarkable results in text classification and inference tasks, especially the large pre-trained language models (Devlin et al. 2018; Minaee et al. 2021). Therefore, it is necessary to build our ROIC-DM based on the power of pre-trained language models.

To achieve that, before training ROIC-DM, we first fine-tune a large pre-trained language model (e.g., BERT (Devlin et al. 2018)) on the target classification dataset. Then, during the reverse process, ROIC-DM generates ϵ_θ not only considering \mathbf{x} , but also the prediction from the fine-tuned pre-trained model. Specifically, we transfer the knowledge of fine-tuned model to ROIC-DM by utilizing its soft-label \mathbf{y}' based on the input \mathbf{x} , which is inspired by the observations from knowledge distillation (Gou et al. 2021) said that model's soft label contains many useful auxiliary information. In ROIC-DM, we incorporate the fine-tuned model's knowledge by directly adding \mathbf{y}' to \mathbf{y}_t . The left part of Figure 1 shows the framework of our ROIC-DM.

4.3 Model Architecture of $f_\theta(\cdot)$

The right part of Figure 1 presents the model architecture of the noise estimator $f_\theta(\cdot)$ in ROIC-DM. Generally, f_θ contains an encoder to extract features from the context \mathbf{x} , a time embedding table to encode the timestep, a normalization layer to make the output be more smooth, and a down projector to output the noise.

Encoder We use the encoder block of the BERT (Devlin et al. 2018) as the feature extractor to convert the words to hidden states. Then, we leverage the hidden state of $[CLS]$ token¹, which is appended at the start of the text, as the feature vector of text \mathbf{x} , i.e., \mathbf{h}_1 in Figure 1.

$$\mathbf{h}_1 \leftarrow \text{Encoder}(\mathbf{x}) \quad (13)$$

¹We also tried using the average of hidden states as the feature vector and obtain similar results.

Time Embedding and Linear Layer The diffusion step t is uniformly sampled randomly from $\{0, \dots, T\}$. Then, we leverage a time embedding table to learn the features of time steps t (Xiao, Kreis, and Vahdat 2021). For \mathbf{y}_t , we utilize a full-connected layer to transform it to the same size as t and then conduct element-wise product to fuse the information of t and \mathbf{y}_t .

$$\mathbf{e}_t = \mathbf{t} \odot \text{Linear}(\mathbf{y}_t) \quad (14)$$

Smoother Smoother is consisted of a softmax function and a layer normalization. Such a combination can be used in certain layers of neural networks to enhance the network’s representational capacity and convergence performance and lead to improved performance and faster training speed in learning tasks (Huang et al. 2023). Here, we use this module to process the fused vector \mathbf{e}_t .

$$\mathbf{d}_t = LN(Softplus(\mathbf{e}_t)) \quad (15)$$

where $LN(\cdot)$ is layer normalization.

Down Projector In the down projector, we first conduct element-wise product to fuse \mathbf{d}_t and text feature vector \mathbf{h}_1 . Then, a stack of linear layers followed by softmax and layer normalization are leveraged to predict the noise ϵ_θ .

$$\epsilon_\theta = \text{projector}(\mathbf{d}_t, \mathbf{h}_1) \quad (16)$$

Table 1: Detailed statistics of the datasets.

| Dataset | Training Set | Test Set | #Avg. words |
|---------|--------------|----------|-------------|
| AG NEWS | 120K | 7.6K | 43 |
| SST-2 | 67K | 1.8K | 19 |
| MRPC | 3.7K | 1.7K | 44 |

5 Experiments

In this section, we first introduce the basic experimental settings, and then, present the experimental results with comprehensive analysis to showcase the superiority of our proposed methods.

5.1 Datasets

In this paper, we conduct experiments on three widely used text classification and inference datasets: AG NEWS (Zhang, Zhao, and LeCun 2015), SST-2 (Socher et al. 2013), and MRPC (Dolan and Brockett 2005). The statistics of these involved datasets are illustrated in Table 1, including the size of the training/test set and the average word count of the training samples. The training and test set division is following (Liu et al. 2022). We use the whole test set to evaluate model accuracy and randomly select 500 samples for robustness evaluation since the attack process is seriously slow (Morris et al. 2020; Moon et al. 2023).

