Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

Wenjie Wang

Wenjie Wang

11/23/2022

Date

Towards the Robustness of Deep Learning Systems Against Adversarial Examples in Sequential Data

By

Wenjie Wang
Doctor of Philosophy

Department of Computer Science

Li Xiong, Ph.D.
Advisor

Jinho Choi, Ph.D.
Committee Member

Vaidy Sunderam, Ph.D.
Committee Member

Ian Molloy, Ph.D.
Committee Member

Accepted:

Kimberly Jacob Arriola, Ph.D, MPH
Dean of the James T. Laney School of Graduate Studies

Date

Towards the Robustness of Deep Learning Systems Against Adversarial Examples in Sequential Data

By

Wenjie Wang
M.S., Emory University
Atlanta, United States, 2019

Advisor: Li Xiong, Ph.D.

An abstract of
A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Department of Computer Science
2022

Abstract

Towards the Robustness of Deep Learning Systems Against
Adversarial Examples in Sequential Data
By Wenjie Wang

Deep learning has achieved state-of-the-art performance in various real-world applications, including computer vision (CV), natural language processing (NLP), speech recognition, and clinical informatics. Although deep learning systems are powerful, they are overly sensitive to perturbation in the input which would not fool a human observer. Recent studies have shown that adversarial examples can be generated by applying small perturbations to the inputs such that the well-trained deep neural networks (DNNs) will misclassify. With the increasing number of safety and security-sensitive applications of deep learning models, the robustness of deep learning models to adversarial inputs has become a crucial topic.

Research on the adversarial examples in computer vision (CV) domains has been well studied. However, the intrinsic difference between image and sequential data has placed great challenges for directly applying adversarial techniques in CV to other application domains such as speech, health informatics, and natural language processing (NLP).

To solve these gaps and challenges, my dissertation research combines multiple studies to improve the robustness of deep learning systems against adversarial examples in sequential inputs. First, we take the NLP and health informatics domains as examples, focusing on understanding the characteristics of these two domains individually and designing empirical adversarial defense methods, which are 1) RADAR, an adversarial detection for EHR data, and 2) MATCH, detecting adversarial examples leveraging the consistency between multiple modalities. Following the empirical defense methods, our next step is exploring certified robustness for sequential inputs which is provable and theory-backed. To this end, 1) We propose WordDP, certified robustness to word substitution attacks in the NLP domain, leveraging the connection of differential privacy and certified robustness. 2) We studied the certified robustness methods to univariant time-series data and propose an adversarial attack in the Wasserstein space which is more appropriate for measuring the in-distinguishability for time-series data.

Towards the Robustness of Deep Learning Systems Against Adversarial Examples in Sequential Data

By

Wenjie Wang
M.S., Emory University
Atlanta, United States, 2019

Advisor: Li Xiong, Ph.D.

A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Department of Computer Science
2022

Acknowledgments

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Deep learning has achieved state-of-the-art performance in various real-world applications, including computer vision (CV) [53], natural language processing (NLP) [107], speech recognition [27] and clinical informatics [85]. Although deep learning systems are powerful, they are overly sensitive to perturbation in the input which would not fool a human observer [91]. Many studies have revealed that adversarial examples can be generated by applying small perturbations to the inputs such that the well-trained deep neural networks (DNNs) will misclassify [14]. With the increasing demand for the safety and security of these applications, how to provide a robust deep learning system that does not overly react to adversarial perturbations has become a crucial topic.

Since the discovery of adversarial examples, many attempts have been made to develop algorithms to generate adversarial examples [77], as well as the countermeasures to study the defense mechanisms [69]. From the attacker's perspective, since 2014 when researchers found that adversarial examples can be easily crafted and mislead the DNNs, many attempts have been made to develop algorithms to generate adversarial examples. Generating adversarial examples can be generalized as an optimization problem: minimize

the perturbation while leading the model to misclassify. Researchers have interpreted this optimization problem from many angles and developed various methods. They can be categorized as white-box attack, black-box attack, and real-world attack. White attack means that the attackers have full access to the model details including the parameters, while in black-box attacks the attackers can only have permission to the model outputs. White-box is more research-oriented which gives us the insight to study the interpretability of DNNs. However, black-box attack is more related to real-world applications. Real-World Attack is under another scenario where inputs are not directly fed into the model but perceived from physical equipment such as cameras and sensors. This research direction is more related to real-world applications and aimed to address the security of DNNs in real-world scenarios.

From the defender's perspective, countermeasures to adversarial examples can be categorized based on their strategies: 1) reactive: detect adversarial examples after DNNs are built such as detection; 2) proactive: make DNNs more robust before adversaries generate adversarial examples such as distillation, adversarial training, and gradient masking. Besides the empirical defense methods, certified robustness is the new direction to improve the robustness of deep learning models that can provide theory-backed and provable defense mechanisms for adversarial attacks. The general attempt is to transform a deterministic base classifier into a probabilistic randomized classifier by adding noise layers.

The reasons that researchers have been motivated to study the potentials of adversarial examples include: 1) Adversarial examples are a security concern for the real world. 2) To test the worst-case robustness of machine learning algorithms. 3) To improve the robustness of DNNs, where robustness refers to ensuring that small changes in the input will not result in dramatic shift of its output.

Gaps and challenges.

Neural networks were first developed and widely applied in computer vision. Therefore, an adversarial example first emerged and was extensively studied in the image domain [14]. However, adversarial threats also exist in other applications where the inputs

are not in the form of an image. Sequential data is a typical example, which includes more common and general data formats such as text, speech, and other temporal data (EHR or trajectory). Little work has been done in the adversarial area for more general data inputs, yet considerable threats still exist.

In the case of text data like spam, adversarial alterations may take the form of substituting synonyms for words that are common in non-spam messages (word substitution attacks) [80]. In the medical domain, studies have shown that adversarial attacks can also be executed successfully against highly accurate medical classifiers [28].

The differences between image and sequential data have placed great challenges to directly apply techniques in the image to sequential inputs, which are listed as follows:

- Perturbation measurement. Image features are continuous while many of the temporal data features are discrete. Generating adversarial examples is an optimization problem that minimizes the perturbation magnitude while maximizing the prediction probability of the target class. To measure the perturbation scales, we can use L_p norm, the pixel-wise distance in the image domain. However, the perturbation measurement for sequential data needs to be defined.

- Imperceptible perturbations. Small perturbations in the image are normally imperceptible to human eyes, but changes in sequential data can be easily perceived. For example, character-level insertion or grammar mistakes in texts are obvious when reading. Significant changes in a patient's vital signal can also be easily detected by healthcare providers. From the aspect of attackers, more imperceptible attack algorithms need to be developed. From the perspective of defenders,

- Data representation and modeling. Image patterns can be learned by the convolutional network. However, learning the representation and capturing the pattern of time-series data is more challenging due to the temporal dependency in addition to the correlations between attributes.

To fill the gap of adversarial examples in sequential data and address the challenges, my dissertation research focuses on improving the robustness of deep learning systems against adversarial examples in sequential inputs, more specifically on two applications, NLP and the health informatics domain.

1.2 Research Contributions

My dissertation research combines multiple studies to improve the robustness of deep learning systems against adversarial examples in sequential inputs. First, we take the NLP and health informatics domains as examples, focusing on understanding the characteristics of these two domains individually and designing empirical adversarial defense methods, which are 1) RADAR, an adversarial detection for EHR data, and 2) MATCH, detecting adversarial examples leveraging the consistency between multiple modalities. Following the empirical defense methods, our next step is exploring certified robustness for sequential inputs which is provable and theory-backed. To this end, 1) We propose WordDP, certified robustness to word substitution attacks in the NLP domain, leveraging the connection of differential privacy and certified robustness. 2) We studied the certified robustness methods to univariant time-series data and propose an adversarial attack in the Wasserstein space which is more appropriate for measuring the in-distinguishability for time series data.

1.2.1 An Adversarial Detection Method for Sequential EHR Data (Sec. 4)

As adversarial threats are important yet under-explored in the clinical domain, in this work, we devoted to enhancing the robustness of deep learning systems on temporal EHR data. We focus on developing a defense method that takes into consideration the unique temporal dependencies of the sequential data. Our goal is to benefit from the autoencoder's [74] reconstruction ability to distinguish adversarial examples and clean examples on EHR data.

A recurrent autoencoder consisting of an encoder and decoder is trained on natural temporal examples and learns the manifold of the natural examples [69]. At the test phase, given an input x, the autoencoder will push the reconstructed output x' closer to the manifold. Adversarially designed examples can be interpreted as out-of-manifold examples that are far away from natural example manifold. This reconstruction error can be used as a major criterion to detect adversarial examples.

To more effectively model the multivariate time series data, we build an autoencoder by integrating attention mechanism [6] with bi-directional LSTM [39] cell to capture both past and future of the current time frame and their interdependence. In addition, to address the sparsity and high dimensionality of EHR data, our method introduces prediction uncertainty of the constructed output as additional detection criteria, besides l_p-norm reconstruction error and prediction divergence of the target classifier.

Contributions. Our key contributions are:

1. We propose RADAR, the first effort to defend adversarial examples on temporal EHR data. While EHR is used for evaluation, RADAR is also applicable to sequential text data.

2. In order to more effectively model the multivariate time series data, we build an autoencoder by integrating attention mechanism [6] with bi-directional LSTM [39] cell to capture both past and future of the current time frame and their interdependence.

3. To address the sparsity and high dimensionality of EHR data, our method introduces prediction uncertainty of the constructed output as an additional detection criteria, besides l_p-norm reconstruction error and prediction divergence of the target classifier.

1.2.2 Detecting Adversarial Examples Leveraging the Consistency between Multiple Modalities (Sec. 5)

This work also focuses on enhancing the robustness of deep learning systems on EHR data. we build an empirical defense mechanism MATCH that uses additional modalities. As real-world data always comes in multiple modalities, we believe that the correlations between different modalities for the same entity can be exploited to defend against attacks. We propose MATCH system to detect whether an input is adversarial, under the circumstance that one modality has been compromised, by measuring the consistency between the compromised modality (clinical notes) and another uncompromised modality (temporal EHR). we conduct a case study on predicting the 30-days readmission risk using an EHR dataset. Experimental results show that MATCH outperforms existing defense techniques in the text domain due to the special characteristics of clinical notes.

Contributions. Our main contributions are as follows:

1. We apply adversarial attack methods to the clinical summaries of electronic health records (EHR) dataset to show the vulnerability of the state-of-the-art clinical deep learning systems.

2. We introduce a novel adversarial example detection method, MATCH, which automatically validates the consistency between multiple modalities in data. This is the first attempt to leverage multi-modality in adversarial research.

3. We conduct experiments to demonstrate the effectiveness of the MATCH detection method. The results validate that they outperform existing state-of-the-art defense methods in the medical domain.

1.2.3 Certified Robustness to word substitution attacks via Differential Privacy (Sec. 6)

By studying the randomized smoothing on the embedding space, we conclude that the existing efforts focused on the image domain cannot be easily adapted to text domain. Rather than derive the certified bound ion the embedding space, it is more reasonable and effective to explore the certified robustness in the word space.

In this work, We aim to improve the robustness of text classification models by explore the certified robustness in the word space. We propose a novel approach WordDP to certified robustness against word substitution attacks in NLP via differential privacy (DP). Differential privacy is a framework that protects the information of individual record in the database by randomizing computations, such that the change of the algorithm's output is bounded when small perturbation is applied on the database.

This stable output guarantee is in parallel with the definition of robustness: ensuring that small changes in the input will not result in dramatic shift of its output. Therefore, we leverage this connection to provide a certified defense mechanism to adversarial examples.

Contributions. Our main contributions are as follows:

1. We propose WordDP to establish the connection between DP and certified robustness for the first time in text classification domain

2. We leverage conceptual exponential mechanism to achieve WordDP and formally prove an L-word bounded certified condition for robustness against word substitution attacks.

3. We develop a simulated exponential mechanism via uniform sampling and weighted averaging to overcome the computation bottleneck of the conceptual exponential mechanism without compromising the certified robustness guarantee

4. Extensive experiments validate that WordDP outperforms existing defense methods

and achieves over 30× efficiency improvement in the inference stage than the state-of-the-art certified robustness mechanism

1.2.4 Certified Robustness to the uni-variant time series data in the Wasserstein space (Sec. 7)

To explore the certified robustness to time series data, we first propose a stronger adversarial attack method to time series data.

The notion of adversarial indistinguishability, in the context of computer vision, is typically bounded by L_∞ or other norms. However these norms are not appropriate for measuring indistinguishiability for time series data. In this work, we propose adversarial examples in the Wasserstein space for time series data for the first time and use Wasserstein distance to bound the perturbation between normal examples and adversarial examples. We introduce Wasserstein projected gradient descent (PGD), an adversarial attack method for perturbing univariant time series data. We leverage the closed form solution of Wasserstein distance in the 1D space and apply projection efficiently with the gradient descent method. Followed by that, we evaluate Wasserstein smoothing [58], a potential certified robustness method to Wasserstein adversarial examples that can provide certified bound in the Wasserstein space.

Contributions. Our Contributions can be summarized as follow:

1. We study adversarial examples in the Wasserstein space for time series data for the first time which better capture the distance and are more natural and imperceptible.

2. We utilize the characteristics of univariant time series data and propose a projected gradient descent attack method which efficiently projects (bounds) adversarial examples in the Wasserstein ball.

3. We develop a two-step projection that first projects an adversarial example to a L_p

norm ball and then use the projected example as the starting point for Wasserstein projection to overcome the computing bottleneck of direct Wasserstein projection.

4. We empirically evaluate the proposed attack on several Electrocardiogram (ECG) datasets in the health care domain. Extensive results demonstrate that the Wasserstein PGD is powerful and can attack most of the target classifiers with a high attack success rate and yield more imperceptible and natural examples than attacks in the Euclidean space.

5. We evaluate Wasserstein smoothing designed for image data as a baseline certified robustness approach against Wasserstein attack which suggest that there is space for stronger defense mechanisms tailored to time series data.

1.3 Organization

The remainder of this thesis is organized as follows. In Section 2, we give a brief overview of the related works. In Section 3, we provide some preliminaries on adversarial examples, certified robustness and differential privacy. Section 4, Section 5 Section 6 and Section 7 introduce our works. Section 8 is the conclusion and includes our future works.

Chapter 2

Related Works

2.1 Adversarial Attack Algorithms

Since 2014 when researches found that adversarial examples can be easily crafted and mislead the DNNs, many attempts have been made to develop algorithms to generate adversarial examples. Generating adversarial examples can be generalized as an optimization problem: minimize the perturbation while leading the model to misclassify. Researchers have interpreted this optimization problem from many angles and developed various methods. They can be categorized as white-box attack, black-box attack, and real-world attack. White-box attack means that the attackers have full access to the model details including the parameters, while in black-box attacks the attackers can only have the permission to the model outputs. White-box is more research-oriented that gives us the insight to study the interpretability of DNNs. However, black-box attack is more related to real-world applications. Real-World Attack is under another scenario where inputs are not directly fed into the model but perceived from physical equipment such as cameras and sensors. This research direction is more related to real-world applications and aimed to address the security of DNNs in the real world scenario.

2.1.1 White-Box Attack

L_BFGS. Adversarial examples were first introduced in paper: Intriguing properties of neural networks [91]. This was the first work that generalized the adversarial example generation task into a constrained optimization problem:

$$Minimize \quad ||r||_2 \quad such \quad that : f(x+r) = l \quad and \quad x + r \in [0,1]^m \tag{2.1}$$

where r indicates the perturbation. They converted this constrained optimization problem into:

$$Minimize \quad c * ||r||_2 + loss(f(x+r), l), \quad such \quad that : x + r \in [0,1]^m \tag{2.2}$$

They approximate adversarial examples by using a box-constrained L-BFGS method to find suitable constant c. The major findings of this work can be summarized as:

- Adversarial examples are easy to find.
- Generated adversarial examples could also be generalized to different models and different training datasets.
- The cause of adversarial examples is more related to the data distribution rather than the model hyperparameters.

FGSM. L-BFGS attack used an expensive linear search method which was time-consuming and impractical. Ian Goodfellow et al [35]proposed a fast Gradient Method to generate adversarial examples by performing one-step gradient update along the direction of the sign of the gradient at each pixel:

$$r = \epsilon sign(\nabla_x J(\theta, x, y)) \tag{2.3}$$

This method is simple and fast but cannot minimize the perturbation. Besides, FGSM is an untargeted attack. Further studies have extended FGSM to targeted attack by slightly modifying the loss function [81, 51]. Other researches intended to iteratively apply FGSM in order to achieve a smaller perturbation[73, 50].

JSMA. JSMA [76] is designed to construct a direct mapping from input perturbation to output perturbation. They use the Jacobian matrix of DNN output w.r.t the input - the forward derivative to represent the adversarial salience map. Based on the pixel-wise salience map, they modify one pixel at each time step. JSMA reduced the number of selected features to perturb, however, this method runs very slow due to its significant computational cost.

Deepfool. Deepfool [73] is proposed to find the closest distance from the original input to the decision boundary of adversarial examples. They performed an iterative line approximation to overcome the nonlinearity of the high dimension. They introduce the basic intuition from a basic affine classifier. The main idea of Deepfool is to reduce the intensity of perturbation.

C&W. C&W [13]is the most powerful attack that requires minimal perturbations to achieve the same attack success rate. They define the objective function to better minimize the distance and penalty term.

2.1.2 Black-Box Attack [75]

Boundary attack. The intuition of boundary attack [11] is to initialize a sample that is already adversarial, performs a random walk along the boundary between the adversarial and the non-adversarial region such that the perturbation is reduced. They initialize the adversarial example by random sampling and then perform random walk by drawing random perturbation from proposal distribution. This iterative process ends when within the maximum number of steps.

ZOO. ZOO [15]is based on C&W attack with two modifications: 1) instead of using the logits layer in the objective function, ZOO uses the model output with a log function; 2) Since black-box attack does not have access to the model parameter and cannot do backward propagation, a zeroth-order optimization is applied to estimate the gradient and use ADAM to do the optimization. As the computation is extremely high, they applied Stochastic coordinate descent that randomly picks pixels to do optimization. ZOO can achieve comparable performance as C&W attack.

2.1.3 Real-World Attack

All works we mentioned above have assumed a threat model in which the adversary can feed data directly into the machine learning classifier. This is not always the case for systems operating in the physical world, for example, those which are using signals from cameras and other sensors as input. In this work [50], they studied how the photo transformation from the camera can affect the attack success rate and compared different image transformation by changing the brightness, contrast, and Gaussian blur.

2.1.4 Non-LP-norm Attack

All the above attack methods we mentioned are L_P-norm attack, which means that the perturbations added to the images are constrained by a L_P-norm distance. Another new merging domain is Non-L_P-norm attack. Non-L_P-norm attack means that the perturbation is not only a pixel-wise perturbation but could be other kinds of modifications such as adding glasses or hat to a figure. Non-Lp-norm attack is often achieved by generative models such as GAN or VAE [7, 84]. There is still a lack of defense techniques for these kinds of adversarial examples.

2.2 Defense Algorithms

Countermeasures to adversarial examples can be categorized based on their strategies: 1) reactive: detect adversarial examples after DNNs are built such as detection; 2) proactive: make DNNs more robust before adversaries generating adversarial examples such as distillation, adversarial training, and gradient masking.

2.2.1 Network Distillation

Network distillation(ND) was originally used in transfer learning to reduce the size of the DNNs. [78] introduced this idea into improving the generalize ability to an unseen dataset and reduce the model to small noise. The intuition of ND is to use the softmax probability of the first model to substitute the ground truth onehot label of the second model. They use a parameter T to control the level of knowledge distillation. To apply this method as a defense method, they use the same architecture for both models, which is basically a two-stage training. From the result, defense distillation can significantly decrease the attack success rate and does not influence model accuracy.

2.2.2 Adversarial Training

Adversarial training refers to training the model with adversarial examples [35]. Adversarial examples are generated in every step of training and inject them into the training set. Adversarial training is proved to be successful in improving the robustness of DNNs. Adversarial training also could provide regularization for DNNs and improve precision as well. However, [92] found that adversarially trained models are more robust to white-box attacks but unsuccessful in black-box attacks. For black-box adversaries, perturbations crafted on an undefended model often transfer to an adversarially trained one. To deal with the perturbation transfer problem, they proposed Ensemble Adversarial Training, a training methodology that incorporates perturbed inputs transferred from other pre-trained

models. Their approach decouples adversarial example generation from the parameters of the trained model and increases the diversity of perturbations seen during training.

2.2.3 Adversarial Detection

One category of the adversarial detector is to train DNN-based binary classifiers as detectors to classify the input data as a clean input or an adversarial example. [34] proposed to train the model using only clean data first and then generate adversarial examples for each clean data. Then, they froze the weight of the first several layers and branch off the main network at some layer and produce the probability of the input being adversarial. [70] directly trained a binary classifier to distinguish adversarial examples from clean examples.

Another direction is to use the difference of the statistical proprieties of adversarial examples to do detection. For example, [26] used higher prediction uncertainty of adversarial example to do detection. Some other works [69, 101, 89] used probability divergence (Jensen– Shannon divergence) as one of its detectors.

However, Carlini and Wagner [12] summarized most of these adversarial detecting methods cannot defend adversarial examples in some cases by slightly change the loss functions. They performed a zero-knowledge attack (adversary does not know the defense), perfect-knowledge attack (knows the defense and model details), and limited-knowledge attack (black-box, know the defense but do not have the detail of both the model and defense) to evaluate ten detectors.

2.2.4 Gradient masking

Gradient masking is a technique that has been applied in many defense methods. Its intuition is to make the model unusable to the attackers, including making the gradient undifferentiable, randomizing the gradients, or leading the vanishing and exploding of the gradient.

However, [5] claimed that obfuscated gradients give a false sense of security. They

proposed new techniques to overcome obfuscated gradients caused by the above three phenomena. From their experiments, seven of eight defenses in ICLR2018 which are based on gradient masking have been broken using their techniques.

2.2.5 State-of-the-art Defense

[67]studied the adversarial robustness of neural networks through the lens of robust optimization. They provided a broad and unifying view on much of the prior work on this topic. They generalized the dense problem into a saddle point optimization problem. This saddle point problem specifies a clear goal that an ideal robust classifier should achieve. They achieve several conclusions: 1) Adversarial training based on the first-order adversary can reliably solve this optimization. 2) Model capacity plays an important role in adversarial robustness.

2.3 Certified Robustness

Most defense mechanisms can be easily broken after introduced. Therefore, it is important to provide Certified Defense that is provable and theory-backed. This has become a future direction to design an adversarial defense mechanism. The intention of Certified robustness is to provide a bound for each image, such that adding any perturbation within a norm bound around the image can be certified to be correctly classified. The evaluation of certified robustness is not only the model performance but also the certified bound that can be achieved. The current state-of-the-art certified defense includes [54, 18].

PixelDP. Lecuyer et al. [54] propose PixelDP to achieve certified robustness by considering an input image as a database in DP parlance and each pixel of the image as each record in DP. PixelDP shows that adding a randomization layer in the model to preserve DP on image pixels guarantees certified robustness of the model against adversarial examples.

Randomized Smoothing. Randomized smoothing is another technique that adds random

noise to the input for achieving certified robustness and has been shown to outperform PixelDP with tighter robustness guarantee. Li et al. [59] derive a certified bound for robustness to adversarial examples using R$\acute{e}$nyi Divergence [33] by adding additive random noise to the input. Cohen et al. [18] leveraged Neyman-Person lemma to analyze the correlation between the highest scored class and the second highest class. Compared to the previous work, they provide a tight certified robustness guarantee for the model. All above-mentioned work are certified within an L_2 radius which means that the adversary cannot alter the prediction within a L_2 unit ball. Lee et al. [56] provide certified robustness for discrete cases where the adversary is L_0 bounded (the number of pixel changes in a figure). Salman et al. [82] further employ adversarial training to improve the certified robustness of models.

2.4 Adversarial Examples in NLP Domain

2.4.1 Attack Algorithms

Attack algorithms in NLP domain refer to generating adversarial examples on input text to make the target model gives the wrong prediction. A well-designed attack algorithm aims to minimize the added perturbation meanwhile fools the model. In the image domain, we can use L_p norm, the pixel-wise distance between adversarial examples and clean examples to measure the scale of the perturbation. In text domain, the evaluation is more complicated because it is import to maintain the grammar correctness, semantic and syntactic validity at the same time. Most popular measurements include: 1) Norm-based measurement on the embedding space. 2) Cosine Similarity between vectors. 3) perplexity to ensure the generated adversarial examples are valid. 4) Word Mover's Distance (WMD) 5) Jaccard similarity coefficient that utilizes intersection and union of word sets of two sentences.

