Utilizing Multimodal Feature Consistency to Detect Adversarial Examples on Clinical Summaries

Wenjie Wang
Emory University
Atlanta, GA, USA
wang.wenjie@emory.edu

Youngja Park
IBM Research
Yorktown Heights, NY, USA
young_park@us.ibm.com

Taesung Lee
IBM Research
Yorktown Heights, NY, USA
taesung.lee@ibm.com

Ian Molloy
IBM Research
Yorktown Heights, NY, USA
molloyim@us.ibm.com

Pengfei Tang
Emory University
Atlanta, GA, USA
pengfei.tang@emory.edu

Li Xiong
Emory University
Atlanta, GA, USA
lxiong@emory.edu

Abstract

Recent studies have shown that adversarial examples can be generated by applying small perturbations to the inputs such that the well-trained deep learning models will misclassify. With the increasing number of safety and security-sensitive applications of deep learning models, the robustness of deep learning models has become a crucial topic. The robustness of deep learning models for healthcare applications is especially critical because the unique characteristics and the high financial interests of the medical domain make it more sensitive to adversarial attacks. Among the modalities of medical data, the clinical summaries have higher risks to be attacked because they are generated by third-party companies. As few works studied adversarial threats on clinical summaries, in this work we first apply adversarial attack to clinical summaries of electronic health records (EHR) to show the text-based deep learning systems are vulnerable to adversarial examples. Secondly, benefiting from the multi-modality of the EHR dataset, we propose a novel defense method, *MATCH* (Multimodal feATure Consistency cHeck), which leverages the consistency between multiple modalities in the data to defend against adversarial examples on a single modality. Our experiments demonstrate the effectiveness of *MATCH* on a hospital readmission prediction task comparing with baseline methods.

1 Introduction

Deep learning has been shown to be effective in a variety of real-world applications such as computer vision, natural language processing, and speech recognition (Krizhevsky et al., 2012; He et al., 2016; Kim, 2014). It also has shown great potentials in clinical informatics such as medical diagnosis and regulatory decisions (Shickel et al., 2017), including learning representations of patient records, supporting disease phenotyping, and conducting predictions (Wickramasinghe, 2017; Miotto et al., 2016). However, recent studies show that these models are vulnerable to adversarial examples (Bruna et al., 2013). In image classification, researchers have demonstrated that imperceptible changes in input can mislead the classifier (Goodfellow et al., 2014). In the text domain, synonym substitution or character/word level modification on a few words can also cause the model to misclassify (Liang et al., 2017). These perturbations are mostly imperceptible to human but can easily fool a high-performance deep learning model.

Adversarial examples have received much attention in image and text-domain, yet very few work has been done on Electronic Health Records (EHR). Most existing works on adversarial examples in medical domains have been focused on medical images (Vatian et al., 2019; Ma et al., 2020). A few works have studied adversarial examples in numerical EHR data (Sun et al., 2018; An et al., 2019; Wang et al., 2020). Despite these attempts, there is no work on evaluating the adversarial robustness of clinical natural language processing (NLP) systems, as well as the potential defense techniques.

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) (Wang et al., 2019) is a

Proceedings of the 3rd Clinical Natural Language Processing Workshop, pages 259–268
November 19, 2020. ©2020 Association for Computational Linguistics

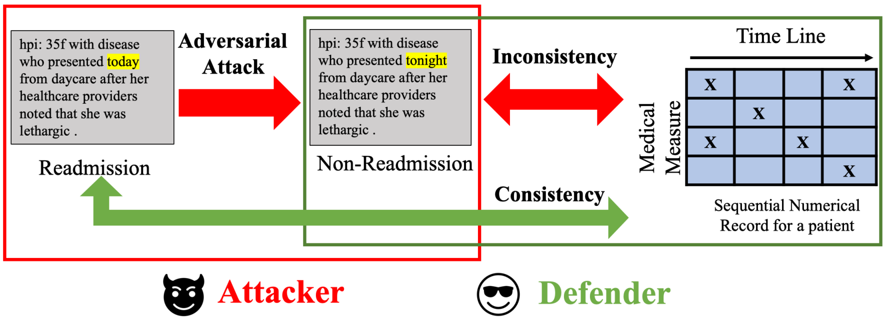

hpi: 35f with disease who presented today from daycare after her healthcare providers noted that she was lethargic .
Readmission
Adversarial Attack
hpi: 35f with disease who presented tonight from daycare after her healthcare providers noted that she was lethargic .
Non-Readmission
Inconsistency
Consistency
Time Line
Medical Measure
X
X
X
X
X
X
Sequential Numerical Record for a patient
Attacker
Defender

Figure 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.