5.2 Baselines

We compare our ROIC-DM with the following defense baselines including two adversarial training algorithms (PGD and FreeLB), one regularization method (InfoBERT), and a text purification method.

PGD (Madry et al. 2018) formulates adversarial training as a minimax problem, which aims to minimize the empirical loss on adversarial examples that could potentially lead to adversarial risk.

FreeLB (Zhu et al. 2020) attempts to improve language models’ robustness by enhancing their generalization abilities. Specifically, FreeLB generates some virtual adversarial samples by injecting adversarial perturbations into word embeddings. Then, FreeLB mixes them with normal training data to improve the tolerance of target models to adversarial samples.

InfoBERT (Wang et al. 2021a) consists of two mutual-information-based regularizers to improve the robustness of the learned representations by suppressing noisy mutual information.

Text Purification (Li, Song, and Qiu 2023b) is a textual adversarial purification algorithm. It utilizes the mask-infill ability of pre-trained models to recover noisy texts and use these purified texts to make predictions.

5.3 Attack Methods and Evaluation Metrics

Textfooler (Jin et al. 2020) and BERT-Attack (Li et al. 2020) have exhibited capable of effectively deceiving robust models in text classification and inference tasks with limited perturbations. These two attack methods have been widely used in adversarial robustness research (Liu et al. 2022; Morris et al. 2020). Therefore, in this paper, we leverage these two adversarial attack methods to assess the robustness of our ROIC-DM.

TextFooler identifies crucial words within the input text for the target model and iteratively replaces them with synonyms until the model’s prediction is modified. BERT-attack utilizes BERT in a manner that preserves semantics when generating substitute words for the identified vulnerable words in the input text.

We utilize the following metrics to evaluate models’ resistance to above mentioned adversarial attacks:

Clean% is the victim model’s accuracy tested on the clean test set, which represents the original performance of the victim model.

Aua% is short for accuracy under attacks(Aua). It measures the prediction accuracy of victim models on the adversarial data generated by certain attacks. This metric reflects the defensive capability of the model against adversarial attacks. A higher *Aua%* indicates a more robust model.

Suc% is the attack success rate. i.e., it is the ratio of the number of texts that have been successfully perturbed by certain attack methods to the number of involved texts. A lower *Suc%* means a more robust model.

5.4 Implementation Details

For ROIC-DM, we set the number of diffusion timesteps to $T = 1000$, and employ a linear noise schedule with $\beta_1 = 1e-4$ and $\beta_T = 0.02$. The optimizer is AdamW (Loshchilov and Hutter 2017) with a linear decay learning rate starting

Table 2: Experimental results of adversarial robustness evaluation. The best performance is marked in **bold**. For the MRPC task, we attack both hypothesis and premise. Methods labelled by \dagger are fine-tuning baselines without considering adversarial defence.

| Dataset | Method | Clean% | TextFooler | | BERT-Attack | |
|---------|---|-------------|-------------|-------------|-------------|-------------|
| | | | Aua% | Suc% | Aua% | Suc% |
| AG NEWS | BERT \dagger | 94.0 | 20.5 | 78.9 | 14.6 | 84.3 |
| | PGD (Li and Qiu 2021) | 94.8 | 36.2 | 61.3 | 32.8 | 65.7 |
| | Free LB (Zhu et al. 2020) | 94.7 | 34.8 | 63.5 | 12.7 | 86.7 |
| | InfoBERT (Ishida et al. 2020) | 94.9 | 30.4 | 65.9 | 20.4 | 78.0 |
| | Text Purification (Li, Song, and Qiu 2023b) | 93.0 | 51.0 | 42.0 | 44.5 | 48.5 |
| | ROIC-DM | 95.1 | 78.7 | 18.4 | 49.0 | 47.0 |
| SST-2 | BERT \dagger | 92.2 | 13.9 | 80.4 | 13.3 | 80.6 |
| | PGD | 93.2 | 13.4 | 82.1 | 13.4 | 84.5 |
| | Free LB | 92.2 | 19.4 | 80.3 | 12.1 | 87.1 |
| | InfoBERT | 92.9 | 20.4 | 76.7 | 16.6 | 82.7 |
| | Text Purification | 91.8 | 42.6 | 53.4 | 33.5 | 62.4 |
| | ROIC-DM | 94.1 | 55.4 | 39.5 | 46.4 | 49.7 |
| MRPC | BERT \dagger | 83.8 | 6.4 | 92.8 | 9.4 | 89.5 |
| | PGD | 84.3 | 6.9 | 92.2 | 11.5 | 82.3 |
| | Free LB | 83.8 | 8.2 | 91.0 | 10.3 | 87.7 |
| | InfoBERT | 87.9 | 13.9 | 84.1 | 17.4 | 78.9 |
| | Text Purification | 82.5 | 32.5 | 54.5 | 28.6 | 61.2 |
| | ROIC-DM | 90.8 | 53.3 | 39.8 | 38.3 | 58.5 |