There are many ways to categorize adversarial attacks on texts:

- Model access: White-box vs. Black-White. White-box attack means that the attack-

ers have full access to the model details including the parameters, while in black-box attacks the attackers can only have the permission to the model outputs.

- Semantic group: Character-level vs. Word-Level vs. Sentence-level attack. The attack level clarifies the basic semantic group that the attacker targets.

- Attack strategy: Optimization-based vs. Score-based. For optimization-based attacks, generating adversarial examples is an optimization problem. For score-based approach are those by modifying important words in a sentence.

- Target models: In mast cases the target model is classifiers. However, in text domain, there are some studies focused on generative models such as Neural Machine Translation (NMT).

In this section, I will overview on the state-of-the-art attacks following the category of Optimization-based approach vs. Score-based Approach.

Notations.

F represents the target model that the attackers attack on. It could be a text classifier, or Q&A system or a NMT.

$x = x_1, x_2, ..., x_n$ represents the input sentence where x_i denotes the i_{th} word. x' is the corresponding adversarial example of clean example x.

y represents the ground truth label.

Optimization-based approach

Jacobian-based attack. In [77], the authors adopted the idea of JSMA in image domain to text domain. They compute the Jacobian of the output w.r.t the embedding space of the text input. Iteratively, they select the i_{th} word at each step and perturb this word by adding the sign of the Jacobian matrix of the j_{th} class to i_{th} word. However, the difficulty is that, the perturbed embedding cannot be converted back to an existing word in the vo-

cabulary. Therefore, they further project the perturbed examples onto the closest vectors in the embedding space to get valid embeddings.

iAdv-Text. The idea of iAdv-Text [83] is very similar to [77]. There are two major differences. First, They use the sign of the gradient of the loss w.r.t the embedding of input x. Second, they only restrict the directions of the perturbation in the embedding space toward existing words in the input word embedding space by adding a direction vector. Note, in [77], they project the perturbed examples onto the closest vectors in the embedding space to get valid embeddings. They also involve the generated adversarial examples in the adversarial training. From their experiment, their method can decrease the test error rate of the target model.

Combinatorial Optimization. In this paper [104], the authors point out that the existing attacks are far from perfect, largely because of unsuitable search space reduction and inefficient optimization algorithms are employed. For the search space reduction problem, the authors propose to use Sememe to search for the substitute word instead of the word-embedding based, synonym-based substitution. This is because sememe-based method can obtain the grammaticality and naturality of original input compared to word-embedding based substitution, and have a larger search space compared with synonym-based substitution. For the inefficient optimization problem, the authors propose to apply Particle Swarm Optimization (PSO) to search for the optimal adversarial example in the discrete searching space composed of all substitution words of all the words in a sentence. They conduct exhaustive experiments to evaluate the attack by attacking BiLSTM and BERT on three benchmark datasets. Experimental results demonstrate that their method can achieve much higher attack success rates and crafts more high-quality adversarial examples compared with PWWS and Genetic Attack (will be mentioned in the next section).

Score-based Approach

TextFool. The intuition of TextFool [63] is to firstly identify text items that possess significant contribution to the classification. These important words will generate "hot phrase". For all training samples, the hot phrases obtained previously will be collected. These phrases will be inserted in, deleted from, or modified in the original text to generate adversarial examples. They use the cost gradient to identify the importance of each word. Cost gradient is defined as $\nabla_x J(F, x, y)$. This cost gradient is adopted from FGSM, the most popular attack in the image domain. However, the drawback of this algorithm is that, the process is performed manually. Therefore, they does not report the overall attack success rate on the dataset. They just randomly sample an example from each class and demonstrate the effectiveness of their attack.

AdvGen. AdvGen[17] is proposed to attack on Neural Machine Translation (NMT). For a selected word, they use a language model to measure the probability of substituting the original word to candidate word in the vocabulary. Then, they select the top k candidate word x_i and generate adversarial examples by:

$$\underset{x_i}{\operatorname{argmax}}\, sim(e(x) - e(x_i)), \nabla_{e(x)} J) \tag{2.4}$$

where $sim()$ denotes a similarity measurement and $e(x)$ refers to the word embedding. They further use generated adversarial examples to adversarially train the NMT and improve the robustness of the target model.

MHA. MHA [106] proposed to generate fluent adversarial examples. They utilize the Metropolis- Hastings (M-H) sampling to sample words to be replaced and the words to be replaced with under a pre-distribution $\pi(x)$, and language model to guarantee the fluent of the adversairal example. At each iteration, a proposal $g(x'|x)$ is made to jump from x to x'.

If this proposal pass the accepted rate, which is defined as:

$$\alpha(x, x') = \min\{1, \frac{\pi(x)g(x'|x)}{\pi(x')g(x|x')}\} \tag{2.5}$$

the algorithm jumps to x'. Otherwise, it stays at x.

DeepWordBug. DeepWordBug [31] generate adversarial examples by character transformations. They first identify the most important tokens that change the network output most significantly. Then, random transformation such as character level insertion, deletion and swap are introduced until the prediction label flipped. However, these transformations can introduce typos such as misspelling into the sentence, which can be easily detected and adjusted by autocorrector.

Hotflip. Hotflip [25] uses atomic flip operations to generate adversarial examples. It uses directional vectors to represent the character-level operations such as insertion, deletion and swap. Given a one-hot representation of the input, a character flip in the j_{th} character of the i_{th} word can be represented by a directional vector. Then the best character swap can be found by maximizing the first-order approximation of loss change along the operation vector, which is the gradient of the loss w.r.t the input x times the directional vector. Using the beam search, HotFlip can efficiently find the best directions for multiple flips.

TextBugger. TextBugger [60] can generating adversarial examples in both black-box and white-box settings. In the white-box scenario, Jacobian matrix was used to calculate the importance of each word. In the black-box scenario, they compute the importance of each word by querying the target model with the original sentence and sentence with the word removed. The difference between these queries' results can represent the score of the word in the sentence. After sorting the words based on the importance score, they use character-level and word-level modification to adversarial examples.

Genetic Attack. Genetic attack [3] is inspired by the process of natural selection. At each generation, the algorithm will generate a population by creating a set of distinct modifi-

cations to the previous generation. For each population member, a fitness score will be calculated as the target label prediction probability. If the prediction is equal to the target label, the process ends with successfully generated the adversarial examples. Otherwise, a new child sentence is then synthesized from a pair of parent sentences by independently sampling using a uniform distribution. These child sentences will be used a the parent sentence to generate the population for the next generation. They evaluated the algorithm on two tasks: Sentiment Analysis and Textual Entailment and achieve a high attack success rate.

PWWS. PWWS [80] proposes to use synonyms to substitute the original word to generate adversarial examples, such that guarantee the lexical correctness with little grammatical error and semantic shifting. They first generate synonyms sets for each word in the sentence and select the synonym in the set that can cause the greatest prediction probability change as the candidate substitution for the original word. Then, they incorporate word saliency to determine the replacement order, which is the degree of change in the output classification probability if a word is set to unknown. They report the attack success rate and word replacement rate on three datasets (IMDB, AG'News and Yahoo's Answers) and outperformed four baseline attacks.

TEXTFOOLER. TEXTFOOLER [43] is a very simple but effective attack algorithm. It first ranks the importance of the words in the sentence and iteratively modifies one word at each time followed by the importance rank. They modify the ranked word by generating a candidate word set which consists of synonyms measured by cosine similarity. Then, the candidate word with the least confidence score of label y will be selected as the best replacement word for the original word. This process will terminate until the prediction label being changed. Another contribution of this paper is that they conducted a comprehensive evaluation of three state-of-the-art deep learning models over five popular text classification tasks and two textual entailment tasks, including the Bert model. From the experiments, it achieved the state-of-the-art attack success rate and perturbation rate compared to the Ge-

netic Attack. Besides, they also performed a human evaluation on the grammar correctness. The evaluation scores are close between adversarial examples and clean examples.

2.4.2 Defense Algorithms

The reason we study adversarial examples for NLP system is to improve the security and robustness of these DNNs. However building a defense mechanism is much more challenging than adversarial attacks, which is the same situation in the image domain. This is because the DNN is still not interpretable and the input space is extremely large, making it impossible to build an actual adaptive defense method. In this section, I will discuss two main tracks of defense works: Adversarial training and Adversarial detection.

Adversarial Training. Adversarial training has been widely used in the image domain and has also been adapted to text domain. Adversarial training refers to training the model with adversarial examples [35]. Adversarial examples are generated in every step of training and inject them into the training set. Adversarial training is proved to be successful in improving the robustness of DNNs in the image domain. However, In text domain, adversarial training is not usually useful because of the discrete nature of text data and the different ways of applying perturbations (insertion, deletion and swap) compared with image inputs. Miyato [71] applied the adversarial training to the text domain and achieved the state-of-the-art-performance. Wang [98] proposed Synonyms Encoding Method (SEM), which tried to find a mapping between the word and their synonymous neighbors before the input layer. This can be considered as an adversarial training method via data augmentation. Then this mapping works as an encoder applied to the classifier. The classifier is forced to be smooth in this way. However, SEM can only work for synonym substitution attacks.

Overfitting is also one of the major reason why the adversarial training is sometimes not useful and effective specific to attacks that are used to generate adversarial examples in the training stage.

Adversarial Detection. Most detection methods use spelling check. Gao [31] used Python's Autocorrect 0.3.0 to detect character-level adversarial examples. [60] took advantage of a context-aware spelling check service to do the similar work. However, these detections are not effective for word-level attacks. Zhou [109] proposed a framework learning to discriminate perturbations (DISP), which learns to discriminate against the perturbations and restore the original embeddings.

Others. The above-mentioned methods share the same limitation: very specific to a task or a specific model. In [45], they propose to build a framework that can reuse across multiple tasks, allowing us to only worry about robustness once: during its construction. They introduce robust encodings (RobEn): a simple framework that confers guaranteed robustness, without making compromises on model architecture. The core component of RobEn is an encoding function, which maps sentences to a smaller, discrete space of encodings. Systems using these encodings as a bottleneck confer guaranteed robustness with standard training, and the same encodings can be used across multiple tasks.

2.4.3 Certified Robustness

Certified robustness with IBP. [42] and [38] provided a model that is provably robust to all synonymous substitution attacks. Interval Bound Propagation (IBP) is a simple bounding mechanism that giving the upper and lower bound of an input, we can obtain the upper and lower bound of the output. The intuition of using IBP to provide certified robustness bound is to compute an upper bound on the model's loss in the forward pass when given an adversarially perturbed input. Then we can efficiently train models to minimize this bound via backpropagation. The key idea is to compute upper and lower bounds on the activations in each layer of the network, in terms of bounds computed for previous layers. These bounds propagate through the network in a standard forward pass until we obtain bounds on the final output.

Certified Robustness with Tree Structure. Safer [102] is a certified robust method

based on a new randomized smoothing technique. It replaces the original classifier with a smoothed classifier. Given the synonym sets, they generate the perturbation sets from it. When an input sentence X arrives, they draw perturbed sentences from a given distribution and average their outputs to see with they satisfy a certain condition. If so, this input is treated as certified robust. They reported the certified accuracy which is defined as how many percentages of examples can satisfy the certified condition.

2.5 Adversarial Examples in Clinical Research

Very recently, it has been pointed out that medical machine learning systems may be uniquely susceptible to adversarial examples [28]. Several works studied adversarial examples in medical image models [29, 65, 95, 61].

In 2019, a research team from Harvard and MIT [28] first pointed out that medical machine learning systems may be uniquely susceptible to adversarial examples. They gave a high-level overview of the potential vulnerability in state-of-art medical machine learning systems. Following that, they demonstrated that adversarial examples are capable of manipulating deep learning systems on medical computer vision models [29]. Ma et al. [65] provided a deeper understanding of the nature that medical image DNN models can be more vulnerable to adversarial examples. In 2019, Vatian [95] conducted experiments to show that the degree of manifestation of adversarial examples varies depending on the type of training model.

In 2020, Li et al. [61] generated adversarial 3D MRI brain image targeting conventional deep neural network and hybrid deep model with additional features informed by anatomical context. They found that the hybrid model is much more robust to adversarial perturbations than the conventional deep neural network.

A few works explored the adversarial examples on sequential EHR data. Sun et al. [90] proposed an RNN-based time-preferential minimum attack strategy. Their attack al-

gorithm is similar to the C&W attack in image domain. Although this work demonstrated the vulnerability of the DNN model on EHR data, the main goal of this paper is to identify susceptible locations in EHR data and facilitate physicians. In 2019, Wang et.al [4] proposed a saliency score based adversarial attack on longitudinal EHR data that requires a minimal number of perturbations and minimizes the likelihood of detection. The limitation of this work is that their medical features are binary coded so it is not applicable to continuous features.

2.6 Differential Privacy

Gradient Perturbation. Gradient perturbation is a widely used technique that injects perturbation to the gradient of each parameter to guarantee DP for deep learning models. Song et al. [88] first propose the gradient perturbation method by injecting perturbation to the gradients during parameter updates with stocastic gradient descent (SGD). Bassily et al. [8] improve the gradient perturbation by leveraging privacy amplification via sampling [9] (Lemma II.2 in [8]) and strong composition [24] (Lemma II.3 in [8]) to achieve a tighter bound. Abadi et al. [1] make further improvement by proposing a novel privacy composition tool: moments accountant, which can compute the overall privacy cost during training and achieve a tighter bound. Shokri et al. [86] propose the gradient perturbation method under the distributed learning scenario. Wang et al. [96] replace SGD optimizer used in previous work with stochastic variance-reduced gradient (SVRG) [100] to achieve a faster optimization. However, it requires the loss function l to be convex, G-Lipschitz and β-smooth. Lee et al. [57] and Yu et al. [103] improve the gradient perturbation method by dynamically allocating the privacy budget per iteration and leverage zero-concentrated DP (zCDP) [57] to analyze the privacy cost.

Input Perturbation. Input perturbation is a technique that adds noise to the original training data to achieve DP models. Fukuchi et al. [30] first attempted to use Taylor expansion

to transform input perturbation into gradient perturbation. Although input perturbation framework theoretically guarantees that model trained with perturbed inputs is DP, this work imposes several constraints on the loss function, which cannot be practically applied with deep learning systems. Kang et al. [47] propose an input perturbation that generalizes the constraints on the loss function to less strict conditions. They also take a further step by finding that different training data will affect the model in different ways [46]. However, this work requires a pre-trained model that should also be DP, which also requires privacy budget. In summary, all the above works impose strict constraints on the loss function to analyze DP for input perturbation. These constraints can not be satisfied by typical deep learning models.

Chapter 3

Preliminaries

3.1 Adversarial Word Substitution and Certified Robustness

Adversarial Word Substitution. Consider a sentence of ω words $\mathbf{X} = (x_1, x_2, ..., x_i, ..., x_\omega)$, where each word x_i belongs to a synonym set of $\kappa(i)$ number of synonyms $\mathbf{S}(x_i) = \{x_i^1, x_i^2, ..., x_i^{\kappa(i)}\}$. Following common practice [102], we also assume the synonymous relation is symmetric, such that x_i is in the synonym set of all its synonyms $x_i^2, ..., x_i^{\kappa(i)}$ and $\mathbf{S}(x_i^j) = \mathbf{S}(x_i^k)$ for all $j, k \in [\kappa(i)]$. The synonym set $\mathbf{S}(x_i)$ can be built by following GLOVE [79].

Definition 3.1.1. (L-Adversarial Word Substitution Attack) For an input sentence $\mathbf{X}$, an L-adversarial word substitution attack perturbs the sentence by selecting at most L ($L \leq \omega$) words $x_{\tau_1}, ..., x_{\tau_L}$ and substitutes each selected word x_{τ_i} with one of its synonyms $x'_{\tau_i} \in \mathbf{S}(x_{\tau_i})$. We denote an attacked sentence by $\mathbf{X}'$ and the set of all possible attacked sentences by $\mathscr{S}(L)$.

Certified Robustness. In general, we say a model is robust to adversarial examples when its prediction result is stable when applying small perturbations to the input.

Definition 3.1.2. (Certified Robustness to Word Substitution Attack) Denote a multi-class classification model by $f(X) : \mathcal{X} \mapsto c \in \mathcal{C}$, where c is a label in the possible label set $\mathcal{C} = \{1, ..., C\}$. In general, $f(\mathbf{X})$ outputs a vector of scores $f^y(X) = (f^{y_1}, ..., f^{y_C}) \in \mathcal{Y}$, where $\mathcal{Y} = \{\mathbf{y} : \sum_{i=1}^{C} f^{y_i} = 1, f^{y_i} \in [0, 1]\}$, and $c = \arg\max_{i \in \mathcal{C}} f^{y_i}$. A predictive model $f(X)$ is robust to L-adversarial word substitution attack on input $\mathbf{X}$, if for all $\mathbf{X}' \in \mathscr{S}(L)$, it has $f(X) = f(X')$, which is equivalent to

$$y_c(X') > \max_{i \in \mathcal{C}: i \neq c} y_i(X'). \tag{3.1}$$

In the following, we refer to the above robustness as L-certified robustness for short.

3.2 Differential Privacy and Exponential Mechanism

Differential Privacy. The concept of DP is to prevent the information leakage of an individual record in the database by introducing randomness into the computation. More specifically, DP guarantees the output of a function over two neighbouring databases are indistinguishable.

Definition 3.2.1. (Differential Privacy [22]) A randomized mechanism $\mathcal{A}$ is ϵ-differentially private if, for all neighboring datasets $\mathbf{D} \sim \mathbf{D}'$ that differ in one record or are bounded by certain distance and for all events $\mathbf{O}$ in the output space $\mathcal{O}$ of $\mathcal{A}$, we have

$$\mathbb{P}[\mathcal{A}(\mathbf{D}) \in \mathbf{O}] \leq e^{\epsilon} \mathbb{P}[\mathcal{A}(\mathbf{D}') \in \mathbf{O}]. \tag{3.2}$$

Exponential Mechanism. The exponential mechanism is a commonly utilized DP mechanism in the discrete domain, which consists of the utility score function, sensitivity, and sampling probability distribution as its key ingredients.

Definition 3.2.2. (**Exponential Mechanism [68]**) Denote the score function $u(\mathbf{D}, \mathbf{r}) : \mathcal{D} \times \mathcal{R} \mapsto \mathbb{R}$, which maps each pair of input dataset $\mathbf{D} \sim \mathcal{D}$ and candidate result $\mathbf{r} \in \mathcal{R}$ to a real valued score. Denote the sensitivity by $\Delta_u := \max_{\mathbf{r} \in \mathcal{R}} \max_{\mathbf{D} \sim \mathbf{D}'} |u(\mathbf{D}, \mathbf{r}) - u(\mathbf{D}', \mathbf{r})|$.

The exponential mechanism $\mathcal{M}_E(\mathbf{D}, u, \mathcal{R})$ selects and outputs an element $\mathbf{r} \in \mathcal{R}$ with probability proportional to $e^{\frac{\epsilon u(\mathbf{D},\mathbf{r})}{2\Delta u}}$. The exponential mechanism is ϵ-differentially private.

Chapter 4

RADAR: Recurrent Autoencoder Based Detector for Adversarial Examples on Temporal EHR

4.1 Overview

Electronic Health Record (EHR) is the digital version of a patient's medical history including diagnoses, medications, physician summary and medical image. The automated and routine collection of EHR data not only improves the health care quality but also places great potential in clinical informatics research [85]. Leveraging the information-rich and large volume EHR data, deep learning systems have been applied for assisting medical diagnosis, predicting health trajectories and readmission rates, as well as supporting disease phenotyping [99]. Deep learning models have crucial advantages over the traditional machine learning approaches including the capability of modeling complicated high-dimensional inter-feature relationship within data and capturing the time-series pattern and long-term dependency [90].

However, recent studies show that the statistical boundary of deep learning model is

vulnerable, allowing the creation of adversarial examples by adding imperceptible perturbations on input to mislead the classifier [35]. These adversarial threats are more severe in the medical domain. First, the sparse, noisy and high-dimensional nature of EHR data exposes more vulnerability to potential attackers. Second, some modalities of EHR data such as genetic panels and clinical summary may be generated by a third-party company that has a higher risk being attacked. Finally, medical machine learning systems may be uniquely susceptible to adversarial examples [28] due to high financial interests such as insurance claims.

Despite several attempts on the attack algorithms for temporal EHR data, there is no study on potential defense techniques.In this work, we propose RADAR, a **R**ecurrent **A**utoencoder based **D**etector for **A**dversarial examples on temporal EH**R** data, which is the first effort to defend adversarial examples on temporal EHR data. The intuition is that an autoencoder can learn the manifold of the clean examples. At the test phase, given an input, the autoencoder will reconstruct the input and push the reconstructed output closer to the manifold. As a result, clean examples will have lower reconstruction error since they are closer to the manifold while adversarial examples may have larger error because they have been strategically perturbed. Thus the reconstruction error and additional criteria can be used to detect adversarial examples.

RADAR has two main technical contributions addressing the challenges that are specific to temporal EHR data. First, in order to more effectively model the multivariate time series data, we build an autoencoder by integrating attention mechanism [6] with bi-directional LSTM cell to capture both past and future of the current time frame and their interdependence. By increasing the amount of input information available to the network, RADAR has a higher reconstruction ability which guarantees a higher detectability. Second, to address the sparsity and high dimensionality, besides l_p-norm reconstruction error and prediction divergence of the target classifier between the input and reconstructed output, our method introduces prediction uncertainty of the constructed output as an additional

detection criteria. Our hypothesis is that autoencoder reconstructed output of adversarial examples can result in more uncertainty on the prediction due to its goal of flipping the original class label. This metric focuses on the downstream prediction rather than the data itself thus can overcome the sparsity challenge of EHR data, and provide a critical and complementary criteria for detecting adversarial examples.

RADAR is the first effort to propose defense techniques on temporal EHR data. We evaluate RADAR on a mortality classifier using the MIMIC-III [44] dataset against both existing and our enhanced attacks. Experiments show that RADAR can effectively filter out adversarial examples and significantly improve the target model performance.

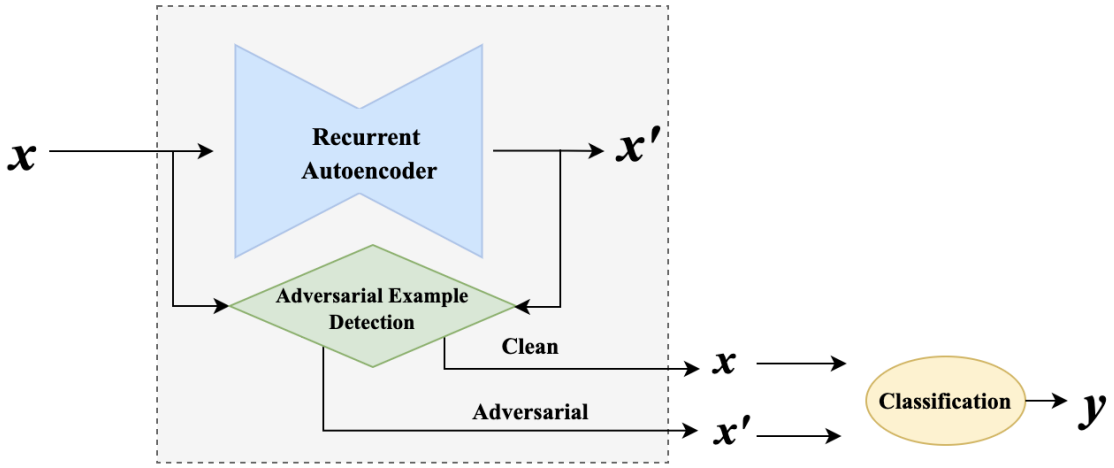


Figure 4.1: The framework of RADAR

4.2 Method

In this section, we first give an overview of the RADAR framework. We then present the details of the recurrent autoencoder architecture, followed by the adversarial example detection criteria. Finally, we present our enhanced attack algorithm.

RADAR is an autoencoder based detector as shown in Figure 4.1. A recurrent autoencoder consisting of encoder and decoder is trained on natural temporal examples and learns the manifold of the natural examples. At the test phase, given an input x, the autoencoder

will push the reconstructed output x' closer to the manifold. Adversarially designed examples can be interpreted as out-of-manifold examples that are far away from natural example manifold. Therefore, when an adversarial example x is fed into a well trained autoencoder, the reconstruction distance between x and x' would be high. The stronger the adversarial perturbation, the larger the reconstruction distance. By contrast, as clean example itself is close to the manifold, the reconstruction distance would be small. Based on a set of carefully designed detection criteria including the reconstruction error, RADAR can detect adversarial examples. As autoencoder can push the reconstructed output closer to the manifold, it can play the role of a reformer. In other words, if an adversarial example is detected, its reconstructed output x' will be treated as reformed output and fed into the classifier.