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 (Miyato et al., 2016) 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.

We note that 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. 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 1 depicts the high-level flow of our system.

The main contributions of this paper include:

- 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.
- 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.
- 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.

2 Related Work

There have been many adversarial works on single modality adversarial tasks. Qiu *et al.* (2019) provided a comprehensive summary of the latest progress on adversarial attack and defense technology, categorized by applications including computer vision, natural language processing, cyberspace security, and physical world. Esmaeilpour *et al.* (2019) reviewed the existing adversarial attacks in audio classification. Since our case study focuses on attack and defense of text modality, we mainly review the text-based attacks and defenses in this section.

2.1 Attack Methods for Text Data

Kuleshov *et al.* (2018) proposed a Greedy Search Algorithm (GSA), which iteratively changes one word in a sentence and substitute the word with one of the synonymous that improves the objective function the most. Alzantot *et al.* (2018) introduced a Genetic Algorithm (GA) which is a population-based synonym replacement algorithm including processing, sampling and crossover. Gong *et al.* (2018) proposed to search for adversarial examples in the embedding space by applying gradient-based methods on text embedding (*Text-FGM*) and then reconstructed the adversarial texts by the nearest neighbor search. Gao *et al.* (2018) presented the *DeepWordBug* algorithm to generate small perturbations in the character-level. This algorithm does not require the gradient. Ren *et al.* (2019) proposed a new synonym substitution method, Probability Weighted Word Saliency (PWWS), which considered the word saliency as well as the classification probability. Jin *et al.* (2019) proposed TextFooler, an adversarial approach by identifying the important words and then prioritize to replace them with the most semantically similar and grammatically correct words. This is the first attempt to attack the emerging BERT model on text classification. We compare these algorithms from the following aspects:

Document level vs. Word level. *Text-FGM* and GA are document level attacks, which apply an attack on the whole text. *DeepWordBug*, GSA, PSWW, and TextFooler are word level attacks that perturb individual words. *DeepWordBug*, PSWW and TextFooler use heuristics to measure the importance of each word and select words to perturb.

Continuous vs. Discrete. *Text-FGM* is a continuous attack, because the gradient-based perturbation is applied on the embedding of the words. All other attacks are discrete attacks, which are applied directly on words.

Semantic vs. Syntactic. GSA, PSWW, Text-FGM and TextFooler can be categorized as a semantic attack since their strategies are to replace words or text with synonyms, while DeepWordBug is a syntactic attack because it is based on character-level modification. GA can generate both semantically and syntactically similar adversarial examples.

Back-box vs. White-box. GSA, GA and *Text-FGM* are white-box attacks, because attackers need to access the model structure and model parameters to calculate the gradient. *DeepWordBug*, TextFooler and PSWW are black-box attacks.

In this paper, we evaluate our detection method against *Text-FGM* and *DeepWordBug*, which represent all the categories mentioned above.

Text-FGM. In *Text-FGM*, any gradient based attacks, such as DeepFool (Moosavi-Dezfooli et al., 2016), Fast Gradient Method (FGM) (Goodfellow et al., 2014) (both FGSM and FGVM) can be applied. Applying FGVM on text is defined as follows. Given a classifier f and a word sequence $x = \{x_1, x_2, ...x_n\}$,

$$emb(x)' = emb(x) + \epsilon(\frac{\nabla L}{||\nabla L||_2}) \qquad (1)$$

where L is the loss function and emb denotes the embedding vector. Then, the adversarial example is chosen as $x_{adv} = NNS(emb(x)')$, where NNS represents the nearest neighbor search algorithm which returns the closest word sequence given a perturbed embedding vector.

In the following work, in order to minimize the number of words that need to be perturbed, we iteratively perform perturbation on one word at a time based on the importance score of the words, instead of applying perturbation on the entire sequence. In this way, we can maximize the overall semantic similarity between clean and adversarial sentences.