Table 3: The comparison of the accuracy of ROIC-DM with different pre-trained advisors on the AG NEWS dataset.

| Method | Accuracy % | Method | Accuracy % | Method | Accuracy % |
|-------------------|-------------|-------------------|-------------|-------------------|-------------|
| BERT | 94.0 | DistilBERT | 93.4 | ALBERT | 94.5 |
| ROIC-DM(-advisor) | 91.8 | ROIC-DM(-advisor) | 91.8 | ROIC-DM(-advisor) | 91.8 |
| ROIC-DM | 95.1 | ROIC-DM | 93.8 | ROIC-DM | 95.5 |

at 1e-4. The batch size is 64. For the pre-trained advisor, we directly use the pre-trained version in the toolkit (Morris et al. 2020) provided by Huggingface².

For the baselines, we directly use the hyper-parameter settings in their released codes (PGD³, FreeLB⁴, InforBERT⁵) to generate experimental results. For Text purification (Li, Song, and Qiu 2023b), we re-implement it according to their paper description since there is no public code available.

For the adversarial attack methods, we directly use the corresponding attack’s official codes in Openattack (Zeng et al. 2021) framework.

All the code related to the experiments will be available at <https://github.com/> after the double-blind review.

5.5 The Robustness of ROIC-DM

In this part, we showcase that our ROIC-DM outperforms traditional language models in robustness, even when the latter are equipped with advanced defense methods.

Table 2 presents the experimental results of ROIC-DM compared with BERT equipped with baseline defense meth-

ods. Note that our ROIC-DM also utilizes BERT as the advisor for the fair comparison. According to the results on clean data (*Clean%*), we can observe that ROIC-DM outperforms its advisor, BERT, on all three datasets for at most 2.9 accuracy scores (*Clean%*). Besides, ROIC-DM is also better than BERT with defenses on the clean data.

Table 4: Comparison of the performance between ROIC-DM(-advisor) and ROIC-DM on the AG News test dataset under BERTAttack and Textfooler adversarial attacks. BERT is used as the advisor.

| Method | Clean % | TextFooler | | BERT-Attack | |
|-------------------|-------------|-------------|-------------|-------------|-------------|
| | | Aua % | Suc % | Aua % | Suc % |
| ROIC-DM(-advisor) | 91.8 | 60.3 | 33.4 | 46.4 | 47.8 |
| ROIC-DM | 95.1 | 78.7 | 18.4 | 49.0 | 47.0 |

When attacked by TextFooler, the vanilla BERT’s performance is dramatically dropped on all three datasets. Text Purification achieves the best performance among baselines, but our ROIC-DM still outperforms it by a large margin (e.g., 27.7 Aua% scores on AG NEWS). Moreover, it is worth mentioning that, Text Purification is harmful to the model’s performance on clean data as shown in *Clean%*.

BERT-Attack is stronger than TextFooler and all the mod-

²<https://huggingface.co/textattack>

³https://github.com/MadryLab/mnist_challenge

⁴<https://github.com/zhuochen03/FreeLB>

⁵<https://github.com/AI-secure/InfoBERT>

Table 5: A case study on the AG NEWS dataset

| | Text | BERT | Text Purification | Ours |
|--------------------------------|--|------|-------------------|------|
| Original | though the violence is far less sadistic than usual, the film is typical miike: fast, furious and full of off-the-cuff imaginative flourishes | ✓ | ✓ | ✓ |
| Attacked by BERT-Attack | since the brutal is very smaller sad graphic than usual, the movie is traditional miike way rapid becomes frantic and - of off the cuff creative flourishes. | ✗ | ✗ | ✓ |

