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

Wenjie Wang[1]✉, Pengfei Tang[1], Li Xiong[1], and Xiaoqian Jiang[2]

[1] Emory University, Atlanta, GA, USA
{wang.wenjie,pengfei.tang,lxiong}@emory.edu
[2] UTHealth, Houston, TX, USA xiaoqian.jiang@uth.tmc.edu

Abstract. Leveraging the information-rich and large volume of Electronic Health Records (EHR), deep learning systems have shown great promise in assisting medical diagnosis and regulatory decisions. Although deep learning models have advantages over the traditional machine learning approaches in the medical domain, the discovery of adversarial examples has exposed great threats to the state-of-art deep learning medical systems. While most of the existing studies are focused on the impact of adversarial perturbation on medical images, few works have studied adversarial examples and potential defenses on temporal EHR data. 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. We evaluate RADAR on a mortality classifier using the MIMIC-III dataset. Experiments show 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%. Besides, we also propose an enhanced attack by introducing the distribution divergence into the loss function such that the adversarial examples are more realistic and difficult to detect.

Keywords: Adversarial Example Detection · Recurrent Autoencoder · Temporal Electronic Health Records (EHR).

1 Introduction

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[26]. 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[33]. 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[30]. Taking advantage of a sufficient amount of training dataset, in some cases, complex neural networks can even exceed capabilities of experienced physicians in head-to-head comparisons[6].

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[10]. 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[8] due to high financial interests such as insurance claims.

Most research on adversarial examples in medical domain has been focused on medical images, such as X-ray and MRI image[20,32] which can be easily adapted from traditional image domain. The attack algorithms in the image domain aim to minimize the perturbation scale while mislead model predictions. This optimization problem can be either directly solved such as in C&W attack[4] or approximated with gradient method such as Fast Gradient Sign Method[10]. A few recent works have studied adversarial examples on temporal EHR data. Sun et al.[30] proposed a Recurrent Neural Network (RNN)-based time-preferential minimum attack strategy to identify susceptible locations on EHR data. An et al.[1] proposed LAVA, a saliency score based adversarial example generation approach that aims to minimize the number of perturbations. However, it only works for binary-coded features and is not applicable for general temporal EHR with continuous or categorical features.

Despite these two attempts on the attack algorithms for temporal EHR data, there is no study on potential defense techniques. The existing defense mechanisms in image domain can be categorized into adversarial training[27], image denoising[7] and detection mechanisms[21,22]. One of the most promising and state-of-the-art detection methods is MagNet[21], which is based on autoencoder and rejects examples with large autoencoder reconstruction errors. As MagNet can work with any pre-trained classifier, only requires clean data for training, and does not depend on specific image features, it has the potential to be adapted for temporal EHR data. However, there are several critical challenges due to the characteristics of temporal EHR data:

- *Multivariate temporal dependency.* The intuition of autoencoder based defense is to learn the representation from clean data. However learning the representation and capturing the pattern of time-series EHR data is more challenging than images due to the temporal dependency between time points in addition to the correlations between attributes. Besides, the significance of each timestamp on the prediction outcomes differ as more recent features may have a stronger influence.
- *Sparsity and high-dimensionality.* Sequential EHR data is extremely sparse, discrete and high-dimensional compared to image data. Therefore, the tradi-

tional distance metrics may not be effective for measuring the autoencoder reconstruction error which cannot capture the real similarity or validity of temporal EHR data.

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. Similar to MagNet, 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.

Different from existing methods, 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[2] 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 which are used in MagNet, 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.

Besides RADAR, we also propose an enhanced attack by introducing distribution divergence into the loss function, making the adversarial examples more realistic and difficult to detect. To our knowledge, RADAR is the first effort to propose defense techniques on temporal EHR data. We evaluate RADAR on a mortality classifier using the MIMIC-III[14] 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.

