MedAttacker: Exploring Black-Box Adversarial Attacks on Risk Prediction Models in Healthcare

Muchao Ye[1], Junyu Luo[1], Guanjie Zheng[2], Cao Xiao[3], Houping Xiao[4], Ting Wang[1], Fenglong Ma[1]
[1]Pennsylvania State University, USA, [2]Shanghai Jiao Tong University, China
[3]Relativity, USA, [4]Georgia State University, USA
[1]{muchao, junyu, ting, fenglong}@psu.edu, [2]gjzheng@sjtu.edu.cn, [3]cao.xiao@relativity.com, [4]hxiao@gsu.edu

2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) | 978-1-6654-6819-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/BIBM55620.2022.9994898

Abstract—**Researchers have conduct adversarial attacks against deep neural networks (DNNs) for health risk prediction in the white/gray-box setting to evaluate their robustness. However, since most real-world solutions are trained by private data and released as black-box services on the cloud, we should investigate their robustness in the black-box setting. Unfortunately, existing work ignores to consider the uniqueness of electronic health records (EHRs). To fill this gap, we propose the first black-box adversarial attack method against health risk prediction models named MedAttacker to investigate their vulnerability. It addresses the challenges brought by EHRs via two steps: hierarchical position selection which selects the attacked positions in a reinforcement learning (RL) framework and substitute selection which identifies substitutes with a score-based principle. Particularly, by considering the temporal context inside EHRs, MedAttacker initializes its RL position selection policy by using the contribution score of each visit and the saliency score of each code, which can be well integrated with the deterministic substitute selection process decided by the score changes. We evaluate MedAttacker by attacking three advanced risk prediction models in the black-box setting across multiple real-world datasets, and MedAttacker consistently achieves the highest average success rate and even outperforms a recent white-box EHR adversarial attack technique in certain cases.**

I. Introduction

The increasingly accumulated electronic health record (EHR) data have advanced the field of health analytics, especially the *health risk prediction* [1]–[3] task, which aims to predict future health status of patients according to their historical EHR data and nowadays is commonly conducted by DNNs. Existing studies mainly explore the robustness of deep health risk prediction models by white/gray-box adversarial attacks [4]–[6], which assume attackers can access the parameters of health risk prediction models. However, in the real world health analytics companies train their models with their private data and release them as *black-box* services on the cloud. Therefore, the assumptions of white/gray-box settings are often invalid in real-world practice because the parameters of proprietary models of companies are inaccessible. Thus, it is desirable to have a black-box adversarial attack method for understanding their robustness.

However, this task is challenging due to the unique structure of EHR data. Compared to text data which is also discrete, EHR is different in its *unordered* diagnosis codes within each visit and the *incidental* nature of hospital visits. That is, each longitudinal EHR is a sequence of patient visits that show their evolving conditions as well as sporadic incidences, where the diagnosis codes of each visit are unordered. Thus, this technical challenge raises the question of how to alleviate these limitations to design a black-box adversarial attack approach against health risk prediction models that takes the properties of EHRs into consideration.

To solve these challenges, in this paper we propose a new black-box adversarial attack method named MedAttacker to explore the robustness of health risk prediction models, as shown in Figure 1. To cope with the unique EHR structure, MedAttacker adopts an approach that bridges score-based and reinforcement learning attacks, which has two steps including **hierarchical position selection** and **substitute selection**. It attacks the risk prediction models by taking the temporal context into consideration, and it can search better globally optimized adversarial examples by adopting a hybrid framework of reinforcement learning and score-based principles.

II. Related Work

Adversarial attack research on EHR data is still in the early stage. Existing progress is that [4] proposes a white-box adversarial attack method for the EHR data that are described by continues values including vital signs and lab measurements, while [5] conducts white-box and gray-box adversarial attacks on the ICD-based EHR data. Additionally, [6] tests an orthogonal matching pursuit-guided method for white-box evasion attack on the discrete EHR data. Nonetheless, the early work neglects the black-box adversarial attack setting, which is more realistic and challenging. Among the usually seen data, text data is the most relative one to the EHR data because the search space of EHR and text data are both discrete. Thus, black-box text adversarial attack methods can be used as baselines in our experiments, including DeepWordBug [7], TextBugger [8], PWWS [9] and a reinforcement learning method [10], which can be categorized into score-based methods and RL ones. As for our work, MedAttacker aggregates the temporal context into the reinforcement learning to make it fits for EHR data, and it can be regarded as a hybrid method of the score-based and the RL ones.