DeepWordBug. *DeepWordBug* first computes the word importance to the target sequence classifier. At each step, it selects the most important word and constructs an adversarial word applying a character level swap, substitution or deletion. It iterates until the label is flipped or the number of words changed is larger than a threshold.

2.2 Defense Methods for Text Data

Few works have been done on defending against adversarial examples in the text domain. Existing defense algorithms can be divided into detection and adversarial training.

Detection. Most detection methods use spelling check. Gao *et al.* (2018) used Python's Autocorrect 0.3.0 to detect character-level adversarial examples. Li *et al.* (2018) 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 *et al.* (2019) proposed a framework learning to discriminate perturbations (DISP), which learns to discriminate the perturbations and restore the original embeddings.

Adversarial Training. Adversarial training has been widely used in the image domain and also been adapted to text domain. Overfitting is 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. Miyato *et al.* (2016) applied the adversarial training to text domain and achieved the state-of-the-art-performance. Wang *et al.* (2019) proposed Synonyms Encoding Method (SEM), which tried to find a mapping between 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 on classifier. The classifier is forced to be smooth in this way. However, SEM can only work for synonym substitution attacks.

2.3 Readmission Prediction

Efforts on building deep learning models for readmission prediction have attracted a growing interest. MIMIC-III (The Multiparameter Intelligent Monitoring in Intensive Care) (Johnson et al., 2016), a publicly available clinical dataset comprising EHR information related to patients admitted to critical care units, has become a common choice for such studies. We demonstrate our framework using a case study on the MIMIC data and adopt the state-of-the-art classification models which are briefly reviewed here.

For numerical records, (Xue et al., 2019) studied the temporal trends of physiological measurements and medications, and used them to improve the performance of ICU readmission risk prediction models. They converted the time series of each variable into trend graphs. Then, they applied frequent subgraph mining to extract important temporal trends. They trained a logistical regression model on grouped temporal trends. (Zebin and Chaussalet, 2019) proposed a heterogeneous bidirectional Long Short Term Memory plus Convolutional neural network (BiLSTM+CNN) model. The combination of them can automate the feature extraction process, by considering both time-series correlation and feature correlation. They outperformed all the benchmark classifiers on most performance measures. At the same time, anothers also proposed a LSTM-CNN based model and achieved comparable performance (Lin et al., 2019). In this work, we adopt the architecture in (Zebin and Chaussalet, 2019) to conduct readmission prediction on sequential numerical records.

For text data, *Clinical BERT* is recently introduced (Huang et al., 2019; Alsentzer et al., 2019) to model clinical notes by applying the BERT model (**?**). They outperformed baselines which use both the discharge summaries and the first few days of notes in ICU. In this work, we adopt *Clinical BERT* to predict readmission on text data.

3 Method

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

3.1 Multi-modality Model Consistency Check

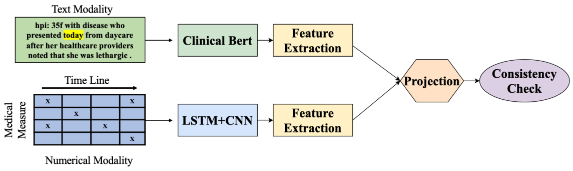


Figure 2: Detection Pipeline

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 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:

$$\min_{\theta_1,\theta_2} MSE(p_{\theta_1}(F_1(m_1)) - p_{\theta_2}(F_2(m_2))) \quad (2)$$

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:

$$||p_{\theta_1}(F_1(m_1)) - p_{\theta_2}(F_2(m_2))||_2 > \delta \quad (3)$$

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 (Devlin et al., 2019) 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 (Zebin and Chaussalet, 2019) . However, as our data preprocessing steps and selected features are different, we modify the architecture to optimize the performance. Our architecture (Figure 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 (Zebin and Chaussalet, 2019). The output of the final layer is used as the extracted features.