2 Preliminaries and Related Work

Neural Networks for Sequential Data. Deep neural networks (DNN) have been increasingly applied to solve difficult real-world tasks. For time-sequence data, Recurrent Neural Network (RNN) is designed for capturing the temporal information among features. A variant of RNN, Long Short-Term Memory

(LSTM) network[12] is proposed to capture not only the short-term dependency but also the long term dependency among temporal features. In order to model both forward (past to current) and backward (current to past) temporal correlation, Schuster et al.[25] proposed a bi-directional structure by feeding the reversed input into RNN model as well.

Autoencoder is a type of neural network architecture that learns the data representation in an unsupervised manner through dimension reduction[11]. Recurrent autoencoder refers to a type of autoencoder whose layers are RNN cells[29], which has been widely applied to sequence to sequence (seq2seq) tasks such as machine translation[5,31]. To solve the long-term dependency problem of recurrent autoencoder, Bahdanau et al.[2] proposed an attention mechanism that calculates the weights of states among all the time steps as the attention scores and computes an element-wise weighted sum of all the states as the context vector. Recurrent autoencoder without attention mechanisms has been applied for EHR data imputation and synthesization[35]. In this paper, we adopt a recurrent autoencoder with attention mechanism for the temporal EHR data and use it for adversarial example detection for the first time.

The applications of RNN on sequential EHR data range from mortality prediction, readmission prediction, to trajectory prediction[24,34,36]. Most works use different datasets with different pre-processing methods, and cannot be directly applied to our data. In this work, since our focus is not on the classification model, we adopt a single layer LSTM model as our target classifier to demonstrate the effectiveness of the proposed adversarial example detection method.

Adversarial Examples. Generating adversarial examples can be formulated as a constrained optimization problem. Given a clean input x, its label y and a classifier F, if $L_p(x, x_{adv}) < C$, such that $F(x_{adv}) \neq y$, x_{adv} is an adversarial example, where L_p represents the L_p-norm of the perturbation and C represents the perturbation constraint. This optimization problem can be either directly solved such as in C&W attack[4] or approximated with gradient method such as Fast Gradient Sign Method (FGSM)[10] and iterative FGSM[15].

Very recently, it has been pointed out that medical machine learning systems may be uniquely susceptible to adversarial examples[8]. Several works studied adversarial examples in medical image models[9,17,20,32]. A few works explored the adversarial examples on temporal sequential EHR data. Sun et al.[30] proposed an RNN-based time-preferential minimum attack strategy. Their attack algorithm is similar to the C&W attack in image domain. An et.al[1] 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. We propose an enhanced attack in this paper and compare it with the attack algorithm in Sun et al.[30]

Defenses against Adversarial Examples. The existing defense methods against adversarial examples (mainly focused on the image domain) can be characterized into three categories:

- Image preprocessing and denoising such as image compression[7, 13] which are image specific and autoencoder based denoiser (HGD)[19]. The drawback of HGD is that it requires a large number of adversarial samples to train the denoiser.
- Detection based defense mechanism. The traditional detection method is usually a binary classifier which is trained on both adversarial samples and clean samples[22]. However, these detectors failed to generalize across various attack schemes. More recently, Mend et al.[21] proposed an autoencoder based detector called MagNet, which rejects samples (as adversarial examples) with large reconstruction errors. One major advantage of MagNet is that it only requires clean examples for training the autoencoder, which significantly increases its generalization ability.
- Adversarial training. Adversarial training[27] utilizes adversarial examples and integrate them in model training. It can be also used in combination with gradient masking[3, 23] which makes gradient-based attacks infeasible or difficult. The drawback of adversarial training is that it lacks the generalization ability to unseen adversarial examples and may compromise the model performance on clean examples. In addition, it requires a larger number of adversarial examples in the training stage.

Until now, there is no defense algorithms proposed for adversarial examples on sequential EHR data. The existing defense strategies for image data are either specific to the image domain, or require large volume of clean and adversarial training data, which is not suitable. MagNet has a strong generalization ability and does not depend on image characteristics. Besides, it does not require adversarial examples in training phase and is independent of the target classifier. In this work, we adapt this autoencoder based detection method and propose the first defense mechanism against adversarial examples on temporal EHR data.