978-1-6654-6819-0/22/$31.00 ©2022 IEEE

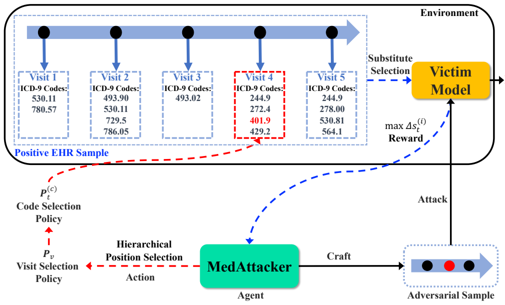

Fig. 1: Overview of MedAttacker.

III. Methodology

A. Problem Definition

In our work, the EHRs of all patients are encoded by a high dimensional dictionary called ICD-9, (International Classification of Diseases, Ninth Revision) and each symptom or abnormal finding is encoded into a unique code. Mathematically, for a specific patient whose EHRs are denoted as $\mathbf{V}$, $\mathbf{V}$ is in the form of $[\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_T]$, where $\mathbf{v}_t$ $(1 \leq t \leq T)$ represents the result of visit t, and T is the total number of visits. Each individual visit $\mathbf{v}_t = [c_1, c_2, \cdots, c_{n_t}]$ includes n_t diagnosis codes encoded by the ICD-9 system.

Problem 1 **(EHR Adversarial Attack).** Let F denote the health risk prediction DNN model. Given the input $\mathbf{V}$ of the patient and the corresponding ground truth label $y \in \mathcal{Y} = \{0, 1\}$, where $y = 1$ represents that patient will suffer from the target disease as a positive case and a negative one otherwise, in the training phase F is trained to generate a prediction score $\hat{y}$ that is as close as to y. Suppose that we have a test sample $\mathbf{V}_{\text{test}}$ whose ground truth label is y_{test}. If prediction $\hat{y} = y_{\text{test}}$, the target of adversarial attack is adding a perturbation $\Delta\mathbf{V}_{\text{test}}$ to construct the adversarial example $\mathbf{V}'_{\text{test}} = \mathbf{V}_{\text{test}} + \Delta\mathbf{V}_{\text{test}}$ such that $\mathbf{V}'_{\text{test}}$ can fool the victim model. That is, the perturbation makes the predicted label change to,

$$\hat{y}' = F(\mathbf{V}'_{\text{test}}) \neq y_{\text{test}}, \tag{1}$$

where the perturbation $\Delta\mathbf{V}_{\text{test}}$ should be as small as possible and is restricted by $||\Delta\mathbf{V}_{\text{test}}|| < \epsilon$. We denote $||\Delta\mathbf{V}_{\text{test}}||$ as the number of diagnosis code changes because EHR data is in a discrete space and ϵ as the maximum allowed attacks.

B. Proposed Method

As shown in Figure 1, the proposed method for adversarial EHR example generation in the black-box setting includes two steps, i.e., hierarchical position selection and substitute selection. In the first step, it frames the position selection as a policy learned through reinforcement learning (RL). In this formulation, the agent is MedAttacker, the environment consists of the EHR sample $\mathbf{V}$ and victim model F, and the state s is represented by the EHR sample. Suppose there are M learning episodes to update the policy parameters, in each episode it will take several steps of actions. Due to the hierarchical characteristics of EHR data, i.e., *code* → *visit* → *EHR*, MedAttacker will select the attacked visit firstly and the attacked diagnosis code within the visit later. They are then grouped as the action a taken by the agent, and the policy is parameterized as $\mathbf{\Theta}$.