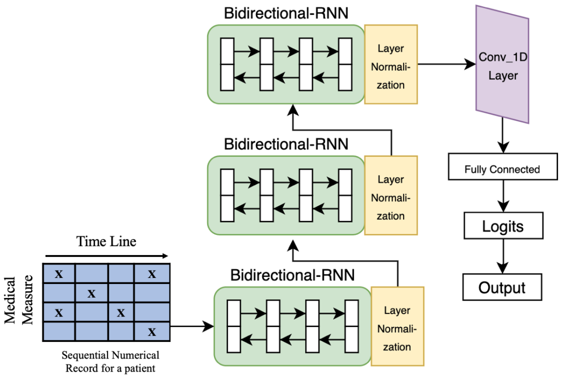


Figure 3: Stacked Bidirectional LSTM+CNN architecture

4 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.

4.1 Data Preprocessing

Clinical Summary. For the clinical summary, which is the target modality the attacker, we directly use the processed data from (Huang et al., 2019). The data contains 34,560 patients with 2,963 positive readmission labels and 48,150 negative labels. In MIMIC-III (Johnson et al., 2016), 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 × time step (120) × features (90).

4.2 Predictive Model Performance

For the clinical summary data, we use the pretrained *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.

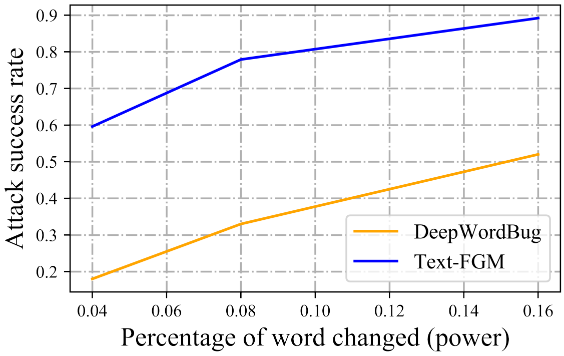
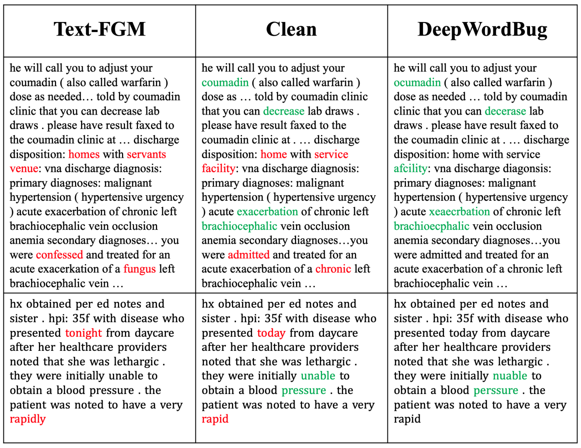
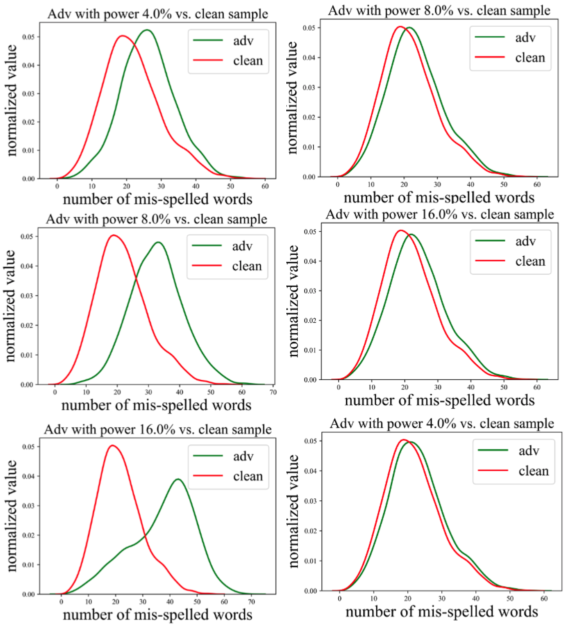