4.2.1 Recurrent Autoencoder Architecture

Temporal EHR data is multivariate time series data. As our goal is to benefit from the autoencoder's reconstruction ability to distinguish adversarial examples and clean examples, it is crucial to build a recurrent autoencoder structure that is capable of learning both temporal correlations and feature correlations. In this work, we adopt the bidirectional-RNN with attention mechanism for temporal EHR. While the architecture is commonly used, the attention mechanism is first used for EHR data.

Our model is a bidirectional-RNN autoencoder which is shown in Figure 4.2. For the RNN cell, we adopt a stacked LSTM cell designed to capture the long-term dependency and remember information for long periods of time. We feed into the bidirectional-RNN autoencoder with input $x_1, x_2, ..., x_t$ and reversed input $x_t, x_t - 1, ..., x_1$. The forward stacked LSTM of the encoder steps through forward input and encodes the input into hidden states h_{1f} for the first stack and h_{2f} for the second stack. Similarly, the backward stacked LSTM works on the reversed input and generates hidden states h_{1b} and h_{2b}. These hidden states are concatenated and a fully-connected layer is applied to form two fixed-length vectors z_1 and z_2. These two vectors are treated as the initial states of stacked LSTM cells in the

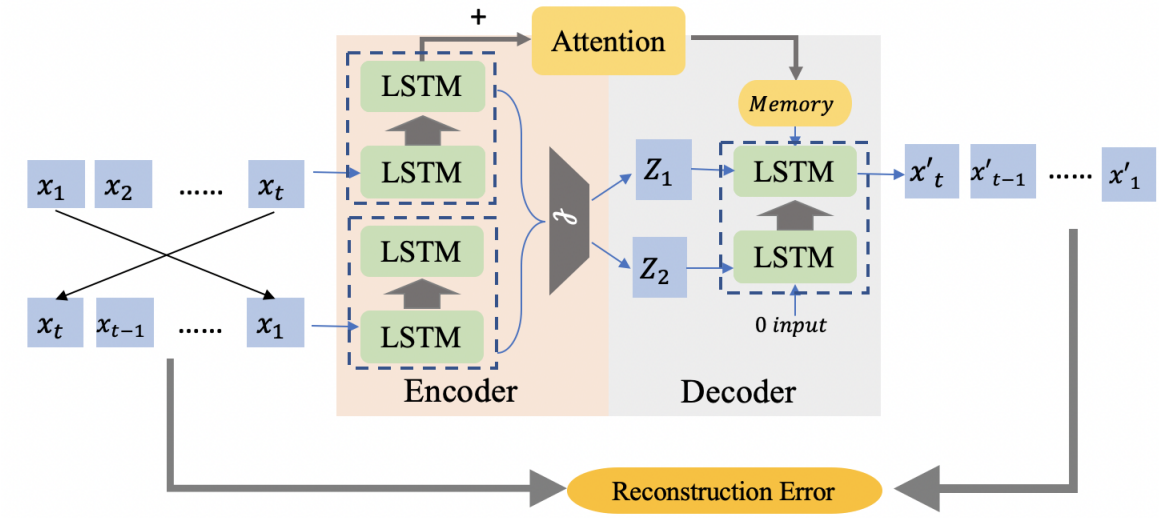


Figure 4.2: BRNN-AE Architecture.

decoder, feeding z_1 to the first stacked LSTM cell and z_2 to the second stacked LSTM cell, which enables the decoder to generate reconstructed output.

One limitation of this encoder and decoder structure is that when the input sequence is long, the fixed-length vector may fail to compress all the information. This issue is significant in temporal EHR data, as the duration of a patient's stay may vary and can be extremely long. To address this, we add the attention mechanism between the encoder and the decoder. Rather than encoding the input sequence into a fixed-length vector, attention forms a weighted sum of each hidden state, referred to as context vectors, allowing the decoder to focus on certain parts of the input when generating its output. In this work, we adopt Bahdanau attention [6] which uses weighted sum of attention weights and encoder hidden states to calculate context vectors and compute the final output of decoder.

We train the autoencoder on clean temporal EHR examples. The loss function is the reconstruction error between the input sequence and the generated output sequence, which is defined as:

$$L(x, x') = \|x, x'\|_2 + L_{reg}(\theta) \tag{4.1}$$

where L_{reg} denotes the L_1 regularization on parameters.

4.2.2 RADAR Detection Criteria

Given an input sequence and the reconstructed sequence, RADAR uses a set of detection criteria to distinguish between a clean example and an adversarial example. Considering the sparsity and high-dimensionality nature of EHR data, our detection criteria includes not only the reconstruction error and prediction divergence that are employed in MagNet, but also the prediction uncertainty of the target classifier.

Reconstruction Error. The reconstruction error between the original and reconstructed sequence is measured by the L_p-norm $L_p(x, x')$. Most commonly used L_p-norm is L_1 norm and L_∞ norm.

Prediction Divergence. In addition to the distance between x and x' in the data space, the prediction divergence between x and x' in their prediction output on the target classifier is also considered. The intuition is that clean examples should have a low divergence. Jensen Shannon Divergence (JSD), a symmetric measurement of the distribution similarity is applied to the target classifier's prediction logits, which is defined as:

$$JSD(l_x||l_{x'}) = \frac{1}{2}KL(l_x||\frac{1}{2}(l_x + l_{x'})) + \frac{1}{2}KL(l_{x'}||\frac{1}{2}(l_x + l_{x'})) \tag{4.2}$$

where l_x and $l_{x'}$ are the classifier's prediction logits of input x and reconstructed output x'. KL denotes the Kullback-Leibler divergence which is a non-symmetric measurement of the difference between two probability distributions. The lower value of JSD, the more similar two distributions are.

Prediction Uncertainty. In addition to the above two measures, we introduce a new criteria based on the prediction uncertainty of the reconstructed output on the target classifier. Our hypothesis is that the reconstructed output of an adversarial examples can result in more uncertainty on the prediction due to its goal of flipping the original class label. Prediction uncertainty focuses on the downstream prediction rather than the data itself thus can overcome the sparsity challenge of EHR data, and provide a critical and complementary criteria

for detecting adversarial examples. Some existing works have proposed methods to measure neural network prediction uncertainty, such as entropy of predictive distribution [62], mutual information and differential entropy [87]. In this work, we use entropy of predictive distribution to reflect uncertainty, which is defined as:

$$Entropy(l_{x'}) = -\sum_{i=1}^{n} s_i log(s_i), \quad \text{where} \quad s_i = \frac{e^{l^i_{x'}}}{\sum_{j=1}^{n} e^{l^j_{x'}}} \tag{4.3}$$

Here, n is the number of prediction classes, s_i is the softmax value of the ith class and $l^i_{x'}$ is the logits value of the ith class of x'.

Given an input x, RADAR detects it as an adversarial example if any one of the above three measurements is greater than a threshold: $M(x, x') > \delta^M$ where M represents reconstruction error, prediction divergence, and prediction uncertainty; and δ^M is the corresponding threshold. In practice, we can choose δ^M to allow a certain percentage of clean examples (e.g. 95%) to pass each criteria. We will study its tradeoff in the experiments section.

4.2.3 Enhanced Attack

In this paper, we also propose an enhanced attack algorithm that addresses the sparsity and high-dimensionality of sequential EHR data to generate more powerful adversarial examples.

Adversarial examples are designed by adding small perturbations to clean examples. For temporal EHR data, a clean example can be represented as $x \in \mathbb{R}^{t \times f} = \{x_1, x_2, ..., x_t\}$, where $x_i \in \mathbb{R}^f$ denotes the f-dimension feature space at the time step i. Given a classifier F, if x_{adv} satisfies that $F(x_{adv}) \neq F(x)$ and $L_p(x, x_{adv}) < C$, we say x_{adv} is the corresponding adversarial example of x. The attack algorithm that we applied to evaluate our proposed defense mechanism is similar to the method proposed in Sun et al. [90]. The purpose of the attack is to maximize the prediction logits on the position of targeted label

(which equals to minimizing the logits on the position of true label) while minimizing the perturbation magnitude, which is formulated as:

$$\underset{x_{adv}}{\operatorname{argmin}} L_y + \alpha L_x, \quad with \tag{4.4}$$

$$L_y = \max\{l(x_{adv})_{y_{true}} - l(x_{adv})_{y_{false}}, -k\} \quad and \quad L_x = ||x_{adv} - x||_p \tag{4.5}$$

where $l(\cdot)_{y_{true}}$ and $l(\cdot)_{y_{false}}$ denotes the logits on the position of true label and false label, as mortality prediction is a binary prediction. A positive value of k ensures a gap between true and adversarial label, which is commonly set to 0. α is a coefficient for the perturbation magnitude.

The L_p-norm is aimed to minimize the EHR location-wise similarity, which does not take into consideration the sparsity and high-dimensionality of sequential EHR data. Therefore, the adversarial examples generated by the attack algorithm can be easily detected by an autoencoder based detection. To craft more powerful adversarial examples, we introduce Gaussian observation [52] into the loss function to force the generated adversarial example to follow the same distribution as clean examples and less detectable by an autoencoder based detection. Gaussian observation is defined as the probability of clean example following the Gaussian distribution with mean as the corresponding adversarial examples and covariance as an identity matrix. Adding the objective of maximizing the Gaussian observation $N(x|x_{adv}, I)$, the attack algorithm can be formulated as a minimization problem:

$$\underset{x_{adv}}{\operatorname{argmin}} L_y + \alpha L_x - \beta N(x|x_{adv}, I) \tag{4.6}$$

where α and β are the coefficients of the two parts of perturbation constraint. For the perturbation magnitude L_x, the L_1 norm induces sparsity on the perturbation and encourages the attack to be more focused on some specific location. By contrast, L_∞ norm encourages the perturbation to be more uniformly distributed with smaller magnitude on each location. In the experiments, we will compare the attack performance of L_1 norm and

L_∞ norm with and without Gaussian observation.

4.3 Experiments

In this section, we will first compare adversarial examples generated by our enhanced attack compared to existing works. Then, we will evaluate the detection performance of RADAR.

4.3.1 Data and Model

Dataset and Model Architecture. MIMIC-III (The Multiparameter Intelligent Monitoring in Intensive Care) dataset [44] is a publicly available clinic dataset containing thousands of de-identified intensive care unit patients' health care records. For mortality prediction, we directly adopt the processed MIMIC-III data from Sun et al. [90] The data contains 3177 positive samples and 30344 negative samples. Each sample consists of 48 timestamps and 19 features at each time step. These 19 variables include vital signs measurements such as heart rate, systolic blood pressure, temperature, and respiratory rate, as well as lab events such as carbon dioxide, calcium, and glucose. Missing features are imputed using average value across all timestamps and outliers are removed and imputed according to interquartile range (IQR) criteria. Then, each sequence is truncated or padded to the same length (48 hours). After imputation and padding, each feature is normalized using min-max normalization.

$$X = \frac{X - min(X)}{max(X) - min(X)} \tag{4.7}$$

The BRNN-AE architecture consists of an encoder with bi-directional two-stacked LSTM cells of units 32 and 64 respectively for both forward and backward LSTM, followed by two fully-connected layers of size 16 and 32 to form two fixed-length vectors as the input to decoder. The decoder consists of an attention layer of size 64 and two-stacked LSTM cells of size 16 and 32.

Pretrained Model Performance. Our target model is a mortality classifier. The network architecture is a simple LSTM of 128 units followed by a fully-connected layer of 32 units and a softmax layer. The 5-fold mean and standard deviation of the model performance is shown in Table 4.1.

Table 4.1: 5-fold cross validation performance of target classifier

Metric	**Accuracy**	**AUC**	**F1**	**Precision**	**Recall**
$Avg \pm STD$	0.894 ± 0.0124	0.812 ± 0.0187	0.603 ± 0.0279	0.536 ± 0.0548	0.702 ± 0.0564

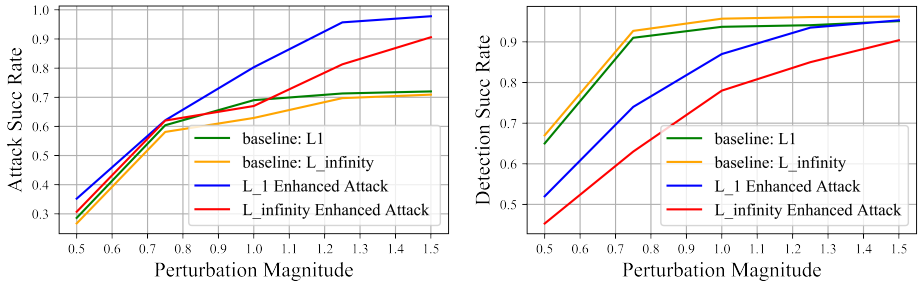
4.3.2 Attack Performance

We use different distance metric to measure the similarity between adversarial examples and clean examples, including L_p-norm and KL divergence. L_p-norm aims to measure EHR location-wise similarity and KL divergence measures the distribution similarity over the whole set of adversarial examples and clean examples. A lower distance means a less detectable attack. In this experiment, the stop criteria for generating each adversarial example is when the prediction label is flipped. Only the successfully attacked examples will be used to calculate the L_p-norm and KL divergence.

Table 4.2 shows the distance metrics of the successfully flipped examples by different attacks. For the baseline attack with no distance optimization, the α and β in equation 4.6 are set to 0. For the L_1-norm attack (Sun et al.[90]) and L_∞-norm attack, α is set to 1 and β is set to 0. The last two columns correspond to our enhanced attacks with Gaussian observation. We observe that the no dist attack (that only aims to flip the label) has the highest distance as expected. Our enhanced attacks based on L_1 and L_∞ have the lowest L_1 and L_∞ distances respectively, and significantly outperform the existing L_1 and L_∞ based attacks. This verifies the benefit of Gaussian observation in our enhanced

Table 4.2: Attack performance comparison

MetricLoss Func	**No dist**	**L_1-norm**	**L_∞-norm**	**L_1-norm enhanced**	**L_∞-norm enhanced**
L_1	3.672	0.815	0.920	**0.524**	0.792
L_∞	0.427	0.138	0.131	0.129	**0.119**
KL	6.521	0.736	0.817	0.811	**0.735**



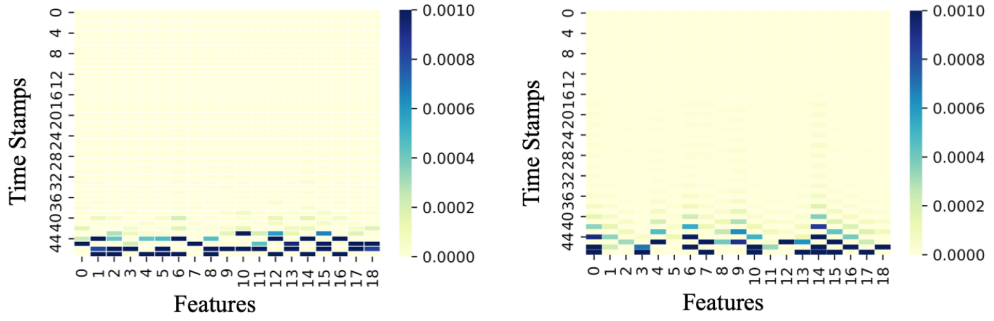
(a) Attack Success Rate (b) Detection Success Rate

Figure 4.3: Comparison between baseline attack and enhanced attack

attacks. By forcing the generated adversarial example to follow the same distribution as clean examples, it not only helps to decrease the KL divergence (in the case of L_∞ based attacks) but more importantly significantly decrease the L_p-norm. The comparison between L_1-norm and L_∞-norm enhanced attacks demonstrates that the L_∞-norm enhanced attack achieves smaller KL divergence, as it encourages the perturbation to be more uniformly distributed with smaller magnitude on each location.

The above results show the comparison of different attack methods for successfully flipped examples. To give a more comprehensive comparison, we also use varying perturbation magnitude as stopping criteria and compare the attack success rate and detection rate (by our detection approach) of different attack methods, which is shown in Figure 4.3. In all cases, our enhanced attacks achieve a higher attack success rate and lower detection rate than the baseline attacks, which confirms the effectiveness of adding Gaussian observation as part of the minimization in the attack.

To illustrate the perturbation introduced by the adversarial examples, we also show the mean perturbation for each of the feature-time points by our enhanced L_∞ attack added to the positive and negative clean examples respectively in Figure 4.4. We observe that most of the perturbation is imposed on the recent time stamps. In addition, interestingly, it requires more perturbation to flip a positive example to negative than vice versa. The reason is that, for an imbalanced dataset, the confidence level is high when classifier predicts an example as positive, which means it requires more perturbation to flip its label.



(a) Positive examples (b) Negative examples

Figure 4.4: Mean perturbation distribution

4.3.3 Detection Performance

In this section, we will first show the impact of varying detection threshold on the clean example pass rate and adversarial example detection rate, and then evaluate the detectability of RADAR in terms of detection rate and the accuracy of the classification model with the detection. We use L_∞-norm enhanced attack and apply varying perturbation bounds of 0.5, 0.75, 1.0, 1.25 and 1.5, which means that the stop criteria for generating each adversarial example is when the perturbation is larger than the perturbation bound.

Selection of Detection Threshold. The threshold of each detection criteria is crucial in the trade-off between the adversarial detection rate and the sacrifice of clean examples, i.e., the true positive and false positive rate. If the threshold is low, it can successfully detect adversarial examples but can also mistakenly filter out clean examples. If the threshold is high, the effectiveness of RADAR will be compromised. Figure 4.5 demonstrates this trade-off by showing the corresponding adversarial detection rate and the clean example pass rate for different thresholds under different perturbation bound. As shown in the figure, a higher perturbation bound results in higher detection rate as expected. When allowing more clean examples to pass, fewer adversarial examples can be detected. The optimal threshold would allow a majority of clean examples to pass while still remaining effective in detecting adversarial examples. In the following experiments, we select the threshold that allows 95% clean example pass rate.

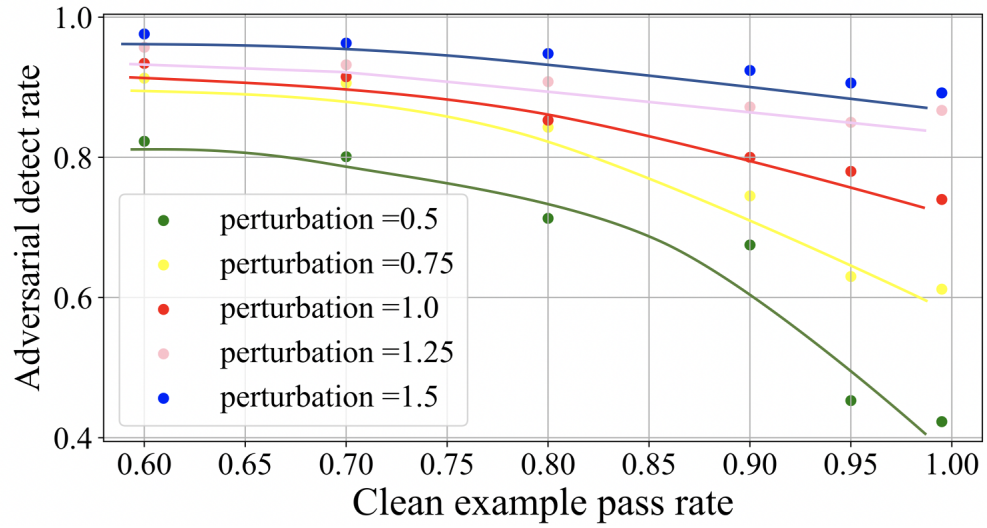
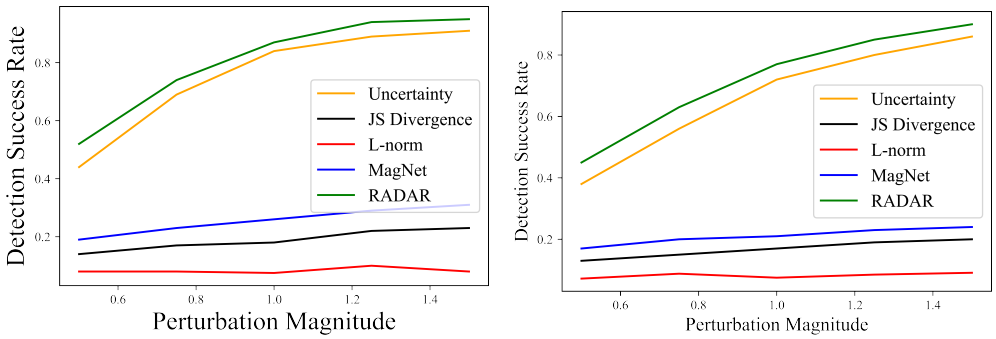


Figure 4.5: The trade-off between adversarial detection rate and clean pass rate



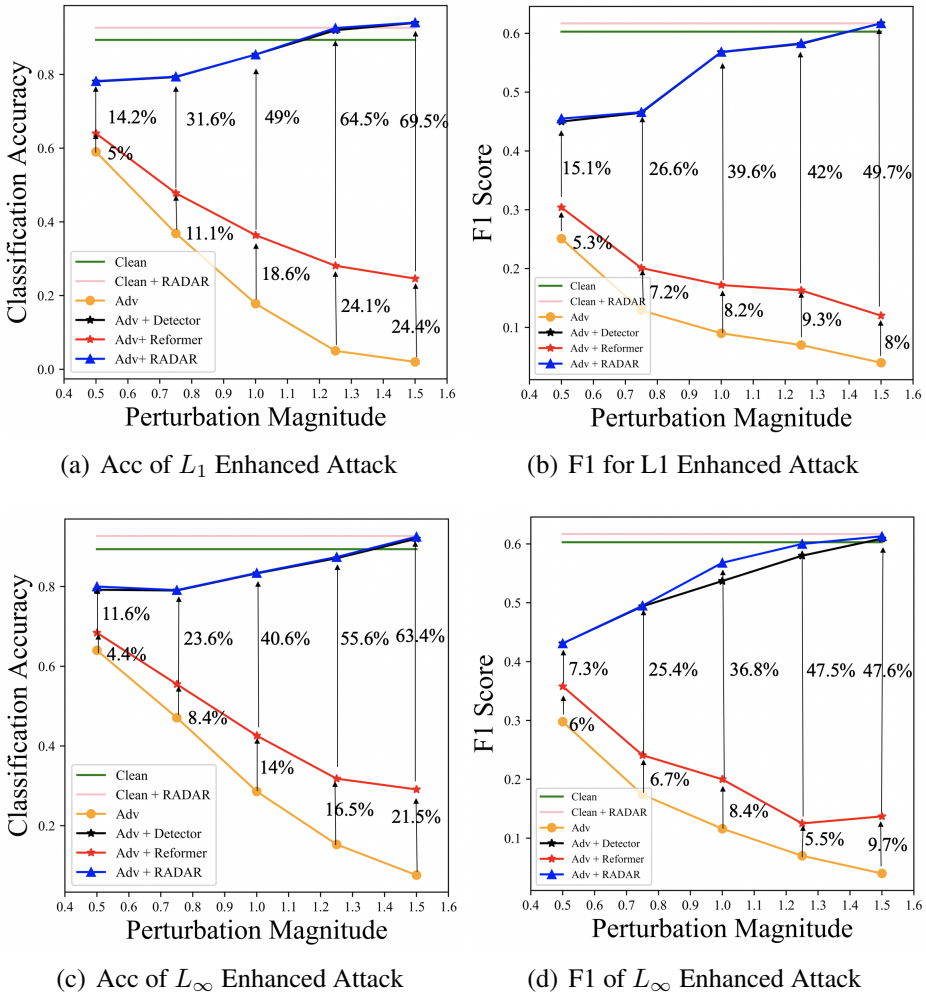
(a) RADAR performance under L_1 enhanced attack

(b) RADAR performance under L_∞ enhanced attack

Figure 4.6: Contribution of each criterion and comparison of RADAR with MagNet

Detection Success Rate. Figure 4.6 shows how much contribution each detection criterion makes to filter adversarial examples. It also compares RADAR (with all three criteria) and the existing MagNet approach (which uses the L-norm and JS Divergence only). With the increase of attack magnitude, the attack detection rate for all criteria/approaches increase as expected. Among the three criteria, our newly introduced prediction uncertainty makes the most and dominating contribution in detecting adversarial examples. As a result, RADAR dramatically outperforms MagNet.