Hierarchical Position Selection. The first step for MedAttacker to generate an adversarial example is to select the position of the attacked diagnosis code in a hierarchical way: *selecting the attacked visit firstly and then deciding the attacked position within the selected visit.* Besides, we adopt a RL framework to select the attacked position for it enables the adversarial example generation to be a stochastic process instead of a deterministic one, which allows us to better approximate the globally optimized adversarial example. Thus, in our RL framework, the action is represented by two sets of parameters, and it is updated by the policy gradient [11] framework. Without the loss of generality, suppose we have a positive test sample $\mathbf{V}$ as shown in Figure 1, which can be correctly predicted by the trained model F, and $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_T]$ has information of T visits. For the i-th visit $\mathbf{v}_t$, it has n_t codes. Thus, the parameters to be learned include $\mathbf{p}_{\text{v}} = [p_1^{(\text{v})}, ..., p_T^{(\text{v})}]$, which is the probability distribution of selecting the **visit position**, and a group of parameters $\mathbf{p}_{\text{c}} = \{\mathbf{p}_1^{(\text{c})}, ..., \mathbf{p}_T^{(\text{c})}\}$, which is the probability distribution of selecting the **code position** when the visit position is determined. For each $\mathbf{p}_t^{(\text{c})} \in \mathbf{p}_{\text{c}}$, $\mathbf{p}_t^{(\text{c})} = [p_1^{(t)}, ..., p_{n_t}^{(t)}]$, where n_t is the number of codes in the t-th visit. Thus, the policy parameters to be learned in the policy gradient framework are $\mathbf{\Theta} = \{\mathbf{p}_{\text{v}}, \mathbf{p}_{\text{c}}\}$. In a learning episode, MedAttacker selects the attacked visit $\mathbf{v}_t$ by sampling from $\mathbf{p}_{\text{v}}$, and it then decides the attacked position by sampling from $\mathbf{p}_t^{(\text{c})}$.

Substitute Selection. After we sample from the policy parameters $\mathbf{\Theta}$ and get the position that we are going to attack, the next step is to select a substitute to replace the attacked diagnosis code and generate the adversarial example. Suppose that the code to be attacked is c_i from visit $\mathbf{v}_t$, and we denote $\mathbb{S}$ as the set of substitute codes.[1] We then use the score changes brought by the substitutes to determine the substitute c'_i for c_i. That is, for each substitute code $c \in \mathbb{S}$, we can calculate the replacement score change by Eq. (2),

$$\Delta\mu_t^{(i)} = F((\mathbf{V} - c_i) \cup c) - F(\mathbf{V}), \tag{2}$$

where $((\mathbf{V} - c_i) \cup c)$ represents the EHR sample where c_i is replaced by c in the attacked position. After obtaining all $\Delta\mu_t^{(i)}$ scores for every c in set $\mathbb{S}$, we determine the best substitute code as $c'_i = \arg\max_{c \in \mathbb{S}} \Delta\mu_t^{(i)}$, where c'_i is the code that we will finally employ to replace the attacked diagnosis code c_i. In this step, we determine the substitute code by the score change instead of sampling by RL for it will be difficult to only use the $\max \Delta\mu_t^{(i)}$ to update position selection and substitute selection policy parameters simultaneously.

Policy Update. Now the final design problem is how to update the policy parameters. For parameter initialization, we find

[1] We define set $\mathbb{S}$ as the set of codes in the same ICD-9 category of c_i as the semantic constraints.