3 Methodology

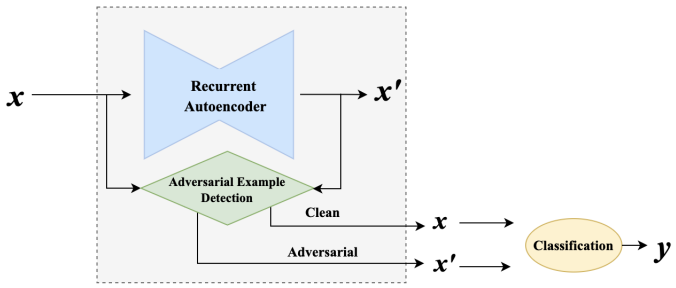


Fig. 1: RADAR pipeline

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 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.

3.1 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.

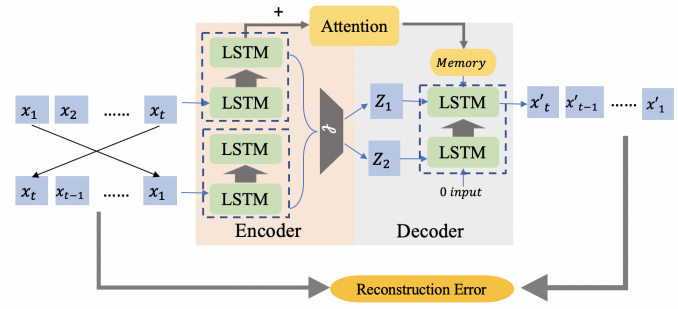


Fig. 2: BRNN-AE Architecture.

Our model is a bidirectional-RNN autoencoder which is shown in Figure 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 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[2] 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:

$$L(x, x') = \|x, x'\|_2 + L_{reg}(\theta) \tag{1}$$

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

3.2 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:

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

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[18], mutual information and differential entropy[28]. In this work, we use entropy of predictive distribution to reflect uncertainty, which is defined as:

$$Entropy(l_{x'}) = -\sum_{i=1}^{n} s_i log(s_i), \quad \text{where} \quad s_i = \frac{e^{l^i_{x'}}}{\sum_{j=1}^{n} e^{l^j_{x'}}} \tag{3}$$

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.

3.3 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.[30]. 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:

$$\arg\min_{x_{adv}} L_y + \alpha L_x, \quad with \tag{4}$$

$$L_y = \max\{l(x_{adv})_{y_{true}} - l(x_{adv})_{y_{false}}, -k\} \quad and \quad L_x = ||x_{adv} - x||_p \tag{5}$$

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[16] 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:

$$\arg\min_{x_{adv}} L_y + \alpha L_x - \beta N(x|x_{adv}, I) \tag{6}$$

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 Experimental Evaluation

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.

Dataset and Model Architecture. MIMIC-III (The Multiparameter Intelligent Monitoring in Intensive Care) dataset[14] 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.[30] 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.

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 1.

Table 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

4.1 Attack Performance

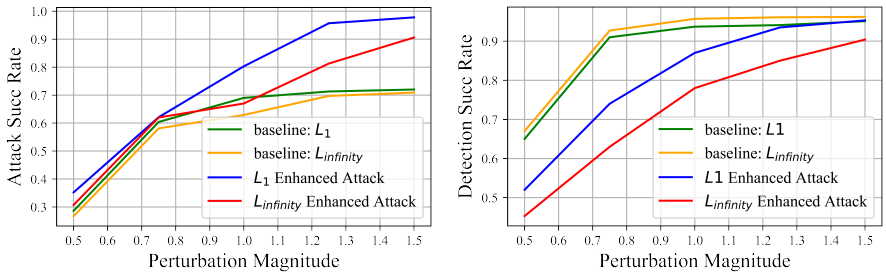
We use different distance metric to measure the similarity between adversarial examples and clean examples, including L_p-norm and KL divergence. L_p-norm aims to measure EHR location-wise similarity and KL divergence measures the distribution similarity over the whole set of adversarial examples and 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 2: Attack performance comparison

Loss Func / **Metric**	**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**

Table 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 6 are set to 0. For the L_1-norm attack (Sun et al.[30]) 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 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.