Model Performance. We also evaluate the performance of RADAR in terms of the im-



(a) Acc of L_1 Enhanced Attack

(b) F1 for L1 Enhanced Attack

(c) Acc of L_∞ Enhanced Attack

(d) F1 of L_∞ Enhanced Attack

Figure 4.7: Performance improvement

provement of the target model's prediction accuracy and F1 score. Since any detection mechanism should not sacrifice the accuracy of clean examples, we report the accuracy of clean examples without RADAR (clean) and with RADAR (clean + RADAR). For the purpose of abalation study, we report the accuracy of adversarial examples under different scenarios: 1) when there is no defense (adv), 2) with detector only (adv + detector), 3) with reformer only (adv + reformer), and 4) with both detector and reformer (adv + RADAR). When the RADAR detector is used, if an example is detected as adversarial, we will flip its classification label and softmax output as the final prediction because our task is a binary classification. When only reformer is used, the autoencoder reconstructed output will be

used for classification.

Figure 4.7 shows the target model accuracy and F1 score vs. varying perturbation magnitude for different methods under different attacks. For clean examples, employment of RADAR as a defense mechanism does not affect the prediction performance and can even improve the accuracy. We speculate the reason is that the clean examples that are originally misclassified are usually close to the classification boundary or are outliers, hence may have a high prediction uncertainty or reconstruction error and be detected as adversarial examples. Once they are detected, their prediction will be automatically flipped, which will be correctly classified. Comparing the adversarial examples, only applying RADAR as a reformer can effectively reform the adversarial examples and improve the accuracy and F1 score by more than 10%. When RADAR works as both detector and reformer, it can additionally improve prediction accuracy by more than 60% and even exceeds the accuracy of clean examples. The F1 scores can also be improved by 40% when the perturbation magnitudes are larger than 1.0. The benefit of reformer on top of detector can be noticed in Figure 4.7. With increasing perturbation magnitude, the model accuracy and F1 score of adversarial examples with no defense and reformer drop dramatically due to the increasing attack power. However, interestingly, the model performance with the detection mechanism increases thanks to the increased detection rate as we have observed earlier. These experiments verify the significant improvement of the model performance and the effectiveness of the RADAR mechanism.

Chapter 5

Detecting Adversarial Examples Leveraging the Consistency between Multiple Modalities

5.1 Overview

Most existing defense mechanisms have focused on a single modality of the data. However, EHR data always comes in multiple modalities including diagnoses, medications, physician summaries and medical image, which presents both challenges and opportunities for building more robust defense systems. This is because some modalities are particularly susceptible to adversarial attacks and still lack effective defense mechanisms. For example, the clinical summary is often generated by a third-party dictation system and has a higher risk to be attacked. We believe that the correlations between different modalities for the same entity can be exploited to defend against such attacks, as it is not realistic for an adversary to attack all modalities. Although there are some existing defense techniques in the text domain, these methods cannot be directly applied to clinical texts due to the special characteristics of clinical notes. On one hand, for ordinary texts, spelling or syntax

checks can easily detect adversarial examples generated by introducing misspelled words. However, there are originally plenty of misspelling words or abbreviations in clinical notes, which places challenges to distinguish whether a misspelled word is under attack. One the other hand, data augmentation is another strategy of some adversarial defense techniques in text domain. For example, Synonyms Encoding Method (SEM) [98] is a data preprocessing method that inserts a synonym encoder before the input layers to eliminate adversarial perturbations. However, for clinical notes, a large number of words are proper nouns which makes it difficult to generate synonym set thus challenging to apply such defense. Adversarial training [71] has also been applied to increase the generalization ability of textual deep learning models. However, no research has studied the effectiveness of applying adversarial training in the training of text-based clinical deep learning systems.

In this work, we propose a novel defense method, Multimodal feATure Consistency cHeck (MATCH), against adversarial attacks by utilizing the multimodal properties in the data. We assume that one modality has been compromised, and the MATCH system detects whether an input is adversarial by measuring the consistency between the compromised modality and another uncompromised modality.

To validate our idea, we conduct a case study on predicting the 30-day readmission risk using an EHR dataset. We craft adversarial examples on clinical summary and use the sequential numerical records as another un-attacked modality to detect the adversarial examples. Figure 5.1 depicts the high-level flow of our system.

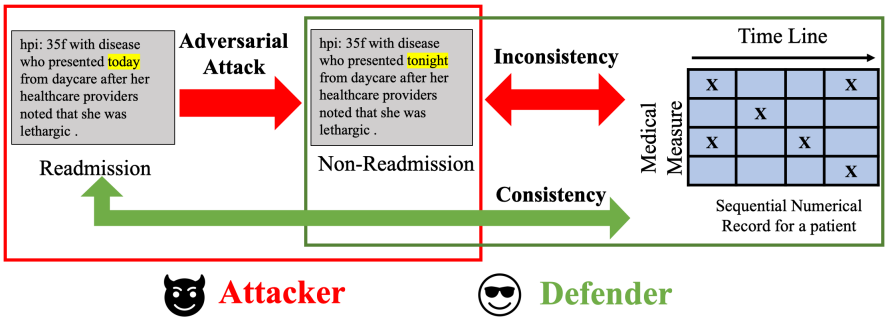


Figure 5.1: Illustration of MATCH: an adversarial attack on the text modal and how MATCH detection finds the inconsistency using the numerical features as another modality.

5.2 Methods

In this section, we will explain our high-level idea and intuitions behind MATCH.

5.2.1 Multi-modality Model Consistency Check

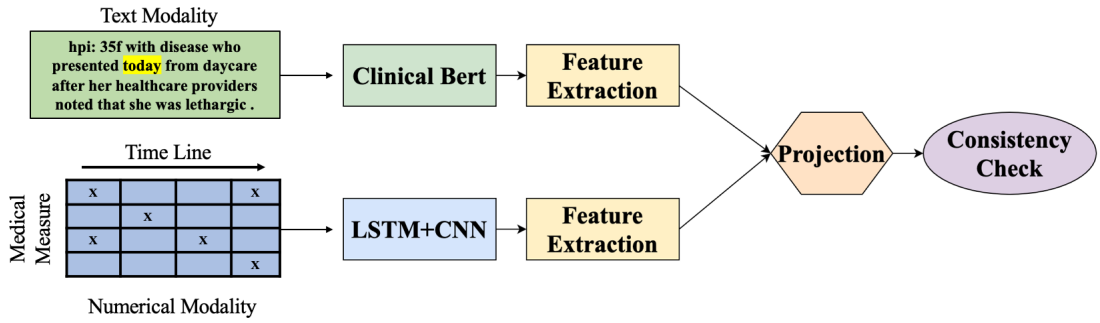


Figure 5.2: Detection Pipeline

System Overview. The main idea of MATCH is to reject adversarial examples if the features from one modality are far away from another un-attacked modality's features. In MATCH, we assume that there is duplicate information in multiple modalities (e.g., 'gray cat' in an image caption and a gray cat in image) and manipulating information can be harder in one modality than another modality. Thus, it is difficult for an attacker to make coherent perturbations across all modalities. In other words, using the gradient to find the steepest change in the decision surface is a common attack strategy, but such a gradient can be drastically different from modality to modality. Moreover, for a certain modality, even if the adversarial and clean examples are close in the input space, their differences would be amplified in the feature space. Therefore, if another un-attacked modality is introduced, the difference between the two modalities can be a criteria to distinguish adversarial and clean examples. Figure 5.2 shows our detection pipeline using text and numerical features. Note that, while we use text and numerical modalities for the experiments, our framework works for any modalities.

We first pre-train two models on two modalities separately. These two models are trained only with clean data, and we use the outputs of their last fully-connected layer

before logits layer as the extracted features. Note that the extracted features from two modalities are in different feature spaces, which requires a "Projection" step to bring the two feature sets into the same feature space. We train a projection model, a fully-connected layer network, for each modality on the clean examples. The objective function of the projection model is:

$$\min_{\theta_1,\theta_2} MSE(p_{\theta_1}(F_1(m_1)) - p_{\theta_2}(F_2(m_2))) \tag{5.1}$$

where m_1 and m_2 represent different modalities. F_i and p_{θ_i} are the feature extractor and the projector of m_i respectively.

Then, a consistency check model is trained only on clean data by minimizing the consistency level between multi-modal features. The consistency level is defined as the L_2 norm of the difference between the projected features from the two modalities. Once all the models are trained, given an input example with two modalities, the system detects it as an adversarial example if the consistency level between two modalities is greater than a threshold δ:

$$||p_{\theta_1}(F_1(m_1)) - p_{\theta_2}(F_2(m_2))||_2 > \delta \tag{5.2}$$

δ is decided based on what percentage of clean examples are allowed to pass MATCH.

Predictive Model and Feature Extractor. For clinical notes, we use pre-trained *Clinical BERT* as our feature extractor. *Clinical BERT* is pre-trained using thr same tasks as [20] and fine-tuned on readmission prediction. *Clinical BERT* also provides a readmission classifier, which is a single layer fully-connected layer. We use this classification representation as the extracted feature.

For sequential numerical records, we adopt the architecture in [105] . However, as our data preprocessing steps and selected features are different, we modify the architecture to optimize the performance. Our architecture (Figure 5.3) employs a stacked-bidirectional-LSTM, followed by a convolutional layer and a fully connected layer. The number of

stacks in stacked-bidirectional-LSTM and the number of convolutional layers, as well as the convolution kernel size are tuned during experiments, which are different from the architecture in [105]. The output of the final layer is used as the extracted features.

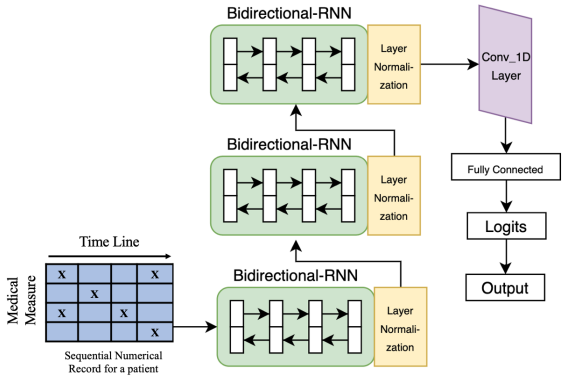


Figure 5.3: Stacked Bidirectional LSTM+CNN architecture

5.3 Experiments

In this section, we first present the attack performance of two text attack algorithms in order to demonstrate the vulnerability of state-of-the-art clinical deep learning systems. Secondly, we evaluate the effectiveness of the MATCH detection method for the readmission classification task using the MIMIC-III data.

5.3.1 Data Preprocessing

Clinical Summary. For the clinical summary, which is the target modality the attacker, we directly use the processed data from [37]. The data contains 34,560 patients with 2,963 positive readmission labels and 48,150 negative labels. In MIMIC-III [44], there are several categories in the clinical notes including ECG summaries, physician notes and discharge summaries. We select the discharge summary as our text modality, as it is most relevant to readmission prediction.

Numerical Data. For the other modality which is used to conduct the consistency check, we use the patents' numeric data in their medical records. We use the patient ID from

the discharge summary to extract the multivariate time series numerical records consisting of 90 continuous features including vital signs such as heart rate and blood pressure as well as other lab measurements. The features are selected based on the frequency of their appearance in all the patients' records.

Then, we apply a standardization for each feature x across all patients and time steps using the following formula: $x = \frac{x-\bar{x}}{std(x)}$. We pad all the sequences to the same length (120 hours before discharge), because this time window is crucial to predict the readmission rate. We ignore all the previous time steps if a patient stayed more than 120 hours and repeat the last time step if a patient's sequence is shorter than 120 hours. We represent the numerical data as a 3-dimensional tensor: patients $\times$ time step (120) $\times$ features (90).

5.3.2 Predictive Model Performance

For the clinical summary data, we use the pre-trained *Clinical BERT*, whose AUC is 0.768. For the numerical data, the performance of our stacked bi-directional LSTM+CNN model produces AUC 0.65. Although the performance of the numerical data is lower than that of *Clinical BERT*, our experiments indicate that it does not affect MATCH's overall performance. The reason is that we only need this prediction model to learn the feature representation. As long as the two models have a comparable performance with each other, the extracted features from the two modalities have a similar representative ability. *Clinical BERT* is also used as the target classifier under attacked.

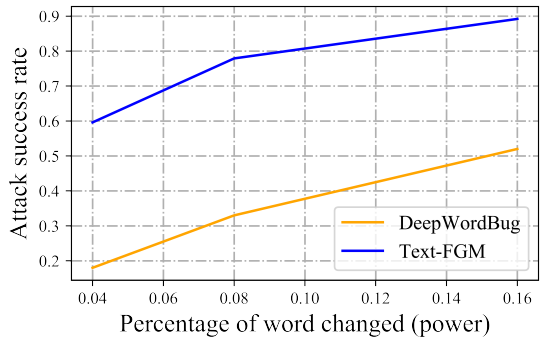


Figure 5.4: Attack Success Rate Comparison between *Text-FGM* and *DeepWordBug*

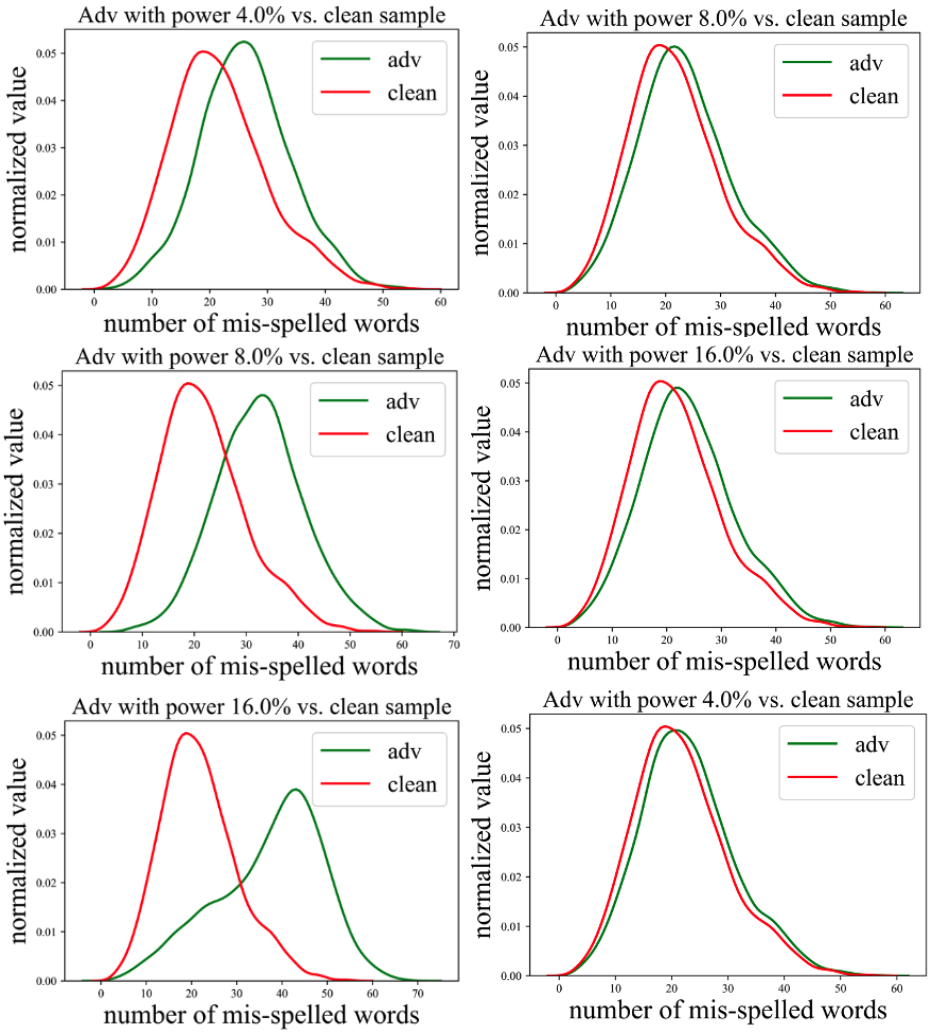
Text-FGM	Clean	DeepWordBug
he will call you to adjust your coumadin (also called warfarin) dose as needed… told by coumadin clinic that you can decrease lab draws . please have result faxed to the coumadin clinic at … discharge disposition: homes with servants venue: vna discharge diagnosis: primary diagnoses: malignant hypertension (hypertensive urgency) acute exacerbation of chronic left brachiocephalic vein occlusion anemia secondary diagnoses… you were confessed and treated for an acute exacerkation of a fungus left brachiocephalic vein …	he will call you to adjust your coumadin (also called warfarin) dose as … told by coumadin clinic that you can decrease lab draws . please have result faxed to the coumadin clinic at . … discharge disposition: home with service facility: vna discharge diagnosis: primary diagnoses: malignant hypertension (hypertensive urgency) acute exacerbation of chronic left brachiocephalic vein occlusion anemia secondary diagnoses…you were admitted and treated for an acute exacerbation of a chronic left brachiocephalic vein …	he will call you to adjust your ocumadin (also called warfarin) dose as needed … told by coumadin clinic that you can decerase lab draws . please have result faxed to the coumadin clinic at . … discharge disposition: home with service afcility: vna discharge diagonsis: primary diagnoses: malignant hypertension (hypertensive urgency) acute xeaecrbation of chronic left brachioecphalic vein occlusion anemia secondary diagnoses…you were admitted and treated for an acute exacerbation of a chronic left brachiocephalic vein …
hx obtained per ed notes and sister . hpi: 35f with disease who presented tonight from daycare after her healthcare providers noted that she was lethargic . they were initially unable to obtain a blood pressure . the patient was noted to have a very rapidly	hx obtained per ed notes and sister . hpi: 35f with disease who presented today from daycare after her healthcare providers noted that she was lethargic . they were initially unable to obtain a blood pressure . the patient was noted to have a very rapid	hx obtained per ed notes and sister . hpi: 35f with disease who presented today from daycare after her healthcare providers noted that she was lethargic . they were initially nuable to obtain a blood perssure . the patient was noted to have a very rapid

Figure 5.5: Example of generated adversarial texts with *Text-FGM* and *DeepWordBug*

5.3.3 Attack Results

In this section, we present the attack performance of two text attack algorithms in order to demonstrate the vulnerability of state-of-the-art clinical deep learning systems. We select two attack algorithms that can present all attack categories we mentioned in the related work: *Text-FGM*,a white-box, semantic attack and *DeepWordBug* a black-box, syntactic attack. Besides, these two attack algorithms will also be used to evaluate the performance of our proposed MATCH, in order to show that MATCH can defense against various kinds of adversarial attacks.

We generate adversarial examples with different attack power levels: 4%, 8%, 16%, which define the maximum percentage of word changes in a text. Then we show the attack success rate under different attack powers, as well as the generated adversarial examples of two attack algorithms. As shown in Figure 5.4, both *Text-FGM* and *DeepWordBug* can produces high attack success rate on the Clinical Bert model. With higher percentage of word changes, the attack success rate also increased for b0th *Text-FGM* and *DeepWordBug*. This is intuitive because as more perturbations being introduced to the input space, the

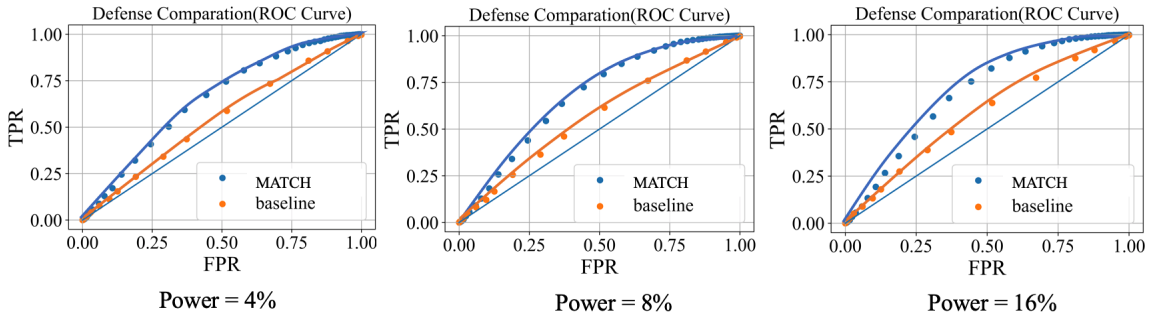


(a) *Text-FGM* (b) *DeepWordBug*

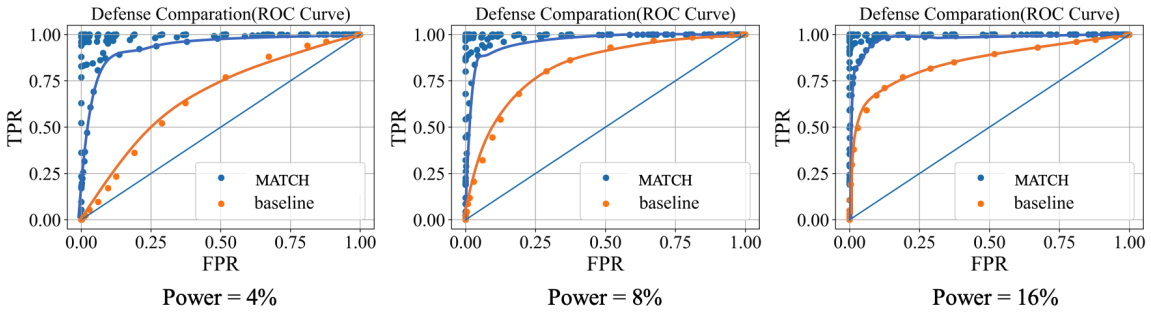
Figure 5.6: Distribution of misspelled words in adversarial /clean text under different attack power

model is more likely to give a wrong prediction. For *Text-FGM*, it achieves almost 80% attack success rate with only 8% of word change, which indicated that the Clinical Bert model are easily fooled and give a wrong prediction. This result indicates the vulnerability of the state-of-the-art text-based medical deep learning systems.

Figure 5.5 shows several examples of our generated adversarial examples from both attack methods compared to the clean examples. The red words represent the changed words in *Text-FGM*, and green words denote the changed words in *DeepWordBug*. It is obvious that even the generated adversarial texts are indistinguishable to human knowledge,



(a) *Text-FGM*



(b) *DeepWordBug*

Figure 5.7: Comparison of the adversarial detection performance between MATCH and misspelling check-based defense.

especially those that generated by *Text-FGM*, but well-trained deep learning models will misclassify.

Besides the attack success rate and the generated adversarial examples, we also present the distribution of the number of misspelled words in the clean and adversarial examples. As shown in Figure 5.6, the number of misspelled word distributions of the clean and the *Text-FGM* adversarial examples are difficult to separate, while the adversarial examples generated by *DeepWordBug* have a large distribution shift compared to that of the clean examples. Further, as the attack power grows, the distribution shift is more distinguishable. This explains why the spelling check service is effective to *DeepWordBug* but not useful for the synonym substitution attack.

5.3.4 Defense Result

In this section, we use *Text-FGM* and *DeepWordBug*, which represent the two types of attacks, semantic vs. syntactic, to evaluate the performance of MATCH

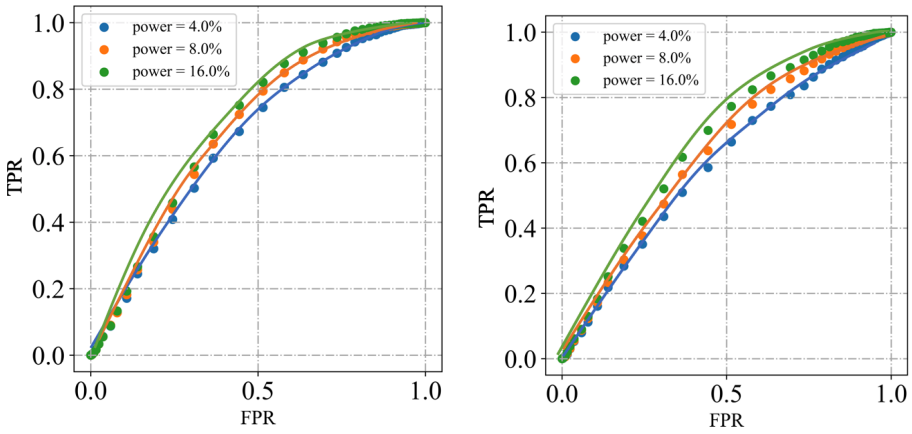
Comparison with Baseline Detection Methods. We use mis-spelling check (*pyspellchecker* form python) as a baseline to compare with MATCH, which is adopted in [31]. As shown in Figure 5.7, we take the attack power (i.e., the percentage of word changes) of 4%, 8% and 16% and use the ROC curve to compare the detection performances between MATCH and the mis-spelling check. ROC curve can represent the correlations between True Positive Rate (TPR) and False Positive Rate (FPR). Here, we want to have higher TPR (adversarial examples can be detected) while achieve lower FPR (clean examples can pass the detector). Given the various detection thresholds δ which allow certain percentage of clean examples to pass detection, these ROC curves illustrate the discriminating ability of MATCH on detecting adversarial examples. Similar to MATCH, we take the number of misspelled words as a threshold and show the discriminating ability given different thresholds. We can note that MATCH significantly outperforms the baseline for both attacks. As mis-spelling check can effectively detect adversarial texts with large misspelling distribution shifts, we take the mis-spelling check as a pre-filter to filter out adversarial examples that are easy to detect. Then, we apply MATCH as a secondary detector. We try different combinations of misspelling word threshold and feature consistency threshold. The blue lines in the charts show the lower boundary of the ROC curves. For *DeepWordBug*, MATCH can achieve close to 100% TPR and 0% FPR. In addition, both MATCH and baseline method works better for *DeepWordBug* because the attack is syntactic, and the examples are easily separable based on the misspelling distribution shifts as observed from Figure 5.6.