that using the **contribution scores** and **saliency scores** is a better way to initialize the policy $\mathbf{p}_\mathrm{v}$ and code saliency scores rather than using the uniform distribution [10] for it takes temporal context into consideration. Thus, we first calculate the output score given by the trained model F when input is $[\mathbf{v}_1, ..., \mathbf{v}_t]$ and then calculate the output score when the input is $[\mathbf{v}_1, ..., \mathbf{v}_{t-1}]$. Next, the contribution score ξ_t is given by their difference, $\xi_t = F([\mathbf{v}_1, ..., \mathbf{v}_t]) - F([\mathbf{v}_1, ..., \mathbf{v}_{t-1}])$, where ξ_t indicates how much information that the whole visit of $\mathbf{v_t}$ can contribute to improving the predicted score given the context $[\mathbf{v}_1, ..., \mathbf{v}_{t-1}]$. By using the normalized $[\xi_1, ..., \xi_T]$ to initialize $\mathbf{p}_\mathrm{v}$, the temporal context is utilized for determining the attacked position. As for the saliency score, for the i-th code c_i in visit $\mathbf{v}_t$, we define the saliency score as $\xi_t^{(i)} = F(\mathbf{V}) - F(\mathbf{V} - c_i)$, where $(\mathbf{V} - c_i)$ denotes the incomplete EHR data where code c_i is removed. If score $\xi_t^{(i)}$ is high, it indicates that attacking c_i can bring more salient influence. Thus, we initialize each $\mathbf{p}_t^{(c)}$ as the normalized $[\xi_t^{(1)}, ..., \xi_t^{(n_t)}]$, which fits with the unordered property of EHRs.

Reward Calculation. Our solution utilizes $\max \Delta\mu_t^{(i)}$ as the reward r to update the policy parameters $\mathbf{\Theta}$, which enables us to integrate the position selection and substitute selection together and help MedAttacker effectively find out the positions useful for adversarial example generation. Thus, in each learning episode, the total rewards of the adversarial example generation process is $J(\mathbf{\Theta}) = \mathbb{E}(\sum_{\ell=0}^{\epsilon-1} \gamma^\ell r_\ell | \mathbf{\Theta})$, where r_ℓ is the reward attained in the step ℓ, and $\gamma \in [0, 1]$ is the discount factor set to be 0.95. We can update $\mathbf{\Theta}$ by the policy gradient method, in which the gradient of $J(\mathbf{\Theta})$ can be approximated by the REINFORCE algorithm [11].

IV. Experiments

A. Experimental Setup

1) Datasets and Victim Models: In our experiments, we use three real-world medical claim datasets, including heart failure, kidney disease and dementia, which are collected by a health information technology company. Since we are conducting adversarial attacks on EHR data, we select three representative DNNs designed for risk prediction task as the victim models in the adversarial attacks, which are Retain [2], SAnD [3] and HiTANet [12]. Code is available at https://github.com/machinelearning4health/MedAttacker.

2) Baselines: Since we are the first to work on black-box adversarial attack on EHRs, most baselines that are used in the experiments are originally designed for the text adversarial attack. In our experiments, we use six baselines, including a naive approach and five black-box adversarial attack methods as follows:(1) Random, a naive method which randomly selects the attacked positions and substitutes; (2) DeepWordBug [7]; (3) TextBugger [8]; (4) PWWS [9] and its varient (5) PWWS-Saliency which only uses the saliency score to determine which word to be attacked; and (6) RL-Attack [10]. In addition, we also use the state-of-the-art **white-box EHR adversarial attack method LAVA** [5] as a baseline to compare the attack effect between black-box adversarial attack frameworks and white-box ones.

3) Implementation: The reinforcement learning environment in our MedAttacker is implemented in the OpenAI Gym package, and the learning rate of the policy parameters is 1×10^{-3}. When implementing the algorithm, we set the hyperparameter $l = 500$. The set of $\mathbb{S}$ for each code is made up of the codes in the same ICD category by the Clinical Classification Software-DIAGNOSES. We set size of $\mathbb{S}$ no more than 10 for efficiency reason.

4) Evaluation Metrics: The first metric is **the number of successful attacks** that each method make, and dividing it by the size of test set can get the **success rate** [8].

B. Performance Evaluation

Since there is always physical restriction on the access of EHR data in real-life attacks, we validate the performance of models under the following restriction: (1) The maximum number of visits per patient in the test set is 20, and (2) the maximum allowed attacks $\epsilon = 5$. The comparison of adversarial attack results between the proposed MedAttacker and baseline methods are shown in Table I. We can see that MedAttacker achieves the best performance over three datasets against three different victim models in 6 out of 9 cases, which demonstrates that MedAttacker has the best generalization ability compared to existing black-box adversarial attack techniques. Besides, MedAttacker always has the highest success rate in the heart failure case against different victim models and constantly achieves the best attack success rate when the victim model is HiTANet. The results above show that the integral design of reinforcement learning and score-based principles can empower MedAttacker to have good ability of generalization.