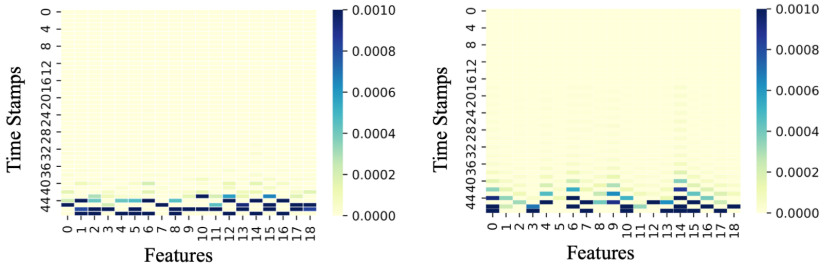
The above results show the comparison of different attack methods for successfully flipped examples. To give a more comprehensive comparison, we also use varying perturbation magnitude as stopping criteria and compare the attack success rate and detection rate (by our detection approach) of different attack methods, which is shown in Figure 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.



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

Fig. 3: Comparison between baseline attack and enhanced 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. 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

Fig. 4: Mean perturbation distribution

4.2 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 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.

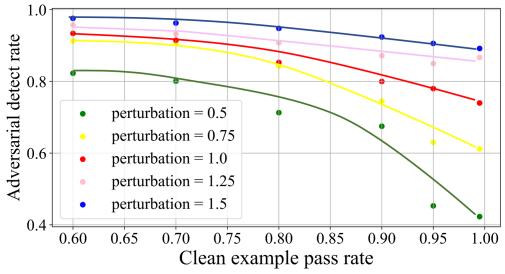
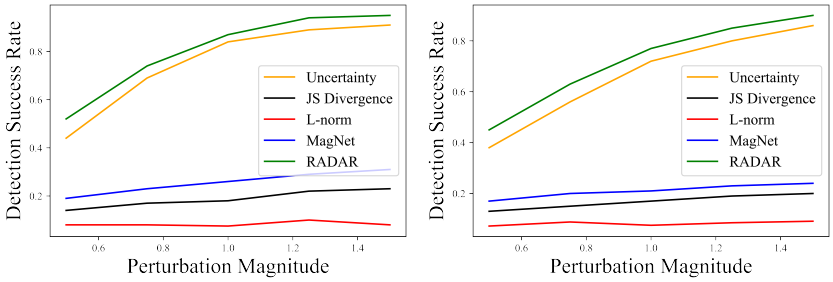