Figure 4: Attack Success Rate Comparison between *Text-FGM* and *DeepWordBug*



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



(a) *Text-FGM*

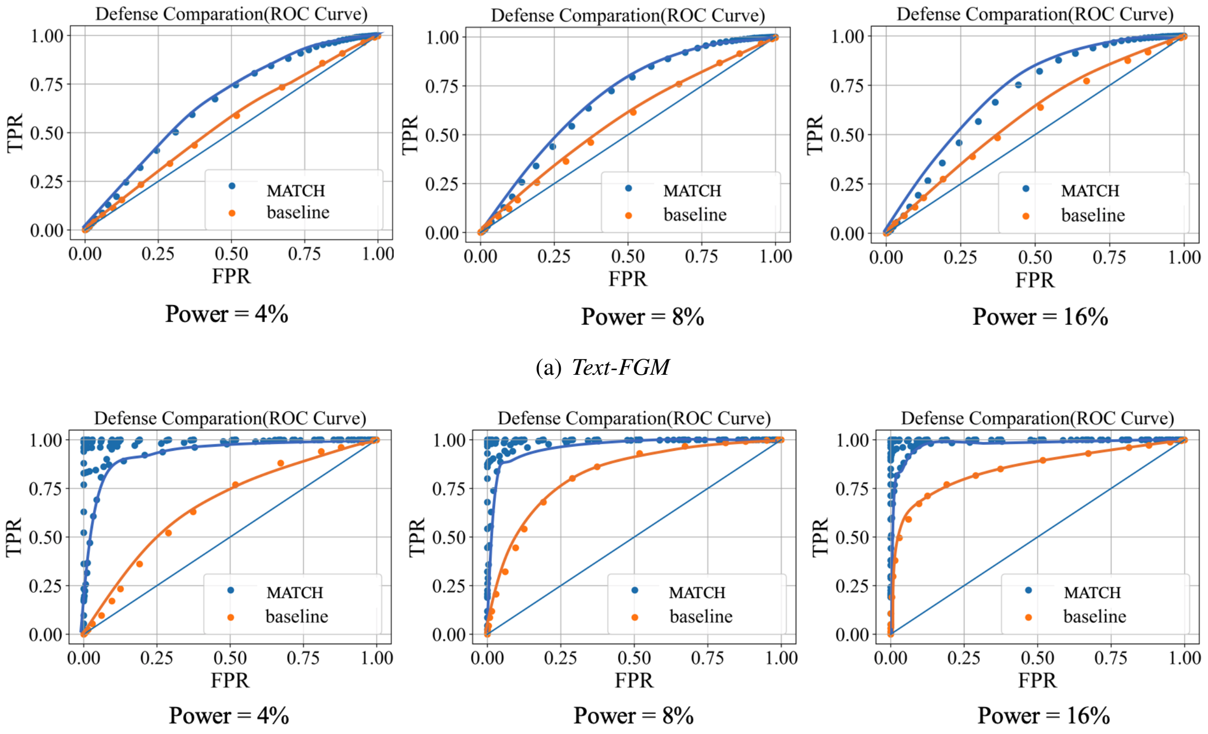
(b) *DeepWordBug*

Figure 6: Distribution of misspelled words in adversarial /clean text under different attack power

4.3 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 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 model is more likely to give a wrong prediction. For *Text-FGM*, it achieves almost 80% attack success rate with only 8% of



(a) *Text-FGM*

(b) *DeepWordBug*

Figure 7: Comparison of the adversarial detection performance between *MATCH* and misspelling check-based defense.

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 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, 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 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.

4.4 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 (Gao et al., 2018). As shown in Figure 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

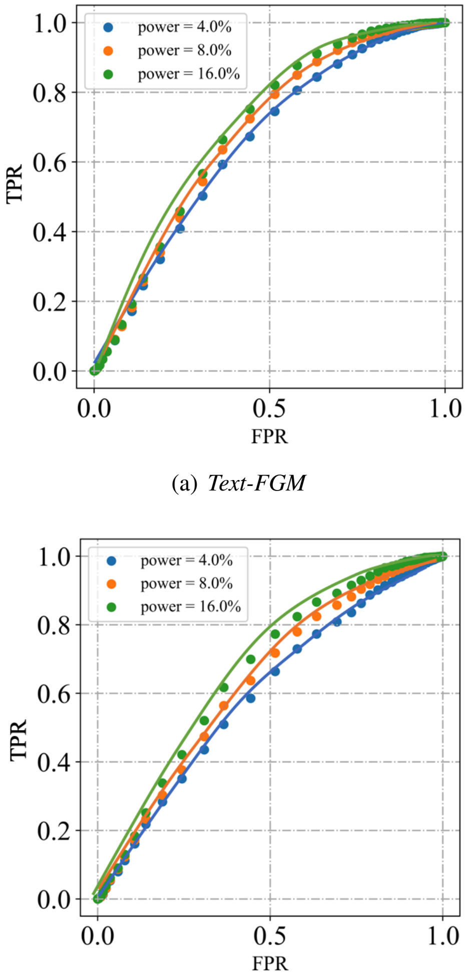
Table 1: Comparison of the Adversarial Detection Accuracy