C. Comparison with White-box Attack

To find the performance gap between black-box adversarial attacks and white-box ones, we also employ LAVA as a baseline of white-box adversarial attack. The experimental results are listed in Table II. We compare them in the case of Retain owning to the availability of official implementation codes of LAVA. Because white-box ones have the knowledge of gradients, it is not surprising to see that LAVA has better performance on the datasets of heart failure and kidney disease from Table II. But compared to LAVA, MedAttacker can still have 72.77% and 81.47% adversarial attack effects on heart failure and kidney disease datasets, respectively. Moreover, MedAttacker can have better attack results on the dementia dataset. This indicates that for certain cases, black-box adversarial attack techniques have the potential to achieve better results against white-box ones.

D. Case Study

We also include a case study to further illustrate the adversarial attack process conducted by MedAttacker. Given a positive EHR sample in Figure 2 in the heart failure dataset whose predicted score of being a positive case by HiTANet is 0.70, MedAttacker selects code “305.00” (*alcohol abuse*) from visit 2, code “724.2” (*lumbago*) and code “496” (*chronic*

TABLE I: Comparison on the number of successful attacks and success rate against different victim models. The first row in each block is the number of successful attacks, and the second row is the success rate.

Dataset	Heart Failure				Kidney Disease				Dementia			
Method	HiTANet	Retain	SAnD	Average	HiTANet	Retain	SAnD	Average	HiTANet	Retain	SAnD	Average
Random	30	18	7	18.3	21	10	32	21.0	24	18	10	17.3
	(1.62%)	(0.97%)	(0.38%)	(0.99%)	(1.25%)	(0.59%)	(1.90%)	(1.25%)	(1.68%)	(1.26%)	(0.70%)	(1.21%)
TextBugger	216	119	4	113.0	182	138	104	141.3	117	109	6	77.3
	(11.69%)	(6.44%)	(0.22%)	(6.11%)	(10.79%)	(8.19%)	(6.17%)	(8.38%)	(8.18%)	(7.62%)	(0.42%)	(5.40%)
DeepWordBug	231	113	12	118.7	248	96	92	145.3	147	87	15	83.0
	(12.50%)	(6.11%)	(0.65%)	(6.42%)	(14.71%)	(5.69%)	(5.46%)	(8.62%)	(10.27%)	(6.08%)	(1.05%)	(5.80%)
PWWS-Saliency	277	129	48	151.3	264	98	229	197.0	277	166	66	169.7
	(14.99%)	(6.98%)	(2.60%)	(8.19%)	(15.66%)	(5.81%)	(13.58%)	(11.68%)	(19.36%)	(11.60%)	(4.61%)	(11.86%)
PWWS	369	162	52	194.3	332	**154**	**239**	241.7	359	204	**77**	213.3
	(19.97%)	(8.77%)	(2.81%)	(10.52%)	(19.69%)	**(9.13%)**	**(14.18%)**	(14.33%)	(25.09%)	(14.26%)	**(5.38%)**	(14.91%)
RL-Attack	347	146	25	172.7	301	132	142	191.7	272	160	30	154.0
	(18.78%)	(7.90%)	(1.35%)	(9.34%)	(17.85%)	(7.83%)	(8.42%)	(11.37%)	(19.01%)	(11.18%)	(2.10%)	(10.76%)
MedAttacker	**426**	**166**	**64**	**218.7**	**369**	149	218	**245.3**	**384**	**210**	63	**219.0**
	(23.05%)	**(8.98%)**	**(3.46%)**	**(11.83%)**	**(21.89%)**	(8.84%)	(12.93%)	**(14.55%)**	**(26.83%)**	**(14.68%)**	(4.40%)	**(15.30%)**

TABLE II: Comparison with LAVA on success rate.