Comparison with Adversarial Training. Besides misspelling-check, we also use Adversarial Training (AT) to compare with MATCH on *Text-FGM*. As mentioned in the related work, AT is widly applied in image domain to improve the robustness of DNNs. As our prediction is a binary classification, and MATCH is a detector, in order to compare with

Table 5.1: Comparison of the Adversarial Detection Accuracy

Attack Levels	Clean	No Defense	MATCH	AT
16%	0.672	0.407	0.525	0.435
8%	0.672	0.450	0.523	0.464
4%	0.672	0.483	0.522	0.471

Adversarial Training, we flip the prediction label of examples which are detected as adversarial examples and compare the accuracy with AT. The results in Table 5.1 show that the accuracy of MATCH is much higher than AT and *No Defense*.



(a) *Text-FGM* (b) *DeepWordBug*

Figure 5.8: Detection Result

Impact of attack power. To better illustrate the impact of attack power, we plot the results of varying attack powers in Figure 5.8. To clarify, for *DeepWordBug* we do not include mis-spelling check as a pre-filter, only showing the performance of MATCH. Under *DeepWordBug* with attack power of 16%, MATCH can detect more than 60% of the adversarial examples, while misclassifying 30% of the clean examples as adversarial. Under *Text-FGM* with attack power of 16%, MATCH can detect more than 60% adversarial examples but only 20% of clean examples are mistaken as adversarial. The ROC curve shows that with a higher attack power, MATCH can more easily distinguish adversarial examples from clean examples.

Chapter 6

Certified Robustness to Word Substitution Attack with Differential Privacy

6.1 Overview

In the context of text classification tasks, adversarial examples can be designed by manipulating the word or characters under certain semantic and syntactic constraints [80, 43, 104, 31]. Among all the attack strategies, word substitution attacks, in which attackers attempt to alter the model output by replacing input words with their synonyms, can maximally maintain the naturalness and semantic similarity of the input. Therefore, in this paper, we consider such word substitution attacks and focus on defending against such attacks. Figure 6.1 shows an example of the word substitution attack where the clean input text is changed into adversarial text by substituting input words from a synonym list.

The wrestling between adversarial attacks and defenses has last for years. Most defense mechanisms mentioned above are not robust enough and only empirically works, which means that when a new defense algorithm being proposed, a stronger attack can be

developed to easily break the defense. Therefore, it is important to provide certified defense that is provable and theory-backed [54]. Certified robustness is the new direction to improve the robustness of deep learning models that can provide theory-backed and provable defense mechanism for adversarial attacks. The general attempt is to transform a base classifier into a randomized classifier by adding certain noise layer.

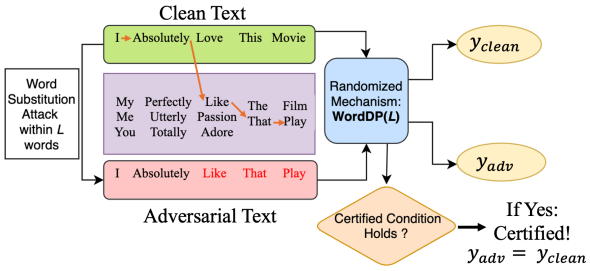


Figure 6.1: Word Substitution Attack and Certified Robustness via WordDP.

Various mechanisms have been developed to defend against adversarial examples in text classification models. However, most of the previous works are only evaluated empirically and have no theoretical analysis or guarantee on the robustness of the methods. Therefore, in this work, we propose a novel approach WordDP to certified robustness against word substitution attacks in text classification via differential privacy (DP) [21]. Figure 6.1 is a high-level illustration. In the inference phase, the input go through a randomized mechanism WordDP. If a clean input satisfies the certification condition of WordDP, its adversarial counterpart is guaranteed to predict the same output label.

DP is a privacy framework that protects the information of individual record in the database by randomized computations, such that the change of the computation output is bounded when small perturbation is applied on the database. This stable output guarantee is in parallel with the definition of robustness: ensuring that small changes in the input will not result in dramatic shift of its output. The idea of providing robustness certification via DP was originally introduced in PixelDP [54] which is specifically designed for norm-bounded adversarial examples in the continuous domain for applications like image classification.

However, it is challenging to directly apply such an idea against word substitution attack, due to the discrete nature of the text input space. Therefore, in this work, we develop

WordDP to achieve the DP and robustness connection in the discrete text space by exploring novel application of the exponential mechanism [68], conventionally utilized to realize DP for answering discrete queries. To achieve this, we present a conceptual certified robustness algorithm that randomly samples word-substituted sentences according to the probability distribution designated by the exponential mechanism and aggregates their inference result as the final classification for the input.

A fundamental barrier limiting the conceptual algorithm from being applied in practice is that the sampling distribution of the exponential mechanism requires an exhaustive enumeration-based sub-step, which needs to repeat the model inference for every neighboring sentences with word substitutions from the input sentence. To overcome this computational difficulty, we develop a practical *simulated exponential mechanism* via uniform sampling and re-weighted averaging, which not only lowers the computational overhead but also ensures uncompromising level of certified robustness.

6.2 Proposed Method

6.2.1 WordDP for Certified Robustness

WordDP. We expand the intuition that DP can be applied to provide certified robustness against textual adversarial examples like word substitution attack by regarding the sentence as a database and each word as a record.

If the randomized predictive model satisfies ϵ-DP during inference, then the output of a potentially adversarial input $X' \in \mathscr{S}(L)$ and the output of the original input X should be indistinguishable. Thus, our proposed approach is to transform a multiclass classification model's prediction score into a randomized ϵ-WordDP score, which is formally defined below.

Definition 6.2.1. (Word Differential Privacy) Consider any input sentence X and its L-word substitution sentence set $\mathscr{S}(L)$. For a randomized function $f_{\mathcal{A}}(X)$, let its prediction

score vector be $y \in \mathcal{Y}$. $f_{\mathcal{A}}(X)$ satisfies ϵ-word differential privacy (WordDP), if it satisfies ϵ-differential privacy for any pair of neighboring sentences $X_1, X_2 \in \mathscr{S}(L)$ and the output space $y \in \mathcal{Y}$.

Remark 1. We stress that WordDP does not seek DP protection for the training dataset as in the conventional privacy area. Instead, it leverages the DP randomness for certified robustness during inference with respect to a testing input.

In practice, for a base model f, a DP mechanism $\mathcal{A}$ will be introduced to randomize it to $f_{\mathcal{A}}$. For an ϵ-WordDP model $f_{\mathcal{A}}$, its expected prediction $\mathbb{E}[f_{\mathcal{A}}(X)]$ is certified robust. Denote the prediction score vector of $\mathbb{E}[f_{\mathcal{A}}(X)]$ by $\mathbb{E}[f^{\mathbf{y}}_{\mathcal{A}}(X)] = (\mathbb{E}[f^{y_1}_{\mathcal{A}}(X)], ..., \mathbb{E}[f^{y_C}_{\mathcal{A}}(X)]) \in \mathcal{Y}$. Lemma 6.2.2 shows $\mathbb{E}[f^{\mathbf{y}}_{\mathcal{A}}(X)]$ satisfies the certified robustness condition in eq.(3.1), based on Lemma 6.2.1 that shows each expected prediction score $\mathbb{E}[f^{y_i}_{\mathcal{A}}(X)]$ is stable.

Lemma 6.2.1. *For an ϵ-WordDP model $f_{\mathcal{A}}$, its prediction score satisfies the relation, $\forall i \in [C]$,*

$$\mathbb{E}[f^{y_i}_{\mathcal{A}}(X_1)] \leq e^{\epsilon}\mathbb{E}[f^{y_i}_{\mathcal{A}}(X_2)], \forall X_1, X_2 \in \mathscr{L}. \tag{6.1}$$

From the above property, we can derive the certified robustness condition to adversarial examples.

Lemma 6.2.2. *For an ϵ-WordDP model $f_{\mathcal{A}}$ and an input sentence X, if there exists a label c such that:*

$$\mathbb{E}(f^{y_c}_{\mathcal{A}}(X)) > e^{2\epsilon} \max_{i \neq c} \mathbb{E}(f^{y_i}_{\mathcal{A}}(X)), \tag{6.2}$$

then the multiclass classification model $f_{\mathcal{A}}$ based on the expected label prediction score vector $\mathbb{E}[f^{\mathbf{y}}_{\mathcal{A}}(\cdot)]$ is certified robust to L-adversary word substitution attack on X.

The proofs of the above two lemmas can be adapted from the pixelDP to WordDP context based on *Lemma 1* and *Proposition 1* in Lecuyer *et al*. lecuyer2019certified.

6.2.2 WordDP with Exponential Mechanism

In this section, we present the conceptual exponential mechanism-based algorithm to achieve WordDP and the certification procedure.

Exponential Mechanism for WordDP. To obtain the DP classifier $f_{\mathcal{A}}$ given the base model f, we introduce the exponential mechanism $\mathcal{M}_E$ as the randomization mechanism $\mathcal{A}$ and define $f_{\mathcal{A}} := f(\mathcal{M}_E)$. Given an input example, the mechanism selects and outputs L-substitution sentences with a probability based on exponential mechanism. It then aggregates the inferences of these samples by an average as the estimated prediction of the input. Figure 6.2 illustrates the algorithm.

Definition 6.2.2. (Exponential Mechanism for WordDP and L-Certified Robustness) Given the base model f, for any input sentence X and potential L-substitution sentence set $\mathscr{S}(L)$, we define the utility score function as:

$$u(\mathscr{S}(L), X') = e^{-\|f^{\mathbf{y}}(X') - f^{\mathbf{y}}(X)\|_1}, \tag{6.3}$$

which associates a utility score to a candidate output $X' \in \mathscr{S}(L)$. The sensitivity of the utility score is $\Delta_u = 1 - e^{-1}$. Then, the exponential mechanism selects and outputs X' with probability $\mathbb{P}_{X'}$

$$\mathbb{P}_{X'} = \frac{1}{\rho} \exp\left(\frac{\epsilon \cdot u(\mathscr{S}(L), X')}{2\Delta u}\right), \tag{6.4}$$

where $\rho = \sum_{i=1}^{|\mathscr{S}(X,L)|} \exp\left(\frac{\epsilon \cdot u(\mathscr{S}(L), X_i')}{2\Delta u}\right)$ is the normalization factor.

Proposition 6.2.1. *The exponential mechanism $\mathcal{M}(E)$ satisfies ϵ-DP. The composition model function $f_{\mathcal{M}_E}(X) := f(\mathcal{M}_E(X))$ is ϵ-DP and its prediction score vector $\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$-based classification is certified robust to L-adversary word substitution attack on X.*

To show $\mathcal{M}_E$ is ϵ-DP, we prove the sensitivity of the utility score (maximum difference between the utility scores given any two neighboring input) Δ_u is indeed $1 - e^{-1}$ and the remaining follows the definition of the exponential mechanism (c.f.Definition 3.2.2). Since $\|f^{\mathbf{y}}(X_i') - f^{\mathbf{y}}(X)\|_1$ is the prediction probability change which is in $[0, 1]$, we have

$u(\mathscr{S}(L), X_i') \in [e^{-1}, 1]$, which leads to $\Delta_u = 1 - e^{-1}$. Next, since $\mathcal{M}_E(X)$ is ϵ-DP, by the post-processing property (i.e., any computation on the output of the DP mechanism remains DP, Proposition 2.1 in [23].), $f_{\mathcal{M}_E}(X)$ is also ϵ-DP. Subsequently, by Lemma 6.2.2, $\mathbb{E}[f_{\mathcal{M}_E}(X)]$ is L-certified robust on X.

Remark 2. 1) The design of the utility function has the intuition that we wish to assign higher probability to sentences that have minimal impact on the prediction score function. 2) The privacy budget ϵ influences whether the sampling probability distribution is flat (lower ϵ) or peaky (greater ϵ). Too small of an ϵ value will clearly affect the prediction accuracy. For certification purpose, according to the certified condition Lemma 6.2.2, too large of an ϵ value will result in none certified, so ϵ can only be searched within a limited range.

Certification Condition. It is a common practice in certified robustness literature to estimate $\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ via Monte Carlo estimation [54, 18] in the form of $\widehat{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$. That is, we repeat the exponential mechanism-based inference to draw n samples of $f^{\mathbf{y}}_{\mathcal{M}_E}(X'_\tau)$, for $\tau \in [n]$ and let $\widehat{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] = \frac{1}{n}\sum_{\tau=1}^{n} f^{\mathbf{y}}_{\mathcal{M}_E}(X'_\tau)$. The estimation error between $\widehat{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ and $\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ can be bounded based on Hoeffding's inequality with probability η, which guarantees that $\widehat{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] \in [\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] - \sqrt{\frac{1}{2n} ln(\frac{2C}{1-\eta})}, \mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] + \sqrt{\frac{1}{2n} ln(\frac{2C}{1-\eta})}] := [\widehat{\mathbb{E}}^{lb}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)], \widehat{\mathbb{E}}^{ub}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]]$. The next proposition shows that the inference based on the estimated $\widehat{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ (as versus $\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$) can still ensure certified robustness.

Proposition 6.2.2. *Under the same condition with Proposition 6.2.1, if there exists a label c such that*

$$\widehat{\mathbb{E}}^{lb}[f^{\mathbf{y}_c}_{\mathcal{M}_E}(X)] > e^{2\epsilon} \max_{i \neq c} \widehat{\mathbb{E}}^{ub}[f^{\mathbf{y}_i}_{\mathcal{M}_E}(X)], \tag{6.5}$$

the prediction score vector $\widehat{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$-based classification is certified robust with probability η to L-adversary word substitution attack on X.

6.2.3 Simulated Exponential Mechanism

Simulated Exponential Mechanism. The conceptual exponential mechanism in Definition 6.2.2 is computationally impractical. The bottleneck is the need to enumerate the entire $\mathscr{S}(L)$ in order to calculate the probability distribution of $\mathbb{P}_{X'}$ for each $X' \in \mathscr{S}(L)$ and the normalization factor ρ, which essentially requires us to perform inference for $\mathscr{S}(L) \gg n$ times (n is the number of samples) for certifying a single input sentence X.

In the following, we show that we can significantly reduce the computation cost by sampling via a simulated exponential mechanism, which suffices to sample n candidate $L-$substitution sentences and calculate only n times, i.e., the same repetitions as the Monte Carlo estimation. The key insight is based on the different purpose of applying the exponential mechanism between the conventional scenario for achieving DP and our certified robustness scenario. For the former, in order to ensure DP of the final output $f_{\mathcal{M}_E}(X'_\tau)$, the intermediate X'_τ is forced to satisfy DP, i.e., drawn from the exact probability distribution designated by the exponential mechanism. For the latter, while the derivation of the certified robustness relied on the randomness of DP and the exponential mechanism, we do not actually require the DP of the intermediate X'_τ. As a result, it allows us to sample X'_τ from other simpler distributions without calculating the probability distribution of the exponential mechanism, as long as the alternative approach can obtain the equivalent $\widehat{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{A}}(X)]$ for robustness certification.

We develop a simulated exponential mechanism via *uniform sampling and re-weighted average prediction score calculation*. Figure 6.2 shows the simulated mechanism in contrast to the conceptual mechanism. In detail, we sample from $\mathscr{S}(L)$ with uniform probability, which can be efficiently implemented without generating $\mathscr{S}(L)$. Denoting a sample by X'_τ, we calculate its scaled exponential mechanism probability by

$$\bar{\mathbb{P}}_{X'_\tau} = \exp\left(\frac{\epsilon \cdot u(\mathscr{S}(L), X'_\tau)}{2\Delta u}\right), \tag{6.6}$$

which can be obtained via a single inference on X'_τ and the inference on X due to the

omission of the normalization factor ρ that requires the entire $\mathscr{S}(L)$. The inference on X only needs to be computed once and shared by all n Monte Carlo repetitions. Such uniform sampling and scaled probability calculation is repeated for n times, which requires only $n+1$ inferences. Finally, we use the following re-weighted average prediction score (weighted by the scaled exponential mechanism probability) for certified robust prediction,

$$\bar{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] = \sum_{\tau=1}^{n} \bar{\mathbb{P}}_{X'_\tau} \cdot f^{\mathbf{y}}_{\mathcal{M}_E}(X'_\tau). \tag{6.7}$$

The following theorem shows that $\bar{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$-based prediction guarantees certified robustnessand the conceptual exponential mechanism-based inference in Proposition 6.2.2 is certified robust provided $\bar{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ is so.

Theorem 6.2.1. *For any input X, let $\bar{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ be calculated by eq.(6.7). Denote $xbar\mathbb{E}^{lb}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ and $xbar\mathbb{E}^{ub}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ be η-confidence lower and upper bounds, respectively, i.e., $xbar\mathbb{E}^{lb}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] = xbar\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] - \sqrt{\frac{1}{2n} ln(\frac{2C}{1-\eta})}$ and $xbar\mathbb{E}^{ub}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] = xbar\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)] + \sqrt{\frac{1}{2n} ln(\frac{2C}{1-\eta})}$. If there exists a label c such that*

$$\bar{\mathbb{E}}^{lb}[f^{y_c}_{\mathcal{M}_E}(X)] > e^{2\epsilon} \max_{i \neq c} \bar{\mathbb{E}}^{ub}[f^{y_i}_{\mathcal{M}_E}(X)], \tag{6.8}$$

the prediction score vector $xbar\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$-based classification is certified robust with probability η to L-adversary word substitution attack on X.

The proof of Theorem 6.2.1 requires the following lemma, which is adapted from Lemma 6.2.1 from the accurate expectation of $\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ to the simulated expectation $xbar\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$. We stress that during both proofs, we do not use the DP property of $xbar\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(\cdot)]$, but only its equivalent relation to $\widehat{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(\cdot)]$.

Lemma 6.2.3. *For any label $i \in [C]$ and any $X_1, X_2 \in \mathscr{S}(L)$, let $xbar\mathbb{E}[f^{\mathbf{y}}_{\mathcal{M}_E}(X)]$ be computed by eq.(6.7). Then, we have*

$$\bar{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X_1)] \leq e^{\epsilon} \bar{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X_2)]. \tag{6.9}$$

First, we notice that for any $X' \in \mathscr{S}(L)$, it has $\bar{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X')] = \frac{\rho}{|\mathscr{S}(L)|} \widehat{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X')]$ by $xbar\mathbb{P}[X'] = \rho\mathbb{P}[X']$ and the uniform sampling probability $\frac{1}{|\mathscr{S}(L)|}$. Second, since

$\widehat{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X')]$ is ϵ-WordDP , we can show that it satisfies Lemma 6.2.1 by switching $\mathbb{E}[f^{y_i}_{\mathcal{M}_E}(\cdot)]$ there to $\widehat{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(\cdot)]$ here. It follows that:

$$\begin{aligned}\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X_1)] &= \widehat{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X_1)] \cdot (\frac{\rho}{|\mathscr{S}(L)|}) \\ &\leq e^{\epsilon}\widehat{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X_2)] \cdot (\frac{\rho}{|\mathscr{S}(L)|}) = e^{\epsilon}\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X_2)],\end{aligned}$$

which proves the lemma.

Proof. **(Proof of Theorem 6.2.1)** For any $X^{'} \in \mathscr{S}(L)$, by eq.(6.9), we have

as well as

$$\begin{aligned}&\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X')] \leq e^{\epsilon}\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X)] \leq e^{\epsilon}\max_{i\neq c}\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X)] \\ &\leq e^{\epsilon}\max_{i\neq c}(\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X)] + \sqrt{\frac{1}{2n}ln(\frac{2C}{1-\eta})}) \\ &= e^{\epsilon}\max_{i\neq c}\overline{\mathbb{E}}^{ub}[f^{y_i}_{\mathcal{M}_E}(X)].\end{aligned}$$

Equipped with the above two relations, we can prove the claim in Theorem 6.2.1. We show that $\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X)]$ is certified robust for any $X' \in \mathscr{S}(L)$, as follows,

$$\begin{aligned}&\overline{\mathbb{E}}[f^{y_c}_{\mathcal{M}_E}(X')] > \overline{\mathbb{E}}^{lb}[f^{y_c}_{\mathcal{M}_E}(X)]/e^{\epsilon} \\ &> e^{\epsilon}\max_{i\neq c}\overline{\mathbb{E}}^{ub}[f^{y_i}_{\mathcal{M}_E}(X)] > e^{\epsilon}\max_{i\neq c}\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X')].\end{aligned} \tag{6.10}$$

which is $\overline{\mathbb{E}}[f^{y_c}_{\mathcal{M}_E}(X')] > e^{2\epsilon}\max_{i\neq c}\overline{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X)]$.

For completeness, we can also show that the certified robustness of $\overline{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{A}}(X)]$ implies the certified robustness of $\widehat{\mathbb{E}}[f^{\mathbf{y}}_{\mathcal{A}}(X)]$:

$$
\begin{aligned}
&\widehat{\mathbb{E}}[f^{y_c}_{\mathcal{M}_E}(X')] = (\frac{|\mathscr{S}(L)|}{\rho}) \cdot \bar{\mathbb{E}}[f^{y_c}_{\mathcal{M}_E}(X')] \\
&> (\frac{|\mathscr{S}(L)|}{\rho}) \bar{\mathbb{E}}^{lb}[f^{y_c}_{\mathcal{M}_E}(X)]/e^{\epsilon} \\
&> (\frac{|\mathscr{S}(L)|}{\rho}) e^{\epsilon} \max_{i \neq c} \bar{\mathbb{E}}^{ub}[f^{y_i}_{\mathcal{M}_E}(X)] \\
&> (\frac{|\mathscr{S}(L)|}{\rho}) \max_{i \neq c} \bar{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X')] = \max_{i \neq c} \widehat{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X')],
\end{aligned}
\tag{6.11}
$$

which proves $\widehat{\mathbb{E}}[f^{y_c}_{\mathcal{M}_E}(X')] > \max_{i \neq c} \widehat{\mathbb{E}}[f^{y_i}_{\mathcal{M}_E}(X')]$.

□

Finally, we use Hoeffding's inequality to bound the error in $\widehat{\mathbb{E}}[\mathcal{A}(X')]$. With η-confidence error the lower bound and upper bound of $\widehat{\mathbb{E}}[\mathcal{A}(X')]$ can be formulated as:

$$
\begin{aligned}
&\hat{\mathbb{E}}^{lb}[\mathcal{A}(x')] = \hat{\mathbb{E}}[\mathcal{A}(x')] - \sqrt{\frac{1}{2N} ln(\frac{2n}{1-\eta})}, \\
&\hat{\mathbb{E}}^{up}[\mathcal{A}(x')] = \hat{\mathbb{E}}[\mathcal{A}(x')] + \sqrt{\frac{1}{2N} ln(\frac{2n}{1-\eta})}, \\
&\text{such that} \quad \hat{\mathbb{E}}^{lb}[\mathcal{A}(x')] \leq \hat{\mathbb{E}}[\mathcal{A}(x')] \leq \hat{\mathbb{E}}^{up}[\mathcal{A}(x')]
\end{aligned}
\tag{6.12}
$$

where n represents number of classes.