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

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 misspelling 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 mis-spelling 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 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 widely 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 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 1 show that the accuracy of *MATCH* is much higher than AT and *No Defense*.

Impact of attack power. To better illustrate the impact of attack power, we plot the results of varying attack powers in Figure 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.



(a) *Text-FGM*

(b) *DeepWordBug*

Figure 8: Detection Result

5 Conclusion

In this work, we proposed *MATCH*, a novel defense method by taking advantage of another modal's properties to detect adversarial examples on clinical notes. We evaluated our approaches with two different attack strategies: *Text-FGM* and *DeepWordBug*. We conducted experiments on the 30-day readmission prediction task by detecting adversarial examples in text modalities and use numerical modality to do the multi-modal consistency check. Our experiments showed the effectiveness of *MATCH* compared to the baseline methods.

Although we only evaluated *MATCH* on clinical deep learning system and only attack on the clinial text modality, we believe *MATCH* would be a general framework that could work on any multi-modality dataset. In the future, it would be interesting to extending and evaluating the framework for different modalities such as image and audio. Besides, a more complex architecture may be applied to project extracted features.

References

Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2890–2896.

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

Joan Bruna, Christian Szegedy, Ilya Sutskever, Ian Goodfellow, Wojciech Zaremba, Rob Fergus, and Dumitru Erhan. 2013. Intriguing properties of neural networks.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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.

Mohammad Esmaeilpour, Patrick Cardinal, and Alessandro Lameiras Koerich. 2019. A robust approach for securing audio classification against adversarial attacks. *IEEE Transactions on Information Forensics and Security*.

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

Zhitao Gong, Wenlu Wang, Bo Li, Dawn Song, and Wei-Shinn Ku. 2018. Adversarial texts with gradient methods. *arXiv preprint arXiv:1801.07175*.

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

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

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

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

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. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3:160035.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

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

Volodymyr Kuleshov, Shantanu Thakoor, Tingfung Lau, and Stefano Ermon. 2018. Adversarial examples for natural language classification problems.

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

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. 2017. Deep text classification can be fooled. *CoRR*, abs/1704.08006.

Yu-Wei Lin, Yuqian Zhou, Faraz Faghri, Michael J Shaw, and Roy H Campbell. 2019. Analysis and prediction of unplanned intensive care unit readmission using recurrent neural networks with long short-term memory. *PloS one*, 14(7).

Xingjun Ma, Yuhao Niu, Lin Gu, Yisen Wang, Yitian Zhao, James Bailey, and Feng Lu. 2020. Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, page 107332.

Riccardo Miotto, Li Li, Brian A Kidd, and Joel T Dudley. 2016. Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. *Scientific reports*, 6:26094.

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

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. 2016. 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.

Shilin Qiu, Qihe Liu, Shijie Zhou, and Chunjiang Wu. 2019. Review of artificial intelligence adversarial attack and defense technologies. *Applied Sciences*, 9(5):909.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1085–1097.

Benjamin Shickel, Patrick James Tighe, Azra Bihorac, and Parisa Rashidi. 2017. 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.

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

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

Wenjie Wang, Pengfei Tang, Li Xiong, and Xiaoqian Jiang. 2020. Radar: Recurrent autoencoder based detector for adversarial examples on temporal ehr.

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

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

Ye Xue, Diego Klabjan, and Yuan Luo. 2019. Predicting icu readmission using grouped physiological and medication trends. *Artificial intelligence in medicine*, 95:27–37.

Tahmina Zebin and Thierry J Chaussalet. 2019. Design and implementation of a deep recurrent model for prediction of readmission in urgent care using electronic health records. In *2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–5. IEEE.

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