Victim Model	Retain		
Method	Heart Failure	Kidney	Dementia
LAVA	**12.34%**	**10.85%**	11.39%
MedAttacker	8.98%	8.84%	**14.68%**

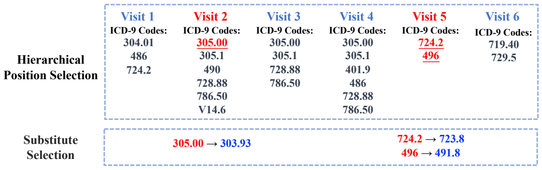

Fig. 2: Illustration of case study.

airway obstruction) from visit 5 to construct the adversarial example. Furthermore, since the substitute set of each code is restricted within the same category in the ICD-9 coding system, the semantic in the adversarial example is similar to the original one. For instance, substitute "723.8" (*cervical syndrome*) and the original code "724.2" are both in the category of "Spondylosis" in the ICD-9 coding system, which can ensure the adversarial example still looks reasonable by humans and victim models. After the attack, HiTANet downgrades the predicted score to 0.33 and predicts it as a negative case, which shows the effectiveness of MedAttacker.

V. Conclusion

Although researchers have investigated their vulnerability by the white/gray-box adversarial attacks, a more realistic black-box setting has not been explored yet for the robustness of risk prediction models. To increase the momentum in this field, in this paper, we introduce a black-box adversarial attack framework named MedAttacker to explore the robustness of health risk prediction models, which is inspired by the score-based and reinforcement learning methods in black-box adversarial attacks. It is more suitable for EHR data because it takes the temporal context of EHR into consideration, and the stochastic position selection and deterministic substitute selection processes can help it better approximate the generation of globally optimized adversarial examples.

Acknowledgement

T. Wang is partially supported by the National Science Foundation (NSF) under Grant No. 1953893, 1951729, 2119331, and 2212323. F. Ma is supported in part by the NSF under Grant No. 2212323. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] F. Ma, Q. You, H. Xiao, R. Chitta, J. Zhou, and J. Gao, "Kame: Knowledge-based attention model for diagnosis prediction in healthcare," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018, pp. 743–752.

[2] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, "Retain: An interpretable predictive model for healthcare using reverse time attention mechanism," in *NeurIPS*, 2016, pp. 3504–3512.

[3] H. Song, D. Rajan, J. Thiagarajan, and A. Spanias, "Attend and diagnose: Clinical time series analysis using attention models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[4] M. Sun, F. Tang, J. Yi, F. Wang, and J. Zhou, "Identify susceptible locations in medical records via adversarial attacks on deep predictive models," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 793–801.

[5] S. An, C. Xiao, W. F. Stewart, and J. Sun, "Longitudinal adversarial attack on electronic health records data," in *The World Wide Web Conference*, 2019, pp. 2558–2564.

[6] Y. Wang, Y. Han, H. Bao, Y. Shen, F. Ma, J. Li, and X. Zhang, "Attackability characterization of adversarial evasion attack on discrete data," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 1415–1425.

[7] J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi, "Black-box generation of adversarial text sequences to evade deep learning classifiers," in *2018 IEEE Security and Privacy Workshops*. IEEE, 2018, pp. 50–56.

[8] J. Li, S. Ji, T. Du, B. Li, and T. Wang, "Textbugger: Generating adversarial text against real-world applications," in *26th Annual Network and Distributed System Security Symposium*, 2019.

[9] S. Ren, Y. Deng, K. He, and W. Che, "Generating natural language adversarial examples through probability weighted word saliency," in *Proceedings of the 57th annual meeting of the association for computational linguistics*, 2019, pp. 1085–1097.

[10] Y. Zang, B. Hou, F. Qi, Z. Liu, X. Meng, and M. Sun, "Learning to attack: Towards textual adversarial attacking in real-world situations," *arXiv preprint arXiv:2009.09192*, 2020.

[11] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[12] J. Luo, M. Ye, C. Xiao, and F. Ma, "Hitanet: Hierarchical time-aware attention networks for risk prediction on electronic health records," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 647–656.