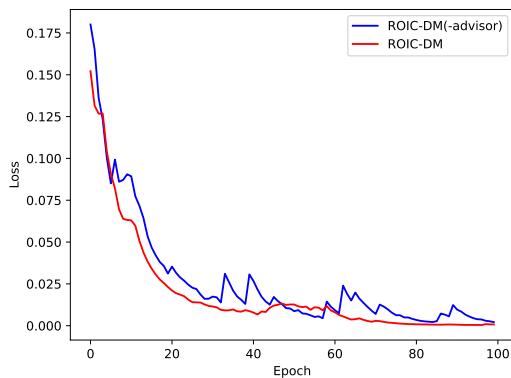


Figure 2: The training loss trend for ROIC-DM and ROIC-DM(-advisor) on the AG NEWS dataset.

els suffer performance deterioration but our ROIC-DM still consistently outperforms all these baseline methods.

5.6 ROIC-DM v.s. Traditional Classifiers

Except for the robustness, another advantage of ROIC-DM is that it can incorporate knowledge from traditional classifiers and achieve better performance.

Table 3 presents the results of ROIC-DM using different pre-trained advisors. Due to the space limitation, we only include the results on AG NEWS. A similar conclusion can be obtained from the other two datasets. As shown in the results, when removing the advisor, ROIC-DM’s performance is slightly worse than traditional classifiers. This may be because these classifiers are based on pre-trained models that have been trained on large-scale datasets with long-term developed model architecture, while our ROIC-DM is a new kind of classification model trained from scratch.

When ROIC-DM uses counterpart advisors, it achieves better performance than its advisors. Specifically, ROIC-DM obtains 1.1, 0.4, and 1.0 higher scores compared to BERT, DistillBERT, and ALBERT, respectively.

5.7 The Impact of Advisor

In this section, we investigate the impacts of advisors for ROIC-DM from the robustness perspective and the training aspect. We compare ROIC-DM with BERT as advisor and

ROIC-DM(-advisor) on the AG News dataset under BertAttack and Textfooler adversarial attacks. As displayed in Table 4, without an advisor, ROIC-DM’s performance declined on both clean and adversarial data. However, if combined the results in Table 4 and Table 2, we can observe that without an advisor, ROIC-DM still exhibits strong robustness, as its performance (60.3 and 46.4 with $AuA\%$ for TextFooler and BERT-Attack respectively) is still much better than BERT with the best baseline defender (51.0 and 44.5 for TextFooler and BERT-Attack respectively). This observation supports our argument that the diffusion model will be more robust than conventional language models.

From the training aspects, Figure 2 shows the training loss curve for ROIC-DM and ROIC-DM(-advisor). The training loss curve for ROIC-DM is smoother than ROIC-DM(-advisor) and the prior’s decreasing speed is also faster than the latter. This phenomenon implies that the advisor can be positive to ROIC-DM’s training process.

5.8 Case Study

Table 5 presents the data sample from the AG NEWS dataset. We utilize BERT-Attack to perturb the original data since it is one of the strongest textual adversarial attacks. We show the comparison with Text Purification as it has the best performance among baselines. For the original text, the vanilla BERT, Text Purification based BERT, and our ROIC-DM can make the correct prediction. After being perturbed, the overall meaning of the sentence is the same as the original text but BERT and Text Purification based BERT cannot make correct predictions. Only our ROIC-DM keeps correct, which indicates its robustness.

6 Conclusion

In this paper, we introduce an innovative model for robust text inference and classification named ROIC-DM, which is built upon the foundational framework of the diffusion model. To enhance the performance of ROIC-DM, we strategically integrate conventional large language models as advisors within the reverse process. The experimental results on three datasets with two competitive textual adversarial attacks indicate that ROIC-DM is more robust to adversarial attacks and can achieve better performance compared with conventional language models.