Till now, our certified robustness condition can be generalized as:

Randomized mechanism $\mathcal{A}$ is jointly decided by L and ϵ, which defines the size of the perturbation set, and the exponential probability respectively. During experiments, WordDP not only generate aggregated perdition results but also indicate whether a given input satisfy above certification criteria under certain neighbour constraint L and ϵ.

Training procedure. To achieve a better certification result, we involve randomness in the training stage, which is also adopted by almost all certified robustness approaches. To do so, we use the data augmentation strategy that utilizes the perturbed sentences for training, i.e., $X' \in \mathscr{S}(L) \setminus X$ given the original training sample X. In practice, we first train the model without data augmentation for several epochs to achieve a reasonable performance, followed by training with perturbed X'. For each training data point, we randomly draw

one neighbour sentence during training (as opposed to multiple draws during certified inference).

6.2.4 Extension of WordDP: Empirical defense method

Beside the certified mechanism with WordDP, we extend the idea of WordDP to an empirical defense method.

The intuition on designing the utility function as formular 6.4 is to assign higher probability to sentence that have minimal impact to the scoring function, which are those examples that are close to the input data point. Therefore, this utility function favors the clean examples that may achieve better performance on clean examples, as data points that closer to clean examples are more likely to be correctly classified. The exponential mechanism based on this utility function can also benefit the accuracy of adversarial examples based on the assumption that majority of examples around adversarial examples should be be benign examples.

To achieve a better result on adversarial examples, we design a utility function that is very similar to formular 6.4 but favors adversarial examples more, which is defined as:

$$u(X_i', X) = e^{\alpha||f(X_i')-f(X)||_1} \tag{6.13}$$

Note that this utility function assigns higher probability to sentence that have greater impact to the scoring function, which are the examples that far away from the input data. Therefore, adversarial example are more likely to be correctly classified according to the aggregated prediction.

The sensitivity Δu also slightly different from the sensitivity in WordDP. Here, $u(X_i', X) \in [1, e^1]$ as $||f(X_i') - f(X)||_1$ is the output probability changes which is bounded by 0 and 1. Therefore, Δu equals to $e^1 - 1$.

In exponential mechanism, the privacy budget ϵ influence whether probability vector is

smoother or rougher. Generally, smaller ϵ just makes all the sampling probability be the same, and generates a flatter distribution. On the contrast, greater ϵ just "leaks more information", and generates rougher distribution that is close to the real distribution(sentence with have higher utility score are more likely to be sampled). For certification purpose, according to the certified condition, too large ϵ will result in 0-certified, such that ϵ should be searched within a certain range.

However, if we only want to design a empirical defense without considering the certified robustness, we can assign larger ϵ to achieve a identical exponential distribution of the sampling distribution based on the utility function 6.13. In our empirical experiments, we set ϵ to be 10, in order to achieve a higher conventional accuracy on the clean example. The trade-off between the accuracy on clean example and adversarial examples will be discussed in section 6.3.2.

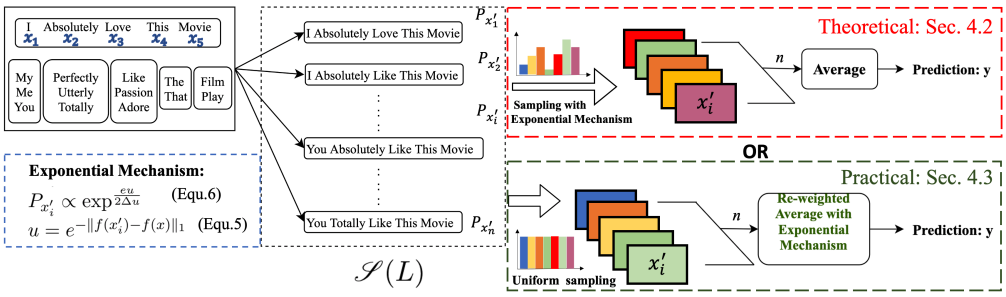


Figure 6.2: with Exponential Mechanism.

6.3 Experiments

We evaluate WordDP on two classification datasets: Internet Movie Database (IMDB) [66] and AG News corpus (AGNews) [107]. IMDB is a binary sentiment classification dataset containing 50000 movie reviews. AGNews includes 30,000 news articles categorized into four classes. The target model architecture we select is a single-layer LSTM model with size of 128. We use Global Vectors for Word Representation (GloVe) [79] for word embedding. The LSTM model achieves 88.4% and 91.8% clean accuracy on IMDB and AGNews,

respectively. We use PWWS [80] to generate adversarial examples on the test dataset. PWWS is a state-of-the-art attack method which uses WordNet to build synonym set and incorporates word saliency to replace selected named entities (NEs) with their synonyms in order to flip the prediction.

6.3.1 Evaluation Metrics and Baselines

We use four metrics to evaluate the effectiveness of WordDP: certified ratio, certified accuracy, conditional accuracy, and conventional accuracy. **Certified Ratio** represents the fraction of testing set that the prediction satisfies the certification criteria: $\frac{\sum_{t=1}^{T} certifiedCheck(\mathbf{X}_t,L,\epsilon)}{T}$, where $certifiedCheck$ returns 1 if Theorem 6.2.1 is satisfied and T is the size of the test dataset. **Certified accuracy (CertAcc)** denotes the fraction of the clean testing set on which the predictions are both correct and satisfy the certification criteria. This is a standard metric to evaluate certified robust model [54]. Formally, it is defined as:

$\frac{\sum_{t=1}^{T} certifiedCheck(\mathbf{X}_t,L,\epsilon)\&corrClass(\mathbf{X}_t,L,\epsilon)}{T}$, where $corrClass$ returns 1 if the classification output is correct. When the accuracy of a model is close to 100%, certified accuracy largely reflects certified ratio. **Conventional accuracy (ConvAcc)** is defined as the fraction of testing set that is correctly classified, $\frac{\sum_{t=1}^{T} corrClass(\mathbf{X}_t,L,\epsilon)}{T}$, which is a standard metric to evaluate any deep learning systems. Note that the input $\mathbf{X}_t$ can be both adversarial or clean inputs. We use this metric to evaluate how WordDP empirically works on adversarial examples.

Besides the above standard metrics, we introduce a new accuracy metric called **Conditional accuracy (CondAcc)** to evaluate the following: when a clean input $\mathbf{X}_t$ is certified within bound L, whether its corresponding L-word substitution adversarial example $\mathbf{X}_t^{adv}$ is indeed correctly classified. The CondAcc can be formulated as:
$\frac{\sum_{t=1}^{T} certifiedCheck(\mathbf{X}_t,L,\epsilon)\&corrClass(\mathbf{X}_t^{adv},L,\epsilon)}{\sum_{t=1}^{T} certifiedCheck(X_t,L,\epsilon)}$.

While certified accuracy is typically evaluated on clean inputs in the literature to show

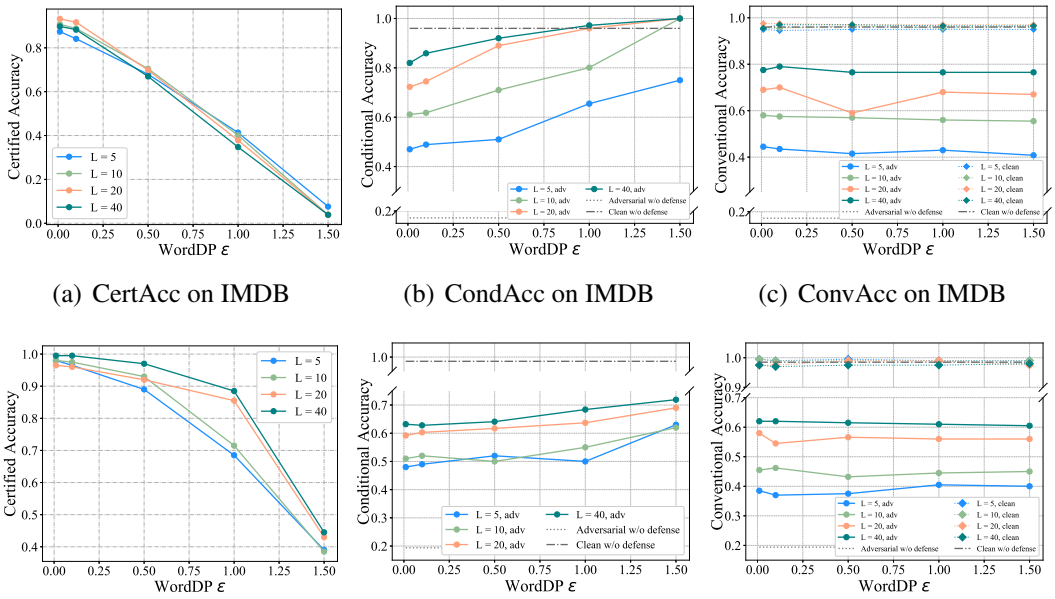
the certified robustness property, conditional accuracy is evaluated on adversarial inputs and provides an informative measure of the classification result of adversarial examples when its counterpart clean input can be certified. This metric is aligned with the definition and purpose of certified robustness. Ideally, if a clean example is successfully certified, adversarial examples created from this clean example should have the same prediction. Therefore, the accuracy of adversarial examples is influenced by the ConvAcc of clean examples.

Comparison Methods. We compare WordDP with the state-of-the-art certified robust method SAFER [102] for text classification. We note that SAFER only reports certified accuracy, without accuracy on adversarial examples. To conduct a fair comparison with WordDP, we rerun SAFER on the adversarial examples and report the comparison in CertAcc and CondAcc. Besides SAFER, we also compare the ConvAcc on adversarial examples with two state-of-the-art defense methods, i.e., IBP [42] and DNE [108], which do not provide certified robustness guarantee. Thus, their defense may be broken by more powerful word substitution attacks in the future.

6.3.2 Certified Results

Certified Accuracy. Figure 6.3 presents the CertAcc, CondAcc and ConvAcc under different ϵ and L, respectively. Each line in the figures represents a certified bound L, which allows L number of words to be substituted. The first row is the results on IMDB, and the second row is on AGNews.

Figures 6.3(a) and 6.3(d) show the certified accuracy on the two datasets. Since the conventional accuracy on the clean examples of our mechanisms is close to 100% (as shown in Figures 6.3(c) and 6.3(f)), the certified accuracy mainly reflects the certified ratio (which we skip in the results). As shown, higher ϵ can result in lower CertAcc. This is intuitive as the condition in Theorem 6.2.1 is more difficult to satisfy when given higher epsilon, i.e. weaker requirement of indistinguishability of the output, hence results in lower certified



(a) CertAcc on IMDB (b) CondAcc on IMDB (c) ConvAcc on IMDB

(d) CertAcc on AGNews (e) CondAcc on AGNews (f) ConvAcc on AGNews

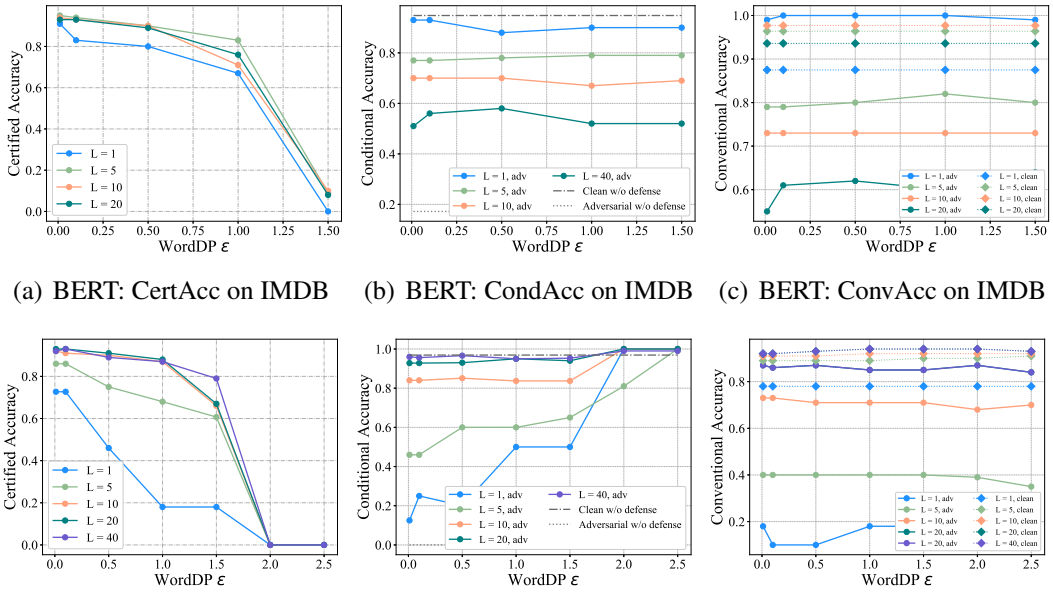
Figure 6.3: Certified Accuracy, Conditional Accuracy and Conventional Accuracy on IMDB and AGNews

ratio. As illustrated in 6.3(a), when ϵ is around 1.5, the mechanism will approach 0 certified ratio. This indicates that ϵ can only be searched within a limited range.

Comparing each line in 6.3(a) and 6.3(d), we note that greater L results in higher CertAcc in most cases for the AGNews dataset. This can be explained by the fact that a greater L means more word substitutions and randomness are introduced in WordDP, making it easier to ensure the indistinguishability of the output, and hence a higher certified ratio.

Accuracy on Adversarial Examples. Figures 6.3(b), 6.3(e), 6.3(c) and 6.3(f) present CondAcc and ConvAcc of the two datasets on adversarial examples, respectively. Note that we only test the adversarial examples that are within the L bound. We also show the CondAcc and ConvAcc for both clean and adversarial examples without any defense mechanisms as a reference. In addition, we show ConvAcc of WordDP with varying parameters on clean examples to show the impact of the mechanism on clean examples.

As shown in the figures, WordDP achieves significantly higher accuracy on adversarial examples compared to no defense while maintaining the close to 100% accuracy on clean



(a) BERT: CertAcc on IMDB (b) BERT: CondAcc on IMDB (c) BERT: ConvAcc on IMDB

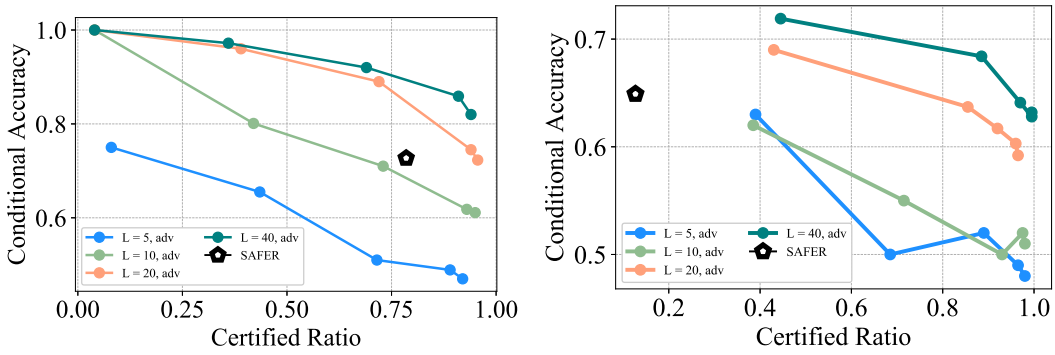
(d) BERT: CertAcc on AGNews (e) BERT: CondAcc on AG-News (f) BERT: ConvAcc on AG-News

Figure 6.4: BERT Results of Certified Accuracy, Conditional Accuracy and Conventional Accuracy on IMDB and AGNews

examples. Conditional accuracy is higher than conventional accuracy as expected, since it is computed only on those adversarial examples with a certified counterpart clean example. Besides, we can observe that with higher ϵ, higher CondAcc on adversarial examples can be achieved. This is because less randomness is introduced in the inference.

In addition, by comparing different L bound under the same ϵ, larger L can yield more accuracy improvement on adversarial examples but less on clean examples. Intuitively, using the aggregated prediction of more distant neighbouring sentences (higher L) can benefit adversarial examples more than clean examples.

WordDP performance on BERT model. Besides evaluating the performance on LSTM model, we also consider the most state-of-the-art language model: BERT [19].As shown in Figure 6.4, the performance of WordDP on Bert has the similar trend of that on LSTM model, which can be summarized as: 1) higher ϵ can result in lower CertAcc. 2) greater L results in higher CertAcc. 3) WordDP achieves significantly higher accuracy on adversarial examples compared to no defense. 4) larger L can yield more accuracy improvement on



(a) IMDB (b) AGNews

Figure 6.5: Certified Ration vs. Conditional Accuracy

adversarial examples. 5) higher higher ϵ can result in lower CondAcc.

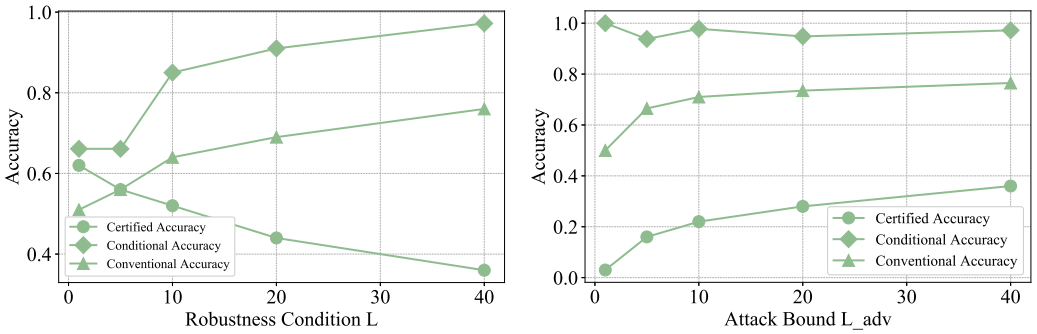
Note that this certified accuracy and conditional accuracy is achieved under the circumstance that WordDP is not integrated in the training process. As Bert is a pre-trained model and the last a few layers are always fine-tuned according to different tasks, it is unrealistic to add a noise layer and incorporate WordDP into the whole training process. Although WordDP can provide considerable empirical accuracy improvement, other Certified Robustness mechanism such as Safer provide near-zero certified accuracy. Therefore, lager space for improvement has been left to study the certified mechanism for pre-trained models. This topic is critical as in both industry and academia, increasingly large model size are used to achieve state-of-the-art performance, and the pre-train plus fine-tune schema has become the mainstream, which makes it crucial to explore certified robustness for pre-trained models.

Trader-off between Certified Ratio and CondAcc. We can see that ϵ has an opposite impact on certified accuracy (certified ratio) and CondAcc, we present the trade-off between the certified ratio and CondAcc of WordDP in Figure 6.5 in comparison with the baseline method SAFER. Ideally, we want both high certified ratio and high condAcc to contribute to overall high accuracy. The black dot represents the baseline SAFER, since the neighbouring sentence generating method of SAFER does not depend on L or ϵ. As illustrated

	ADV	IBP	DNE	SAFER	WordDP
IMDB	0.172	0.722	0.823	0.727	**0.972**
AGNews	0.194	0.823	**0.909**	0.647	0.719

Table 6.1: Empirical comparison on accuracy

on these two datasets, with $L = 20$ and $L = 40$,WordDP can dominate SAFER and achieve a much better performance in both certified ratio and condAcc.



(a) Fixed attack power 40

(b) Fixed defense power 40

Figure 6.6: The trend on accuracy under different defense and attack power

Relation between certified bound L and adversarial attack power L_{adv}. Figure 6.6 presents the three accuracy metrics under different attack power and defense power. In Figure 6.6(a), we fix the attack power L_{adv} to 40, which means allowing less than 40 word substitutions, and adjust the WordDP defense power by using different certified bound L. As discussed in Section 6.2, certified bound L determines the size of neighbouring set. Greater L leads to higher randomness and thus can benefit the CondAcc and ConvAcc on adversarial examples. On the other hand, greater L also makes the certified condition more difficult to be satisfied, which result in lower CertAcc.

In Figure 6.6(b), we fix the certified bound L to 40, which means using the same power of WordDP to defend against adversarial examples generated by varying attack power L_{adv}. As shown in the figure, the performance increases with higher attack power. This is because the adversarial examples with more word changes (higher L_{adv}) are more difficult to generate but easier to defend (due to the nature of PWWS attack algorithm). **Comparison with Empirical Defense.** Besides certified robust method SAFER, we also compare Con-

dAcc of WordDP with baseline empirical defense methods, IBP [42] and DNE [108]. Table 6.1 compares the highest CondAcc achieved by WordDP with the conventional accuracy reported by the baselines (ADV corresponds to no defense). WordDP achieves a much higher accuracy on IMDB dataset compared to IBP, DNE and SAFER. For AGNews, the accuracy of WordDP outperforms SAFER, but is lower than the two empirical defenses. We stress, however, the empirical defense methods do not provide any rigorous certified robustness guarantees and the performance can be significantly dependent on datasets and specific attacks.

Efficiency Comparison. We also compare the efficiency of WordDP with SAFER by computing the average time cost for certifying one input and producing the Monte Carlo sampling-based output. It takes WordDP 6.25s and 3.21s on IMDB and AGNews, respectively. As a comparison, it costs SAFER 230.35s and 96.68s. Thus, WordDP achieves more than $30\times$ efficiency improvement.

Chapter 7

Wasserstein Adversarial Examples on Univariant Time Series Data and its Certified Robustness

7.1 Overview

As deep learning models are increasingly used for time series data, and potential adversarial attacks are present in many applications where the use of time series data is crucial, several works adapted adversarial attacks from images to time series data [40, 48, 72] using L_p norms. In these work, our goal is to develop more powerful adversarial examples on time-series data and study the potential certified robustness mechanism to improve the robustness of time-series deep learning models.

First, we aim to propose a more powerful and natural adversarial examples in time-series data. The notion of indistinguishability of adversarial examples, in the context of computer vision, was originally taken to be L_∞ bounded perturbations, which refers to noise with limited magnitude injected to each pixel [35]. In time series analysis, there are more effective metrics for measuring similarity between two temporal sequences, es-

pecially when the sequences vary in length and speed [2, 10]. Wasserstein distance [93], which is the numerical cost of an optimal transportation problem, allows us to analyze the distance between two time sequences. This distance can be intuitively understood for time sequence as the cost of moving around feature mass from one time step to another (transportation plan) in order to make two sequences the same.

We study the adversarial attack on time series in the Wasserstein space, to generate adversarial examples that have small Wasserstein perturbation so it is more indistinguishable and natural to human, e.g., physician who examines ECG data. Projected gradient descent attack [67] is a widely-used attack method that applies small steps of maximizing the loss objective iteratively and clipping the values of intermediate results after each step (projection to the L_p norm ball) to ensure that they are in a constrained neighbourhood of the original inputs. Similarly, we propose a Wasserstein PGD method to search for adversarial examples in the Wasserstein space for univariant time series. Wasserstein distance cannot be calculated directly without solving an optimization subproblem and has no closed-form solution in most cases, which limits its applications. At present, there are only two cases that the Wasserstein distance can be directly calculated, one is the case of the dimension of inputs being 1, and the other is the inputs following Gaussian distribution. For the univariant time series, we can take advantage of its $1D$ characteristic and use the closed-form Wasserstein distance to apply the projection of intermediate results of each step onto the Wasserstein ball with gradient descent method.

Followed by that, we also study the certified robustness approach to Wasserstein adversarial examples which provides a theoretical guarantee that adversarial examples generated within certain distance bounds can be correctly classified. As Wasserstein adversarial examples are bounded by Wasserstein distance, the existing and well-known certified defense within Euclidean distance is not applicable [18]. Therefore, we adapt Wasserstein Smoothing [58], a certified robustness approach to Wasserstein adversarial examples for image data which transfers Wasserstein distance on the image into L_1 distance on the transport

plan, to univariant time series adversarial examples. From the results, although the defense can achieve some accuracy gain, it still has limitations in many cases and leaves space for developing a stronger certified robustness method to Wasserstein adversarial examples on univariant time series data.

7.2 Proposed Method

In this section, we present our proposed Wasserstein adversarial example against time series models. We rely on the most common method of creating adversarial examples, the variation of projected gradient descent (PGD). The original PGD algorithm uses L_∞ clipping to perform the projection while we will use Wasserstein projection in our method. First, we will explain how to perform the Wasserstein projection. Followed by that, we will explain the Wasserstein PGD algorithm and the two-step projection.