Fig. 5: The trade-off between adversarial detection rate and clean pass rate

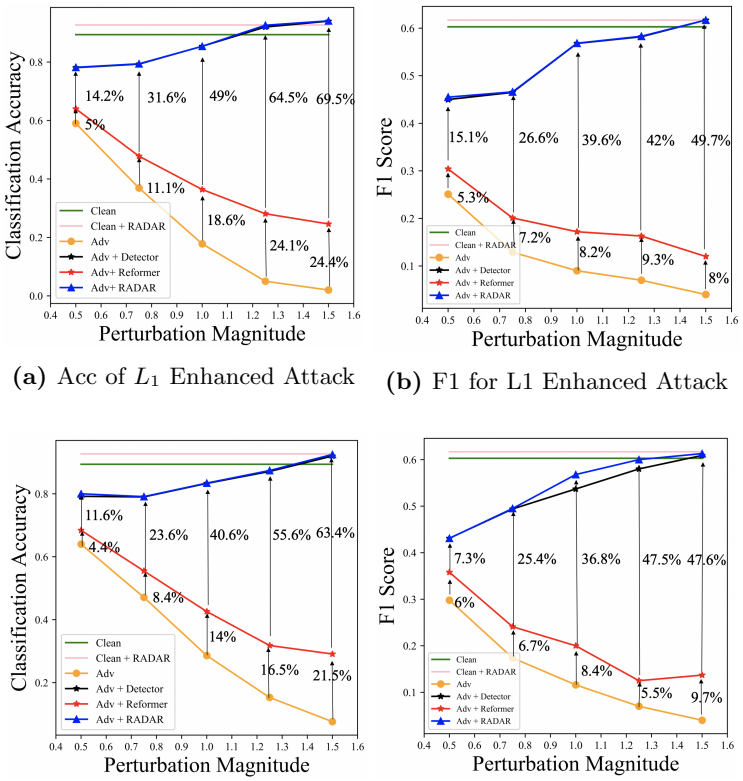


(a) RADAR performance under L_1 enhanced attack

(b) RADAR performance under L_∞ enhanced attack

Fig. 6: Contribution of each criterion and comparison of RADAR with MagNet

Detection Success Rate. Figure 6 shows how much contribution each detection criterion makes to filter adversarial examples. It also compares RADAR



(a) Acc of L_1 Enhanced Attack **(b)** F1 for L1 Enhanced Attack

(c) Acc of L_∞ Enhanced Attack **(d)** F1 for L_∞ Enhanced Attack

Fig. 7: Performance improvement

(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 improvement 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 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 7d. 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.

5 Conclusion

This paper is the first attempt to study potential defense methods for adversarial examples on temporal EHR data. We proposed a recurrent autoencoder based detection method called RADAR to detect adversarial examples according to autoencoder reconstruction error, prediction divergence, and prediction uncertainty. According to the evaluation on a mortality classifier, RADAR can effectively detect more than 90% of adversarial examples and improve the target model accuracy and F1 score by almost 90% and 60% respectively. Besides, we also introduced an enhanced adversarial attack by incorporating the distribution divergence into the loss function of the attack algorithm.

In the future, we plan to evaluate the performance of RADAR on other clinical deep learning systems such as readmission prediction models. In addition, the architecture of RADAR also has great potential to be improved by incorporating other deep learning models that are more powerful to model structural EHR data such as Graph Convolutional Networks (GCN).

Acknowledgement

This work is partially supported by the National Science Foundation (NSF) BigData award IIS-1838200, the Georgia Clinical Translational Science Alliance

under National Institutes of Health (NIH) CTSA Award UL1TR002378, and Air Force Office of Scientific Research (AFOSR) DDDAS award FA9550-12-1-0240. XJ is CPRIT Scholar in Cancer Research (RR180012), and he was supported in part by Christopher Sarofim Family Professorship, UT Stars award, UTHealth startup, the National Institute of Health (NIH) under award number R01AG066749, R01GM114612 and U01TR002062.

References

1. An, S., Xiao, C., Stewart, W.F., Sun, J.: Longitudinal adversarial attack on electronic health records data. In: The World Wide Web Conference (2019)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
3. Buckman, J., Roy, A., Raffel, C., Goodfellow, I.: Thermometer encoding: One hot way to resist adversarial examples (2018)
4. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP). pp. 39–57. IEEE (2017)
5. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
6. Choi, E., Bahadori, M.T., Schuetz, A., Stewart, W.F., Sun, J.: Doctor ai: Predicting clinical events via recurrent neural networks. In: Machine Learning for Healthcare Conference. pp. 301–318 (2016)
7. Das, N., Shanbhogue, M., Chen, S.T., Hohman, F., Chen, L., Kounavis, M., Chau, D.H.: Keeping the bad guys out: Protecting and vaccinating deep learning with jpeg compression (2017)
8. Finlayson, S.G., Bowers, J.D., Ito, J., Zittrain, J.L., Beam, A.L., Kohane, I.S.: Adversarial attacks on medical machine learning. Science **363**, 1287–1289 (2019)
9. Finlayson, S.G., Chung, H.W., Kohane, I.S., Beam, A.L.: Adversarial attacks against medical deep learning systems. arXiv preprint arXiv:1804.05296 (2018)
10. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)
11. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. science **313**(5786), 504–507 (2006)
12. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
13. Jia, X., Wei, X., Cao, X., Foroosh, H.: Comdefend: An efficient image compression model to defend adversarial examples. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6084–6092 (2019)
14. Johnson, A.E., Pollard, T.J., Shen, L., Li-wei, H.L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G.: Mimic-iii, a freely accessible critical care database. Scientific data **3**, 160035 (2016)
15. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial machine learning at scale. arXiv preprint arXiv:1611.01236 (2016)
16. Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O.: Autoencoding beyond pixels using a learned similarity metric. arXiv preprint arXiv:1512.09300 (2015)
17. Li, Y., Zhang, H., Bermudez, C., Chen, Y., Landman, B.A., Vorobeychik, Y.: Anatomical context protects deep learning from adversarial perturbations in medical imaging. Neurocomputing **379**, 370–378 (2020)