References

- Alzantot, M.; Sharma, Y.; Elgohary, A.; Ho, B.-J.; Srivastava, M.; and Chang, K.-W. 2018. Generating Natural Language Adversarial Examples. *arXiv:1804.07998*.
- Austin, J.; Johnson, D. D.; Ho, J.; Tarlow, D.; and van den Berg, R. 2021. Structured Denoising Diffusion Models in Discrete State-Spaces. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 17981–17993. Curran Associates, Inc.
- Chen, H.; Dong, Y.; Wang, Z.; Yang, X.; Duan, C.; Su, H.; and Zhu, J. 2023. Robust Classification via a Single Diffusion Model. *arXiv preprint arXiv:2305.15241*.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Dhariwal, P.; and Nichol, A. Q. 2021. Diffusion Models Beat GANs on Image Synthesis. In Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*.
- Dolan, W. B.; and Brockett, C. 2005. Automatically Constructing a Corpus of Sentential Paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.
- Gong, S.; Li, M.; Feng, J.; Wu, Z.; and Kong, L. 2022. Diffuseq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*.
- Gong, S.; Li, M.; Feng, J.; Wu, Z.; and Kong, L. 2023. DiffuSeq: Sequence to Sequence Text Generation with Diffusion Models. In *The Eleventh International Conference on Learning Representations*.
- Gou, J.; Yu, B.; Maybank, S. J.; and Tao, D. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129: 1789–1819.
- Han, X.; Zheng, H.; and Zhou, M. 2022. CARD: Classification and Regression Diffusion Models. In *Thirty-Sixth Conference on Neural Information Processing Systems*.
- Ho, J.; Jain, A.; and Abbeel, P. 2020a. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.
- Ho, J.; Jain, A.; and Abbeel, P. 2020b. Denoising Diffusion Probabilistic Models. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 6840–6851. Curran Associates, Inc.
- Hoogeboom, E.; Nielsen, D.; Jaini, P.; Forré, P.; and Welling, M. 2021. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in Neural Information Processing Systems*, 34: 12454–12465.
- Huang, L.; Qin, J.; Zhou, Y.; Zhu, F.; Liu, L.; and Shao, L. 2023. Normalization techniques in training dnns: Methodology, analysis and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Ishida, T.; Yamane, I.; Sakai, T.; Niu, G.; and Sugiyama, M. 2020. Do We Need Zero Training Loss After Achieving Zero Training Error? In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 4604–4614. PMLR.
- Jin, D.; Jin, Z.; Zhou, J. T.; and Szolovits, P. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 8018–8025.
- Kingma, D.; Salimans, T.; Poole, B.; and Ho, J. 2021. Variational diffusion models. *Advances in neural information processing systems*, 34: 21696–21707.
- Kong, Z.; Ping, W.; Huang, J.; Zhao, K.; and Catanzaro, B. 2021. DiffWave: A Versatile Diffusion Model for Audio Synthesis. In *International Conference on Learning Representations*.
- Li, L.; Ma, R.; Guo, Q.; Xue, X.; and Qiu, X. 2020. Bert-attack: Adversarial attack against bert using bert. *arXiv preprint arXiv:2004.09984*.
- Li, L.; and Qiu, X. 2021. Token-aware virtual adversarial training in natural language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 8410–8418.
- Li, L.; Song, D.; and Qiu, X. 2023a. Text Adversarial Purification as Defense against Adversarial Attacks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 338–350. Toronto, Canada: Association for Computational Linguistics.
- Li, L.; Song, D.; and Qiu, X. 2023b. Text Adversarial Purification as Defense against Adversarial Attacks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 338–350. Toronto, Canada: Association for Computational Linguistics.
- Li, Q.; Peng, H.; Li, J.; Xia, C.; Yang, R.; Sun, L.; Yu, P. S.; and He, L. 2022a. A survey on text classification: From traditional to deep learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(2): 1–41.
- Li, X.; Thickstun, J.; Gulrajani, I.; Liang, P. S.; and Hashimoto, T. B. 2022b. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35: 4328–4343.
- Li, X. L.; Thickstun, J.; Gulrajani, I.; Liang, P.; and Hashimoto, T. B. 2022c. Diffusion-LM Improves Controllable Text Generation. *CoRR*, abs/2205.14217.
- Liu, Q.; Zheng, R.; Rong, B.; Liu, J.; Liu, Z.; Cheng, Z.; Qiao, L.; Gui, T.; Zhang, Q.; and Huang, X.-J. 2022. Flooding-X: Improving BERT’s resistance to adversarial attacks via loss-restricted fine-tuning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5634–5644.
- Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

- Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*.
- Minaee, S.; Kalchbrenner, N.; Cambria, E.; Nikzad, N.; Chenaghlu, M.; and Gao, J. 2021. Deep learning-based text classification: a comprehensive review. *ACM computing surveys (CSUR)*, 54(3): 1–40.
- Moon, H. C.; Joty, S.; Zhao, R.; Thakkar, M.; and Chi, X. 2023. Randomized Smoothing with Masked Inference for Adversarially Robust Text Classifications. *arXiv preprint arXiv:2305.06522*.
- 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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 119–126. Online: Association for Computational Linguistics.
- Mosca, E.; Agarwal, S.; Rando-Ramirez, J.; and Groh, G. 2022. "That Is a Suspicious Reaction!": Interpreting Logits Variation to Detect NLP Adversarial Attacks. *arXiv preprint arXiv:2204.04636*.
- Mozes, M.; Stenetorp, P.; Kleinberg, B.; and Griffin, L. 2021. Frequency-Guided Word Substitutions for Detecting Textual Adversarial Examples. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 171–186. Online: Association for Computational Linguistics.
- Mrkšić, N.; Séaghdha, D. O.; Thomson, B.; Gašić, M.; Rojas-Barahona, L.; Su, P.-H.; Vandyke, D.; Wen, T.-H.; and Young, S. 2016. Counter-fitting word vectors to linguistic constraints. *arXiv preprint arXiv:1603.00892*.
- Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, 8162–8171. PMLR.
- Papernot, N.; McDaniel, P.; Jha, S.; Fredrikson, M.; Celik, Z. B.; and Swami, A. 2016. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, 372–387. IEEE.
- Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical Text-Conditional Image Generation with CLIP Latents. *arXiv:2204.06125*.
- Ren, S.; Deng, Y.; He, K.; and Che, W. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, 1085–1097.
- Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III* 18, 234–241. Springer.
- Samangouei, P.; Kabkab, M.; and Chellappa, R. 2018. Defense-gan: Protecting classifiers against adversarial attacks using generative models. *arXiv preprint arXiv:1805.06605*.
- Savinov, N.; Chung, J.; Binkowski, M.; Elsen, E.; and van den Oord, A. 2022. Step-unrolled Denoising Autoencoders for Text Generation. In *International Conference on Learning Representations*.
- Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Manning, C. D.; Ng, A.; and Potts, C. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 1631–1642. Seattle, Washington, USA: Association for Computational Linguistics.
- Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265. PMLR.
- Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.
- Wang, B.; Wang, S.; Cheng, Y.; Gan, Z.; Jia, R.; Li, B.; and Liu, J. 2021a. Info{BERT}: Improving Robustness of Language Models from An Information Theoretic Perspective. In *International Conference on Learning Representations*.
- Wang, T.; Wang, X.; Qin, Y.; Packer, B.; Li, K.; Chen, J.; Beutel, A.; and Chi, E. 2020. Cat-gen: Improving robustness in nlp models via controlled adversarial text generation. *arXiv preprint arXiv:2010.02338*.
- Wang, X.; Yang, Y.; Deng, Y.; and He, K. 2021b. Adversarial training with fast gradient projection method against synonym substitution based text attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 13997–14005.
- Xiao, Z.; Kreis, K.; and Vahdat, A. 2021. Tackling the generative learning trilemma with denoising diffusion gans. *arXiv preprint arXiv:2112.07804*.
- Zeng, G.; Qi, F.; Zhou, Q.; Zhang, T.; Hou, B.; Zang, Y.; Liu, Z.; and Sun, M. 2021. Openattack: An open-source textual adversarial attack toolkit. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, 363–371.
- Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level Convolutional Networks for Text Classification. In Cortes, C.; Lawrence, N.; Lee, D.; Sugiyama, M.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.
- Zhou, Y.; Jiang, J.-Y.; Chang, K.-W.; and Wang, W. 2019a. Learning to Discriminate Perturbations for Blocking Adversarial Attacks in Text Classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*,

4904–4913. Hong Kong, China: Association for Computational Linguistics.

Zhou, Y.; Jiang, J.-Y.; Chang, K.-W.; and Wang, W. 2019b. Learning to discriminate perturbations for blocking adversarial attacks in text classification. *arXiv preprint arXiv:1909.03084*.

Zhu, C.; Cheng, Y.; Gan, Z.; Sun, S.; Goldstein, T.; and Liu, J. 2020. FreeLB: Enhanced Adversarial Training for Natural Language Understanding. In *International Conference on Learning Representations*.