7.2.1 Wasserstein Projection

Let (x, y) be a data point and its label, and $\mathcal{B}(x, \epsilon)$ be a ball around x with radius ϵ. The (general) projection of a point w on to $\mathcal{B}(x, \epsilon)$ can be formulated as:

$$\underset{\mathcal{B}(x,\epsilon)}{proj}(w) = \underset{z \in \mathcal{B}(x,\epsilon)}{\operatorname{argmin}} ||w - z||_2^2. \tag{7.1}$$

A Wasserstein ball around sample x with radius ϵ can be defined as:

$$\mathcal{B}_w(x, \epsilon) = \{x + \delta : d_{\mathcal{W}}(x, x + \delta) \leq \epsilon\}, \tag{7.2}$$

where $d_{\mathcal{W}}(u, v)$ refers to the Wasserstein distance between two sample distributions in the space $\mathcal{X} = \mathbb{R}^2$, which can be calculated as:

$$d_{\mathcal{W}}(u, v) = [\inf_{\gamma \in \Pi(u,v)} \int_{\mathcal{X}^2} ||x - y||^p d\gamma(x, y)]^{1/p}. \tag{7.3}$$

Here γ defines the joint probability distribution, called coupling, which has marginal distribution exactly as u and v.

Algorithm 1: 1D_Wasserstein_Projection

Input: Original input: x; Initial adversarial input x^0_{adv}; Wasserstein projection bound: $\mathcal{C}$; Step size: α

Output: Projected Adversarial example: x_{adv}

$i \leftarrow 0$;
L^0_w =$\mathrm{d}_{\mathcal{W}}$(x, x^0_{adv}) (Formula 7.5);
while $L_w \geq \mathcal{C}$ **do**
    $x^{i+1}_{adv} = x^i_{adv} - \alpha \cdot \frac{\partial L_w(x, x^i_{adv})}{\partial x^i_{adv}}$;
    L^i_w = $\mathrm{d}_{\mathcal{W}}$(x, x^i_{adv});
    i++;
    if $i \geq \mathbb{I}$ **then**
        Break;
        Return False;

When the input distribution satisfies the dimension being one, there is a closed-form solution for the above $d_{\mathcal{W}}$:

$$\begin{aligned} d_{\mathcal{W}}(u, v) &= ||F_u^{-1} - F_v^{-1}||^p \\ &= (\int_0^1 ||F_u(\alpha)^{-1} - F_v(\alpha)^{-1}||^p d\alpha)^{1/p}. \end{aligned} \tag{7.4}$$

When p equals 1 and the inputs are in the discrete case, formula 7.4 can be further simplified as:

$$\begin{aligned} d_{\mathcal{W}}(u, v) &= \int_{\mathbb{R}} |F_u(\alpha) - F_v(\alpha)| d\alpha \\ &= \sum_{i=1}^{n} |\sum_{j=1}^{i} u_i - \sum_{j=1}^{i} v_i|. \end{aligned} \tag{7.5}$$

In this case, the transport plan is $t = F_v^{-1} \odot F_u$.

Specifically, projecting w onto the Wasserstein ball around x with radius ϵ is defined as:

$$\underset{\mathcal{B}_{\mathcal{W}}(x,\epsilon)}{proj}(w) = \underset{z \in \mathcal{B}_{\mathcal{W}}(x,\epsilon)}{\operatorname{argmin}} d_{\mathcal{W}}(x, z). \tag{7.6}$$

As $1D$ Wasserstein distance has a closed-form solution and the above formula is also differentiable, we can apply the gradient descend method to the projection, the algorithm is explained in Algorithm 1. Note that this method may not find the exact projection onto the Wasserstein ball, but it can converge to an example within the Wasserstein ball.

7.2.2 Wasserstein PGD Attack

Algorithm 2: Wasserstein PGD Attack on Univarient Time series

Input: Original input: $\mathcal{X} = \{x, y\}$; Target model: $\mathscr{F}_\theta$; Attack Iteration: $\mathbb{T}$; Step size: ϵ; Wasserstein projection bound: ζ; L_∞ Norm clipping value: δ

Output: Adversarial example: x_{adv}

S $t \leftarrow 0$;

Init $x^0_{adv} = x + \mathcal{N}(0, 1)$;

while $t \leq \mathbb{T}$ **do**

`/* gradient descend step */`

$\eta = \epsilon \cdot sign(\nabla_x L_{cross\_entropy}(x, y, \mathscr{F}_\theta))$;

$x^{t+1}_{adv} = x^t_{adv} + \eta$;

`/* norm ball clipping with center` x `and radius` ζ `*/`

$x^{t+1}_{adv} = \min(\max(x^{t+1}_{adv}, x^{t+1}_{adv} - \zeta), x^{t+1}_{adv} + \zeta)$;

`/* Wasserstein ball projection with center` x `and radius` δ `*/`

x^{t+1}_{adv} = 1D_Wasserstein_Projection(x, x^{t+1}_{adv}, δ) (Algorithm 1);

t++;

PGD attack utilizes the concept of back-propagation. It takes the gradient of the loss function over inputs, to generate small perturbations at each time step and iteratively adds the perturbation to the clean input to generate adversarial examples followed by projection (clipping into the L_∞ ball), such that the new adversarial example has a greater tendency towards being misclassified. For Wasserstein adversarial attack, each iteration of the algorithm includes a gradient descent step to update the perturbed example followed by Wassersten projection, which is formulated as:

$$x^{t+1}_{adv} = \underset{\mathcal{B}_\mathcal{W}(x,\epsilon)}{proj} \ (x^t_{adv} + \alpha^T \nabla \mathcal{L}(x^t_{adv}, y)). \tag{7.7}$$

Theoretically, Wasserstein projection can be performed from any starting point. However, direct projection onto the Wasserstein ball using gradient descent can be time consuming and converge to a sub-optimal solution. We propose a two-step projection method (shown in Algorithm 2) that first projects the adversarial example to a norm-ball and then uses the projected example as the starting point for Wasserstein projection. As shown in Figure 7.1, the blue curve refers to the direct Wasserstein projection using gradient descent (this is the gradient descent used to perform projection shown in line 3-5 in Algorithm 1). This is performed after each step of gradient descent that updates the adversarial example (this is the gradient descent used to generate the perturbation shown in line 4-5 in Algorithm 2). The red curve and green curve refer to the two-step projection that first projects (clips) the example to the norm ball (line 6) and then projects it to the Wasserstein ball (line 7). In this way, the search in the Wasserstein space is guided and constrained, which is more effective and efficient.

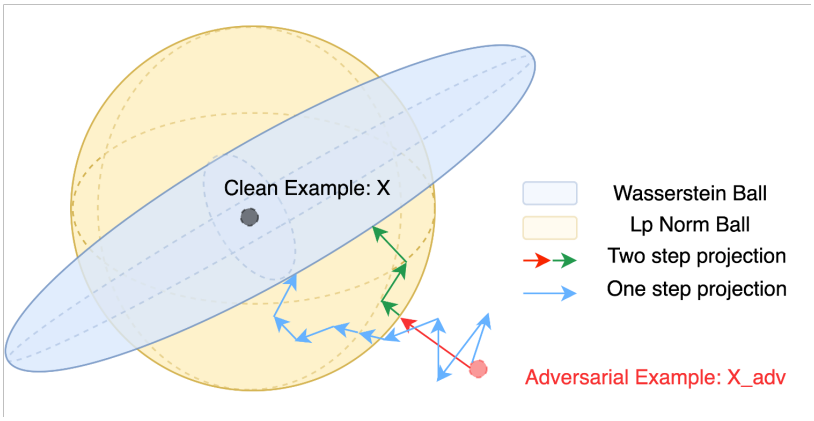


Figure 7.1: Illustration of the difference between direct projection and two-step projection

7.3 Experiments

In this section, we will first describe our experimental settings and then evaluate our proposed Wasserstein PGD in terms of the Attack Success Rate (ASR, the fraction of examples that label has been fliped), and comparison with the PGD attack in the Euclidean space. We

also demonstrate the effectiveness of the 2-step projection. Finally, we will demonstrate the results of certified robustness to the proposed Wasserstein PGD attack.

	Data Description			
Dataset	**TrainSize**	**TestSize**	**Classes**	**SeqLen**
ECG200	100	100	2	96
ECG5000	500	4500	5	160
ECGFiveDays	23	861	2	136
	Model Performance			
Dataset	**MLP**	**FCN**	**CNN**	**ResNet**
ECG200	0.916	0.9	0.83	0.89
ECG5000	0.931	0.939	0.928	0.934
ECGFiveDays	0.979	0.987	0.885	0.993

Table 7.1: Summary of datasets

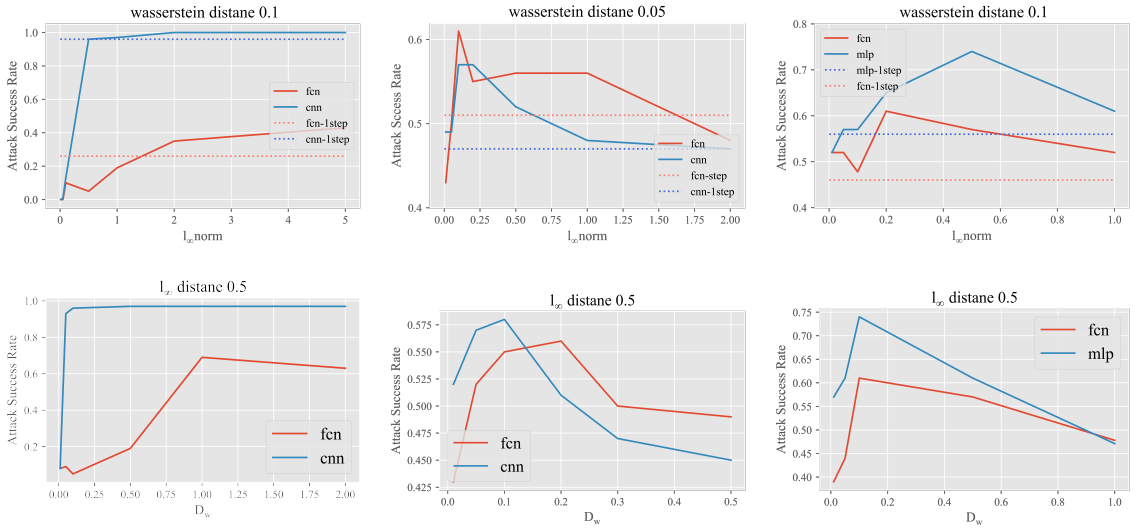


Figure 7.2: Attack Success Rate under different l_∞ and Wasserstein Bound: The columns represent the three dataset respectively.The first row illustrate under the same Wasserstein distance bound, how the attack success rate change with the increase of the l_∞ bound; The Second row illustrate under the same l_∞ bound, how the attack success rate change with the increase of the Wasserstein distance bound.

7.3.1 Experimental Setup

Our experiments are evaluated on Five benchmark time series classification datasets from the publicly available UCR archive [16]. The datasets are selected under the "ECG" cat-

egory for ECG based diagnosis tasks, where an adversarial attack is a potential security concern. In this section, we only show the results on three of the five ECG dataset, which are: 1) ECG200 which includes two classes (normal heartbeat and Myocardial Infarction), and contains 35 half-hour records sampled with the rate of 125 Hz. 2) ECG5000 which is the Beth Israel Deaconess Medical Center (BIDMC) congestive heart failure database, consisting of records of 15 subjects, with severe congestive heart failure. Five labels refes to different levels of heart failure. Records of each individual were recorded in 20 hours, containing two ECG signals, sampled with the rate of 250 Hz. 3) ECGFiveDays which is from a 67-year-old male, including two classes which are two ECG dates. We relegate the full version of the experiments of the other two datasets to the Appendix.

We adopt and evaluate the target deep learning models from [41] including: Multi-Layer Perceptron (MLP) [32], Fully Connected Networks (FCN)[64], Convolutional Neural Networks (CNN) [49] and Residual Networks (ResNet) [36]. The detailed information of the ECG datasets and the performance of the target models on each dataset is listed in Table 7.1.

7.3.2 Attack Success Rate

Our proposed 2-step projection Wasserstein PGD involves a first projection to a L_∞ norm-ball and a second projection to a Wasserstein ball. Figure 7.2 illustrates the impact of these two projections by comparing the ASR under different L_∞ and Wasserstein bounds. The first row shows under the same Wasserstein distance bound, how the ASR changes with the increase of the L_∞ bound, while the Second row shows under the same Ł$_\infty$ bound, how the ASR changes with the increase of the Wasserstein distance bound. Each column represents the results of each dataset and each line in each figure corresponds to a target model. We show two models for each dataset and relegate the full version of the experiments to the Appendix.

Under the same radius of Wasserstein ball, the general trend is that ASR first increases

with the increase of L_∞ bound. This is intuitive as the search space for optimal adversarial examples that satisfy the Wasserstein distance constraint is increasing. It is easier to find an adversarial example that successfully attack the target model and meanwhile satisfy the Wasserstein constraint. However, as the L_∞ bound keeps increasing, it does not help anymore and even hurts the performance because the search in the Wasserstein Space is not guided and constrained any more and can be too large for the projection to find a good solution. Therefore, the ASR stops increasing or starts to decrease. This decreasing trend is more noticeable in ECG200 and ECGFiveDays datasets, while for ECG5000, the ASR increases to and stays at 1 (for the CNN model). Note that our attack is untargeted attack that aims to flip the label to any class other than the original label rather than the targeted label. Therefore, multi-class classification task (ECG5000) is easier to attack than binary classification tasks (ECG200 and ECGFiveDays).

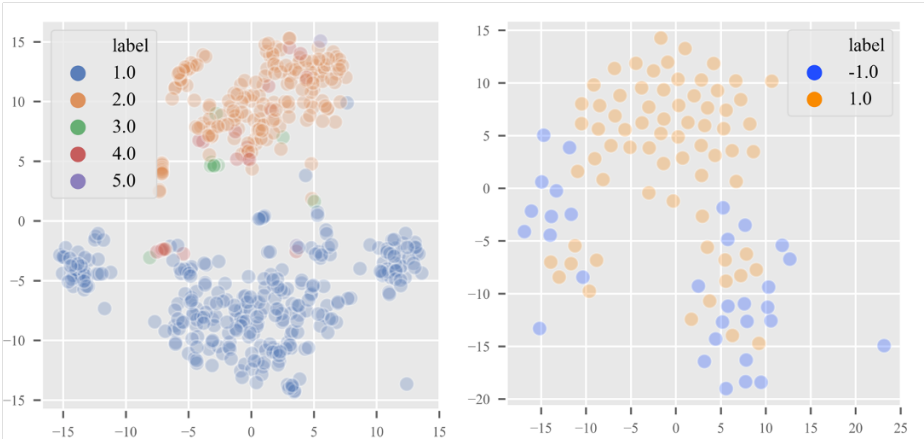


Figure 7.3: t-sne for ECG5000 (left) and ECG200 (right)

To further explain the difference between the datasets, we use t-distributed stochastic neighbor embedding (t-SNE), a nonlinear dimensionality reduction technique for high-dimensional visualization in a low-dimensional space [94], to visualize the datasets. Figure 7.3 shows the 2D t-sne for ECG200 and ECG5000 respectively. Each color represents a class label. We can note that data points of ECG200 is more separable and the class boundary is more clear, while the classes are overlapping for ECG5000 and the boundary is less clear, which makes it easier to attack. This explains why for ECG5000, the ASR increases

to 1 and does not decrease.

On the other hand, under the same L_∞ bound shown in the bottom row of Figure 7.2, larger Wasserstein bound also renders higher ASR at first due to larger search space. As it keeps increasing, the ASR decreases due to the search space being too large and the ineffectiveness of the search, especially when the radius of Wasserstein ball is greater than the Euclidean ball. Overall, the ASR of ECG5000, ECG200 and ECGFivedays can reach 100%, 62% and 74% respectively.

7.3.3 Effectiveness of 2-step Projection

From the perspective of ASR, we compare the 2-step projection with the direct 1-step Wasserstein projection shown as the dotted lines in the top row of Figure 7.2 under the same attack settings. We observe that 2-step projection can achieve a higher ASR in general and optimal attack success rate when choosing the proper L_∞ bound for the first projection.

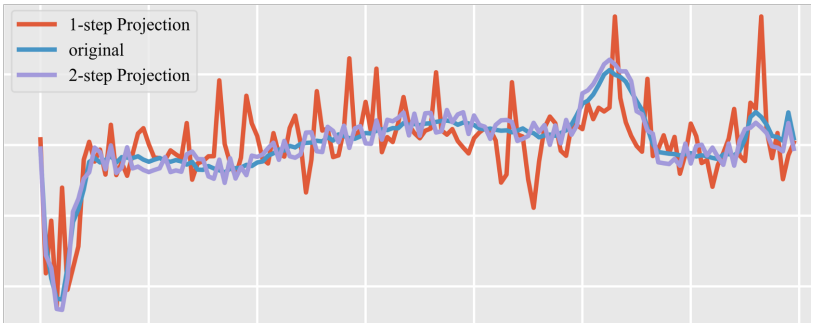


Figure 7.4: Comparison between direct Wasserstein projection (1-step projection) and 2-step projection.

From the perspective of human inspection, Figure 7.4 shows two successful adversarial examples generated by the 2-step projection (purple) and direct projection (red) respectively in comparison with the original example (blue). Although the Wasserstein perturbation distances of the two adversarial examples are both 0.99, the one that is first norm clipped is more imperceptible to human eyes, which has not only small Wasserstein distance but also bounded by L_∞ distance.

7.3.4 Comparison with L_∞ PGD

The intuition of developing the Wasserstein PGD attack is to search for more indistinguishable and natural adversarial examples in the Wasserstein space. Therefore, we compare the adversarial examples generated by Wasserstein PGD with those generated by original PGD in the Euclidean space (L_∞ PGD). On one hand, We draw the utility comparison from two aspects: 1) Under the same attack success rate, Wasserstein PGD is more natural; and 2) Under the same perturbation scale, Wasserstein PGD has a higher attack success rate. On the other hand, as Wasserstein projection involves gradient descent which will add more time cost in generating adversarial examples, we also compare the average time cost of two attack methods.

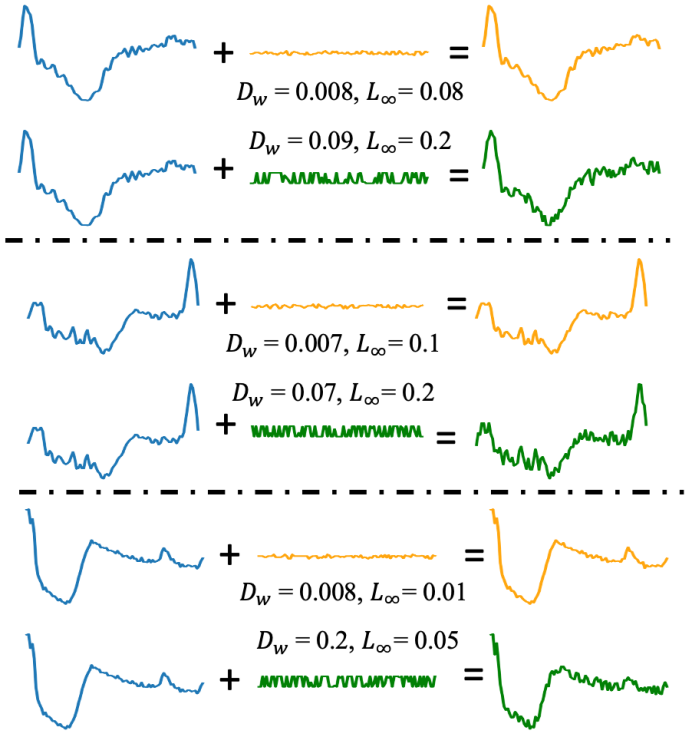


Figure 7.5: Comparison between Wasserstein PGD (yellow) and L_∞ PGD (green) under the same attack success rate.

Under the same attack success rate, Wasserstein PGD is more natural

Figure 7.5 illustrates several comparisons between Wasserstein PGD and L_∞ PGD under the same ASR. We selected three examples randomly. For each figure, the blue curve is

the original input. The yellow curves represent the perturbation and adversarial example generated by Wasserstein PGD, while the green curves represent the L_∞ PGD. Clearly, the perturbation generated from Wasserstein PGD is smaller and more indistinguishable than L_∞ PGD.

Under the same perturbation scale, Wasserstein PGD has a higher attack success rate.

Another aspect to show the effectiveness of Wasserstein PGD is to compare the ASR with the original PGD under the same perturbation scale. However, the two attacks are conducted in different spaces. It is unfair to compare the ASR of adversarial examples generated from the Wasserstein ball and the L_∞ ball with the same radius, as they represent completely different spaces. To overcome this challenge, we use the greatest L_∞ norm of the Wasserstein examples as the radius of L_∞ ball to generate L_∞ PGD adversarial examples. For example, the maximum L_∞ norm of the Wasserstein adversarial examples with 0.01 Wasserstein distance is 0.2. Then we will search for adversarial examples in the L_∞ ball with the radius equal to 0.2. In this way, we have a fair comparison between the Wasserstein PGD and the original PGD attack.

Figure 7.6 shows the comparison of ASR in the way we introduced above. The x-axis refers to different attack settings corresponding to different Wasserstein and L_∞ bounds (note that we have two steps of projections, first to an L_∞ norm ball and second to a Wasserstein ball, while original PGD only uses L_∞ projection). The purple and orange lines correspond to the Wasserstein PGD and original PGD respectively. We can note that in most cases, Wasserstein PGD has a higher attack success rate than the original attack.

From these two aspects, we can conclude that for univariant time series data, Wasserstein PGD not only can generate more natural adversarial examples but also can achieve a higher ASR under the same attack scale.

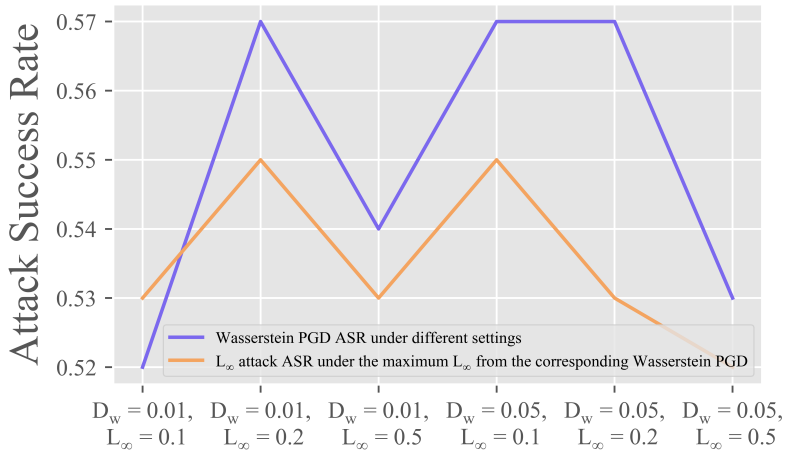
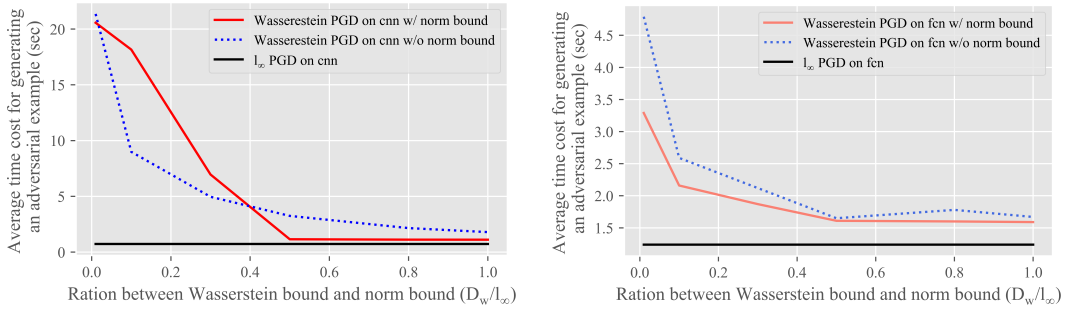


Figure 7.6: Comparison between Wasserstein PGD and l_∞ PGD under the same attack scale.