18. Li, Y., Gal, Y.: Dropout inference in bayesian neural networks with alpha-divergences. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 2052–2061. JMLR. org (2017)
19. Liao, F., Liang, M., Dong, Y., Pang, T., Hu, X., Zhu, J.: Defense against adversarial attacks using high-level representation guided denoiser. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018)
20. Ma, X., Niu, Y., Gu, L., Wang, Y., Zhao, Y., Bailey, J., Lu, F.: Understanding adversarial attacks on deep learning based medical image analysis systems. arXiv preprint arXiv:1907.10456 (2019)
21. Meng, D., Chen, H.: Magnet: a two-pronged defense against adversarial examples. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 135–147. ACM (2017)
22. Metzen, J.H., Genewein, T., Fischer, V., Bischoff, B.: On detecting adversarial perturbations. arXiv preprint arXiv:1702.04267 (2017)
23. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against machine learning. In: Proceedings of the 2017 ACM on Asia conference on computer and communications security. pp. 506–519 (2017)
24. Pham, T., Tran, T., Phung, D., Venkatesh, S.: Predicting healthcare trajectories from medical records: A deep learning approach. Journal of biomedical informatics (2017)
25. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE transactions on Signal Processing **45**(11), 2673–2681 (1997)
26. Shickel, B., Tighe, P.J., Bihorac, A., Rashidi, P.: 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)
27. Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., Webb, R.: Learning from simulated and unsupervised images through adversarial training. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2017)
28. Smith, L., Gal, Y.: Understanding measures of uncertainty for adversarial example detection. arXiv preprint arXiv:1803.08533 (2018)
29. Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using lstms. In: International conference on machine learning. pp. 843–852 (2015)
30. Sun, M., Tang, F., Yi, J., Wang, F., Zhou, J.: Identify susceptible locations in medical records via adversarial attacks on deep predictive models. pp. 793–801 (07 2018). https://doi.org/10.1145/3219819.3219909
31. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in neural information processing systems (2014)
32. Vatian, A., Gusarova, N., Dobrenko, N.V., Dudorov, S., Nigmatullin, N., Shalyto, A.A., Lobantsev, A.: Impact of adversarial examples on the efficiency of interpretation and use of information from high-tech medical images. FRUCT (2019)
33. Wickramasinghe, N.: Deepr: a convolutional net for medical records. (2017)
34. Zebin, T., Chaussalet, T.J.: Design and implementation of a deep recurrent model for prediction of readmission in urgent care using electronic health records. In: IEEE CIBCB (2019)
35. Zhang, J., Yin, P.: Multivariate time series missing data imputation using recurrent denoising autoencoder. In: 2019 IEEE BIBM. pp. 760–764. IEEE (2019)
36. Zheng, H., Shi, D.: Using a lstm-rnn based deep learning framework for icu mortality prediction. In: International Conference on Web Information Systems and Applications. pp. 60–67. Springer (2018)