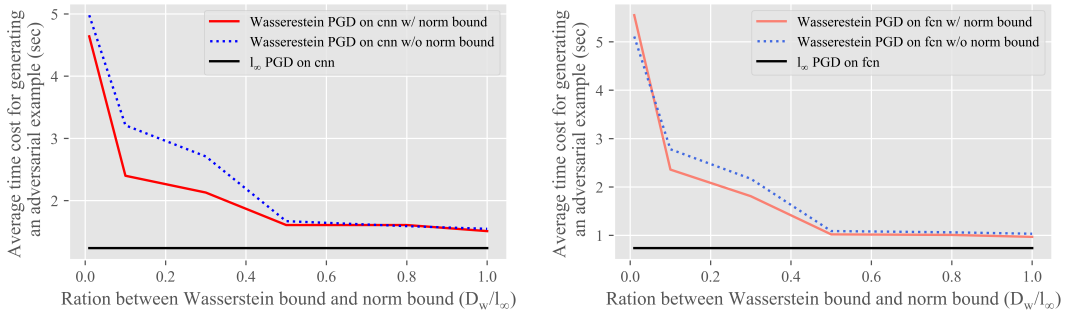
Compare the time cost between Wasserstein PGD and L_∞ PGD

Besides comparing the utility of Wasserstein PGD with L_∞ PGD, we also evaluate how Wasserstein PGD increases the time cost by comparing the average time cost of generating an adversarial example with L_∞ PGD attack, as Wasserstein PGD involves projection into the Wasserstein ball via gradient which will result in more time cost. As the time cost of the Wasserstein projection is largely relative to the size of L_∞ ball (the start point of gradient descent) and the size of the Wasserstein ball (the destination), we compare the average time cost according to the ratio between Wasserstein bound and L_∞ bound.

As shown in Figure 7.7, the red and blue curves are the Wasserstein PGD with and without the first stage of norm clipping, while the black line represents the baseline L_∞ PGD attack, whose value is irrelevant to the ratio between Wasserstein and L_∞ bound. We show the result on two datasets and two models: ECG200 and ECG5000, with CNN and FCN models. We can note that Wasserstein PGD will result in more time cost than L_∞ PGD, especially when the ratio between Wasserstein bound and L_∞ bound is large. However, when the ratio approaches 1, this increase in time cost is neglectable. By comparing the red and blue curves we can conclude that when the ratio between the Wasserstein bound and L_∞ bound approaches 1, the first stage of norm clipping can effectively guide the search for Wasserstein adversarial examples. However, when the ratio increases(approaches 0), even



(a) ECG200 CNN (b) ECG200 FCN



(c) ECG5000 CNN (d) ECG5000 FCN

Figure 7.7: The average time cost of generating an adversarial example with Wasserstien PGD (with and without norm bound clipping) and L_∞ PGD attack.

with the bound of norm clipping, the search also tends to be a random search.

7.3.5 Countermeasure against Wasserstein PGD

To better study the nature of Wasserstein adversarial examples, we also explored certified robustness approach as a potential defense mechanism for Wasserstein PGD. We consider certified robustness in contrast to other empirical defense methods as it is the most powerful and principled defense method to date.

We adopt Wasserstein smoothing [58] which is originally designed for image data to the univariant time series data. The basic idea of Wasserstein smoothing is to define a reduced transport plan and map the Wasserstein distance on the input space to the L_1 norm on the transport plan. The base classifier is transformed into a smoothed classifier by adding Laplacian noise $Laplace(0, \sigma)^r$ to the reduced transport plan, where r is corresponding to

the input dimension. Because of the mapping, smoothing in the transport domain can be performed using existing L_1 robustness certification provided by [55] and mapped back to the Wasserstein Space. The strict form of the certified condition can be stated as:

Theorem 7.3.1. *For any normalized probability distribution input $x \in R^{n*m}$ with correct class c, if:*

$$\bar{f}_c(x) \geq e^{2\sqrt{2}\zeta/\sigma}(1 - \bar{f}_c(x)) \tag{7.8}$$

then for any perturbed x that $d_W(x, x) \leq \zeta$, we have

$$\bar{f}_c(x) \geq \underset{i \neq c}{max} \bar{f}_i(x) \tag{7.9}$$

The detailed proof and the design of the reduced transport plan can be referred to [58]

We evaluate how Wasserstein Smoothing empirically works on the proposed Wasserstein PGD attack with two evaluation metrics [97]: **Certified accuracy (CertAcc)** which denotes the fraction of the clean testing set on which the predictions are correct and also satisfy the certification criteria. Formally, it is defined as: $\frac{\sum_{t=1}^{T} certifiedCheck(\mathbf{X}_t, L, \epsilon) \& corrClass(\mathbf{X}_t, L, \epsilon)}{T}$, where $certifiedCheck$ returns 1 if Theorem 7.3.1 is satisfied and $corrClass$ returns 1 if the classification output is correct. T is the size of the test dataset. **Conventional accuracy (ConvAcc)** is defined as the fraction of testing set that is correctly classified, $\frac{\sum_{t=1}^{T} corrClass(\mathbf{X}_t, L, \epsilon)}{T}$, which is a standard metric to evaluate any deep learning systems.

Figure 7.8 illustrates how the certified accuracy changes under different Wasserstein certified radius ζ in Equation 7.8. During the experiments, the Laplace parameter σ which controls the noise to the transport plan is set to 0.01 and the soft prediction result is averaged over 100 times of sampling. As Wasserstein radius increases, the certified accuracy decreases, which means less fraction of examples can satisfy the certification condition in Equation 7.8. When Wasserstein radius ζ is set over 0.1, the certified accuracy stops decreasing. We can note from this result that: first, under a certain Laplacian distribution, the certified radius is limited. Second, to a very small certified radius, for examples less than

0.01, the certified accuracy can achieve 100%.

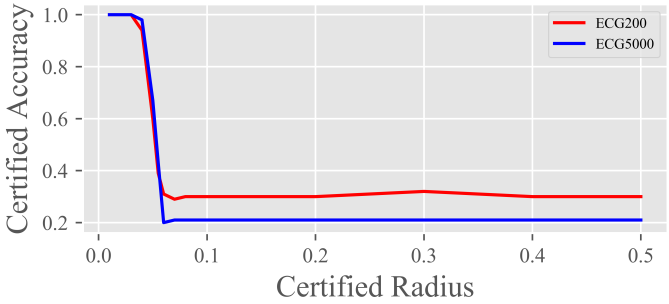


Figure 7.8: Comparison of Certified Accuracy under different Wasserstein radius with $\sigma = 0.01$

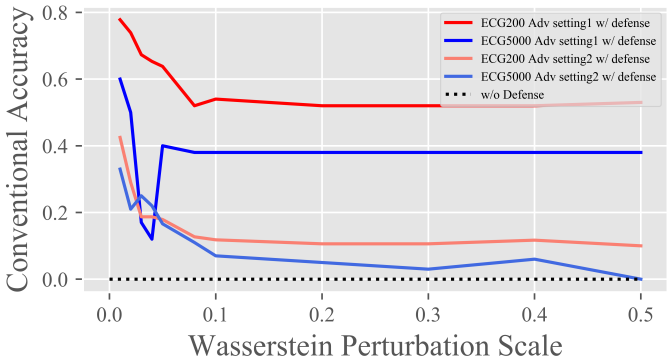


Figure 7.9: Comparison of Conventional Accuracy of successfully attacked adversarial examples.

Figure 7.9 demonstrates whether Wasserstein smoothing can empirically increase the conventional accuracy of adversarial examples generated by Wasserstein PGD. In the experiments, we test on successfully generated adversarial examples, so the baseline accuracy of the dataset is 0. Laplace parameter σ is set to 0.1 and the soft prediction result is averaged over 100 times of sampling. The x-axis refers to the Wasserstein attack scales. The two red lines represent the adversarial examples generated from ECG200 under two different settings, L_∞ norm set to 0.1 and 0.2, and the two blue lines represent the adversarial examples generated from ECG5000 where L_∞ norm set to 0.1 and 0.2. We can note from the figures the following. First, when the Wasserstein attack scales are small, Wasserstein smoothing can render some accuracy gain. The accuracy decreases with the increase of Wasserstein perturbation scales as expected. When the Wasserstein distance increase over 0.06, which is also the greatest certified radius, the accuracy gain is limited and does not change anymore. We also randomly select two adversarial examples with Wasserstein perturbations

around 0.06. As shown in Figure 7.10, the adversarial ECG is fairly indistinguishable to human eyes. Yet Wasserstein Smoothing can not provide reasonable defense at this level. Second, with ECG200 and ECG5000 under setting2 (L_∞ norm set to 0.2), the overall accuracy gain is very small for all perturbation scales.

We can conclude from the results that the existing Wasserstein smoothing has limited success in both certified ratio and conventional accuracy gain. This suggests that there is still space for developing stronger certified robustness method to Wasserstein PGD tailored to time series data instead of using the general transport plan based smoothing designed for image data.

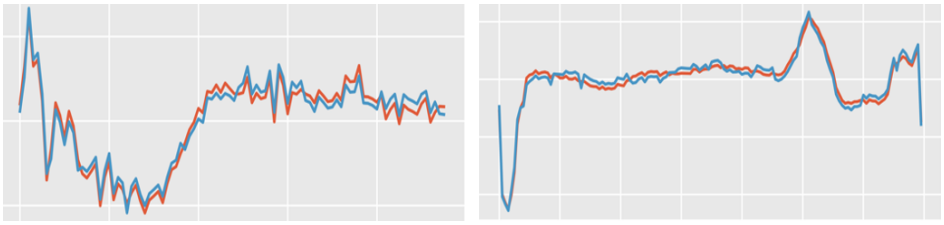


Figure 7.10: The comparison between adversarial examples and clean examples at $d_{\mathcal{W}}$ scale around 0.06.

Chapter 8

Conclusion and future work

My dissertation thesis combines several works to improve the robustness of deep learning systems to adversarial examples in sequential data, especially EHR data and NLP. We propose and evaluate several empirical defense algorithms as well as certified robustness algorithms. In section 4, we propose RADAR, the first effort to defend adversarial examples on temporal EHR data. We benefit from the autoencoder's reconstruction ability to distinguish adversarial examples and clean examples on EHR data. In order to more effectively model the multivariate time series data, we build an autoencoder by integrating attention mechanism with bi-directional LSTM cell to capture both past and future of the current time frame and their interdependence. Experimental results showed that RADAR can filter out more than 90% of adversarial examples and improve the target model accuracy by more than 90% and F1 score by 60%.

In section 5, we propose MATCH to enhance the robustness of deep learning systems on EHR data. MATCH system aims to detect whether an input is adversarial, under the circumstance that one modality has been compromised, by measuring the consistency between the compromised modality (clinical notes) and another uncompromised modality (temporal EHR). we conduct a case study on predicting the 30-days readmission risk using an EHR dataset. Experimental results show that MATCH outperforms existing defense

techniques in the text domain due to the special characteristics of clinical notes.

In section 6, we propose a novel approach WordDP to certified robustness against word substitution attacks in NLP via differential privacy (DP). We establish the connection between DP and certified robustness for the first time in text classification domain, and leverage conceptual exponential mechanism to achieve WordDP and formally prove an L-word bounded certified condition for robustness against word substitution attacks. Extensive experiments validate that WordDP outperforms existing defense methods and achieves over 30× efficiency improvement in the inference stage than the state-of-the-art certified robustness mechanism

n section 7, we develop more powerful adversarial examples on times series data and study the potential certified robustness mechanism to improve the robustness of time-series deep learning models. We first develop an attack algorithm by adopting Wasserstein distance to better capture the perturbation magnitude instead of the common L_2 distance on image, and propose Wasserstein PGD to generate more natural adversarial examples to human eyes and achieve higher attack success rate. Followed by that, we evaluate Wasserstein smoothing which is designed for image data as a potential certified robustness method to Wasserstein adversarial examples that can provide certified bound in the Wasserstein space. Results show that the existing Wasserstein smoothing has limited success in both certified ratio and conventional accuracy gain. This suggests that there is still space for developing stronger certified robustness method to Wasserstein PGD tailored to time series data instead of using the general transport plan based smoothing designed for image data.

Future works. There are a few research directions that we would like to explore in the future. In section 6, we attempted to apply WordDP to the most state-of-the-art NLP model BERT. As Bert is a pre-trained model and the last a few layers are always fine-tuned according to different tasks, it is unrealistic to add a noise layer and incorporate WordDP into the whole training process. Although WordDP can provide considerable empirical accuracy improvement, other certified robustness mechanism such as Safer provide near-zero

certified accuracy. Therefore, it would be meaningful to study the certified mechanisms for pre-trained models, as in both industry and academia, increasingly large model size are used to achieve state-of-the-art performance, and the pre-train plus fine-tune schema has become the mainstream. Besides involving certified robustness mechanism in the training, We would also adopt adversarial training to improve overall conventional accuracy. To better evaluate the effective of WordDP, we will also adopt more difficult dataset such as google benchmark.

Instead of studying certified robustness in the word space, we also want to study the certified robustness in the word embedding space. We would study both attack and certified defending algorithms on the generative models generated embedding or contextual embeddings.

From broader picture of view, it would be interesting to extend the certified robustness definition to applications other than classifications, such as regression, real-time prediction and text generation, as well as other model architectures.

Bibliography

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016.

[2] Helmut Alt and Michael Godau. Computing the fréchet distance between two polygonal curves. *International Journal of Computational Geometry & Applications*, 5(01n02):75–91, 1995.

[3] Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, 2018. Association for Computational Linguistics.

[4] Sungtae An, Cao Xiao, Walter F. Stewart, and Jimeng Sun. Longitudinal adversarial attack on electronic health records data. In *The World Wide Web Conference*, 2019.

[5] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*, 2018.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[7] Shumeet Baluja and Ian Fischer. Adversarial transformation networks: Learning to generate adversarial examples. *arXiv preprint arXiv:1703.09387*, 2017.

[8] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 464–473. IEEE, 2014.

[9] Amos Beimel, Shiva Prasad Kasiviswanathan, and Kobbi Nissim. Bounds on the sample complexity for private learning and private data release. In *Theory of Cryptography Conference*, pages 437–454. Springer, 2010.

[10] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA, USA:, 1994.

[11] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017.

[12] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14, 2017.

[13] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017.

[14] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey. *arXiv preprint arXiv:1810.00069*, 2018.

[15] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26, 2017.

[16] Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, and Gustavo Batista. The ucr time series classification archive, July 2015. `www.cs.ucr.edu/~eamonn/time_series_data/`.

[17] Yong Cheng, Lu Jiang, and Wolfgang Macherey. Robust neural machine translation with doubly adversarial inputs. *ACL*, 2019.

[18] Jeremy M Cohen, Elan Rosenfeld, and J Zico Kolter. Certified adversarial robustness via randomized smoothing. *arXiv preprint arXiv:1902.02918*, 2019.

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[21] Cynthia Dwork. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, pages 1–19. Springer, 2008.

[22] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[23] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 2014.

[24] Cynthia Dwork, Guy N Rothblum, and Salil Vadhan. Boosting and differential privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 51–60. IEEE, 2010.

[25] Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. Hotflip: White-box adversarial examples for text classification. *ACL*, 2018.

[26] Reuben Feinman, Ryan R Curtin, Saurabh Shintre, and Andrew B Gardner. Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*, 2017.

[27] Weijiang Feng, Naiyang Guan, Yuan Li, Xiang Zhang, and Zhigang Luo. Audio visual speech recognition with multimodal recurrent neural networks. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 681–688. IEEE, 2017.

[28] Samuel G Finlayson, John D Bowers, Joichi Ito, Jonathan L Zittrain, Andrew L Beam, and Isaac S Kohane. Adversarial attacks on medical machine learning. *Science*, 2019.

[29] Samuel G Finlayson, Hyung Won Chung, Isaac S Kohane, and Andrew L Beam. Adversarial attacks against medical deep learning systems. *arXiv preprint arXiv:1804.05296*, 2018.

[30] Kazuto Fukuchi, Quang Khai Tran, and Jun Sakuma. Differentially private empirical risk minimization with input perturbation. In *International Conference on Discovery Science*, pages 82–90. Springer, 2017.

[31] Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 50–56. IEEE, 2018.

[32] Matt W Gardner and SR Dorling. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric environment*, 32(14-15):2627–2636, 1998.

[33] Manuel Gil. *On Rényi divergence measures for continuous alphabet sources*. PhD thesis, Citeseer, 2011.

[34] Zhitao Gong, Wenlu Wang, and Wei-Shinn Ku. Adversarial and clean data are not twins. *arXiv preprint arXiv:1704.04960*, 2017.

[35] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[36] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[37] Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*, 2019.

[38] Po-Sen Huang, Robert Stanforth, Johannes Welbl, Chris Dyer, Dani Yogatama, Sven Gowal, Krishnamurthy Dvijotham, and Pushmeet Kohli. Achieving verified robustness to symbol substitutions via interval bound propagation. *EMNLP-IJCNLP*, 2019.

[39] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[40] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Adversarial attacks on deep neural networks for time series classification. *2019 International Joint Conference on Neural Networks (IJCNN)*, Jul 2019.

[41] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Deep learning for time series classification: a review. *Data mining and knowledge discovery*, 33(4):917–963, 2019.

[42] Robin Jia, Aditi Raghunathan, Kerem Göksel, and Percy Liang. Certified robustness to adversarial word substitutions. *EMNLP-IJCNLP*, 2019.

[43] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? natural language attack on text classification and entailment. *arXiv preprint arXiv:1907.11932*, 2019.

[44] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3:160035, 2016.

[45] Erik Jones, Robin Jia, Aditi Raghunathan, and Percy Liang. Robust encodings: A framework for combating adversarial typos. *ACL*, 2020.

[46] Yilin Kang, Yong Liu, Lizhong Ding, Xinwang Liu, Xinyi Tong, and Weiping Wang. Differentially private erm based on data perturbation. *arXiv preprint arXiv:2002.08578*, 2020.

[47] Yilin Kang, Yong Liu, Ben Niu, Xinyi Tong, Likun Zhang, and Weiping Wang. Input perturbation: A new paradigm between central and local differential privacy. *arXiv preprint arXiv:2002.08570*, 2020.

[48] Fazle Karim, Somshubra Majumdar, and Houshang Darabi. Adversarial attacks on time series. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3309–3320, Oct 2021.

[49] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[50] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[51] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

[52] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*, 2015.

[53] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[54] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019.

[55] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. *2019 IEEE Symposium on Security and Privacy (SP)*, May 2019.

[56] Guang-He Lee, Yang Yuan, Shiyu Chang, and Tommi Jaakkola. Tight certificates of adversarial robustness for randomly smoothed classifiers. In *Advances in Neural Information Processing Systems*, pages 4910–4921, 2019.

[57] Jaewoo Lee and Daniel Kifer. Concentrated differentially private gradient descent with adaptive per-iteration privacy budget. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1656–1665, 2018.

[58] Alexander Levine and Soheil Feizi. Wasserstein smoothing: Certified robustness against wasserstein adversarial attacks. In *International Conference on Artificial Intelligence and Statistics*, pages 3938–3947. PMLR, 2020.

[59] Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Certified adversarial robustness with additive noise. In *Advances in Neural Information Processing Systems*, pages 9464–9474, 2019.

[60] Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. Textbugger: Generating adversarial text against real-world applications. *arXiv preprint arXiv:1812.05271*, 2018.

[61] Yi Li, Huahong Zhang, Camilo Bermudez, Yifan Chen, Bennett A Landman, and Yevgeniy Vorobeychik. Anatomical context protects deep learning from adversarial perturbations in medical imaging. *Neurocomputing*, 379:370–378, 2020.

[62] Yingzhen Li and Yarin Gal. Dropout inference in bayesian neural networks with alpha-divergences. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2052–2061. JMLR. org, 2017.

[63] Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. Deep text classification can be fooled. *IJCAI*, 2018.

[64] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[65] Xingjun Ma, Yuhao Niu, Lin Gu, Yisen Wang, Yitian Zhao, James Bailey, and Feng Lu. Understanding adversarial attacks on deep learning based medical image analysis systems. *arXiv preprint arXiv:1907.10456*, 2019.

[66] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150, 2011.

[67] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[68] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 94–103. IEEE, 2007.

[69] Dongyu Meng and Hao Chen. Magnet: a two-pronged defense against adversarial examples. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*.

[70] Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations. *arXiv preprint arXiv:1702.04267*, 2017.

[71] Takeru Miyato, Andrew M Dai, and Ian Goodfellow. Adversarial training methods for semi-supervised text classification. *stat*, 1050:7, 2016.

[72] Gautam Raj Mode and Khaza Anuarul Hoque. Adversarial examples in deep learning for multivariate time series regression. *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, Oct 2020.

[73] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016.

[74] Andrew Ng et al. Sparse autoencoder. *CS294A Lecture notes*, 2011.

[75] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017.

[76] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pages 372–387. IEEE, 2016.

[77] Nicolas Papernot, Patrick McDaniel, Ananthram Swami, and Richard Harang. Crafting adversarial input sequences for recurrent neural networks. In *MILCOM 2016-2016 IEEE Military Communications Conference*, pages 49–54. IEEE, 2016.

[78] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597. IEEE, 2016.

[79] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[80] Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. Generating natural language adversarial examples through probability weighted word saliency. Association for Computational Linguistics.

[81] Andras Rozsa, Ethan M Rudd, and Terrance E Boult. Adversarial diversity and hard positive generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 25–32, 2016.

[82] Hadi Salman, Greg Yang, Jerry Li, Pengchuan Zhang, Huan Zhang, Ilya Razenshteyn, and Sebastien Bubeck. Provably robust deep learning via adversarially trained smoothed classifiers. *arXiv preprint arXiv:1906.04584*, 2019.

[83] Motoki Sato, Jun Suzuki, Hiroyuki Shindo, and Yuji Matsumoto. Interpretable adversarial perturbation in input embedding space for text. *IJCAI*, 2018.

[84] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security*, pages 1528–1540, 2016.

[85] Benjamin Shickel, Patrick James Tighe, Azra Bihorac, and Parisa Rashidi. Deep ehr: a survey of recent advances in deep learning techniques for electronic health record (ehr) analysis. *IEEE journal of biomedical and health informatics*, 22(5):1589–1604, 2017.

[86] Reza Shokri and Vitaly Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1310–1321, 2015.

[87] Lewis Smith and Yarin Gal. Understanding measures of uncertainty for adversarial example detection. *arXiv preprint arXiv:1803.08533*, 2018.

[88] Shuang Song, Kamalika Chaudhuri, and Anand D Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 245–248. IEEE, 2013.

[89] Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. *arXiv preprint arXiv:1710.10766*, 2017.

[90] Mengying Sun, Fengyi Tang, Jinfeng Yi, Fei Wang, and Jiayu Zhou. Identify susceptible locations in medical records via adversarial attacks on deep predictive models. pages 793–801, 07 2018.

[91] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[92] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017.

[93] SS Vallender. Calculation of the wasserstein distance between probability distributions on the line. *Theory of Probability & Its Applications*, 18(4):784–786, 1974.

[94] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[95] Aleksandra Vatian, Natalia Gusarova, Natalia V. Dobrenko, Sergey Dudorov, Niyaz Nigmatullin, Anatoly A. Shalyto, and Artem Lobantsev. Impact of adversarial examples on the efficiency of interpretation and use of information from high-tech medical images. *FRUCT*, 2019.

[96] Di Wang, Minwei Ye, and Jinhui Xu. Differentially private empirical risk minimization revisited: Faster and more general. In *Advances in Neural Information Processing Systems*, pages 2722–2731, 2017.

[97] Wenjie Wang, Pengfei Tang, Jian Lou, and Li Xiong. Certified robustness to word substitution attack with differential privacy. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1102–1112, 2021.

[98] Xiaosen Wang, Hao Jin, and Kun He. Natural language adversarial attacks and defenses in word level. *arXiv preprint arXiv:1909.06723*, 2019.

[99] Nilmini Wickramasinghe. Deepr: a convolutional net for medical records. 2017.

[100] Lin Xiao and Tong Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

[101] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155*, 2017.

[102] Mao Ye, Chengyue Gong, and Qiang Liu. Safer: A structure-free approach for certified robustness to adversarial word substitutions. *ACL*, 2020.

[103] Lei Yu, Ling Liu, Calton Pu, Mehmet Emre Gursoy, and Stacey Truex. Differentially private model publishing for deep learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 332–349. IEEE, 2019.

[104] Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6066–6080, 2020.

[105] Tahmina Zebin and Thierry J Chaussalet. Design and implementation of a deep recurrent model for prediction of readmission in urgent care using electronic health records. In *IEEE CIBCB*, 2019.

[106] Huangzhao Zhang, Hao Zhou, Ning Miao, and Lei Li. Generating fluent adversarial examples for natural languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, 2019. Association for Computational Linguistics.

[107] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657, 2015.

[108] Yi Zhou, Xiaoqing Zheng, Cho-Jui Hsieh, Kai-wei Chang, and Xuanjing Huang. Defense against adversarial attacks in nlp via dirichlet neighborhood ensemble. *arXiv preprint arXiv:2006.11627*, 2020.

[109] Yichao Zhou, Jyun-Yu Jiang, Kai-Wei Chang, and Wei Wang. Learning to discriminate perturbations for blocking adversarial attacks in text classification. *arXiv preprint arXiv:1909.03084*, 2019.