Hindawi
Wireless Communications and Mobile Computing
Volume 2020, Article ID 8859489, 10 pages
https://doi.org/10.1155/2020/8859489

Research Article

Neural Model Stealing Attack to Smart Mobile Device on Intelligent Medical Platform

Liqiang Zhang[1], **Guanjun Lin**[2], **Bixuan Gao**[1], **Zhibao Qin**[1], **Yonghang Tai**[1] and **Jun Zhang**[1]

[1]*Yunnan Key Laboratory of Opto-Electronic Information Technology, Yunnan Normal University, Kunming 650000, China*
[2]*The School of Information Engineering, Sanming University, Sanming, Fujian 365004, China*

Correspondence should be addressed to Yonghang Tai; taiyonghang@126.com and Jun Zhang; junzhang@ynnu.edu.cn

Received 6 August 2020; Revised 8 October 2020; Accepted 22 October 2020; Published 26 November 2020

Academic Editor: Weizhi Meng

To date, the Medical Internet of Things (MIoT) technology has been recognized and widely applied due to its convenience and practicality. The MIoT enables the application of machine learning to predict diseases of various kinds automatically and accurately, assisting and facilitating effective and efficient medical treatment. However, the MIoT are vulnerable to cyberattacks which have been constantly advancing. In this paper, we establish a MIoT platform and demonstrate a scenario where a trained Convolutional Neural Network (CNN) model for predicting lung cancer complicated with pulmonary embolism can be attacked. First, we use CNN to build a model to predict lung cancer complicated with pulmonary embolism and obtain high detection accuracy. Then, we build a copycat model using only a small amount of data labeled by the target network, aiming to steal the established prediction model. Experimental results prove that the stolen model can also achieve a relatively high prediction outcome, revealing that the copycat network could successfully copy the prediction performance from the target network to a large extent. This also shows that such a prediction model deployed on MIoT devices can be stolen by attackers, and effective prevention strategies are open questions for researchers.

1. Introduction

The number of intelligent Medical Internet of Things (MIoT) deployed online has been constantly increasing, reaching 20.35 billion in 2017, and the estimated number will continually increase to 75.44 billion in the next decade [1]. Besides, according to the International Data Corporation (IDC), the last five years have witnessed a 17.0% annual growth rate in IoT spending from approximately $700 billion in 2015 to nearly $1.3 trillion in 2019 [2]. Among them, MIoT accounts for a large proportion. Tan and Varghese [3] pointed out that there is a huge potential for the application of IoT in the health industry. Nevertheless, practical constraints must be taken into consideration. Vicini et al. [4] presented an approach to combine vending machines with IoT technology to facilitate a healthy lifestyle. However, cyberattacks are not new to IoT, leading to terrible consequences [5, 6]. Most of the MIoT are without any defense mechanism. With the widespread application of IoT devices, cyberattacks are also improving, posing a more severe threat to the secure operation not only of IoT devices but also of the entire cyberspace [7, 8].

With an increasing number of IoT-related cyber incidents being reported, experts and researchers from the IoT industry and academia have been working to design secure systems and solutions to combat the attacks of various types [9, 10]. Many researchers have devoted extensive efforts to ensuring MIoT security and privacy, providing practical guidance for MIoT security. Fu et al. [11] highlight both opportunities and possible threats that IoT faces in two important application scenarios—the home and hospital. Yang et al. [12] provide an extensive survey, presenting the

classification of MIoT attacks from perspectives of MIoT security research, threats, and open issues. Boejen and Grau [13] have utilized Unmanned Aerial Vehicles (UAV) to launch an attack in a simulated smart hospital environment and compromise a small collection of wearable healthcare sensors. Sethuraman et al. [14] have proposed a new deep learning approach, DFEL, for real-time cyberattack detection in the IoT environment and presented the robustness of high accuracy and significant time savings.

However, there are not many studies that investigate the attacks targeting the services deployed on the MIoT devices, particularly the MIoT-based AI services, for example, machine learning-based disease prediction/detection services. Unlike the model Mohan [15] has raised, using lightweight encryption and attribute-based authorization to protect the model, in our model when selecting the data set, we used the patient data in a specific area (Yunnan, Chongqing), which greatly reduced the risk of attacking the established network by exploiting the vulnerability of the data set. At the same time, we store the prediction model of lung cancer complicated with pulmonary embolism in the cloud to further protect our model with the protection measures provided by the cloud. In this paper, we study a scenario where a trained Convolutional Neural Network (CNN) [16] model for predicting lung cancer complicated with pulmonary embolism can be stolen by attackers. Specifically, we build a Copycat CNN [17] using only a small amount of data labeled by the original network, aiming to steal the established prediction model. We prove that the stolen model can successfully copy the prediction performance with a minor difference of approximately 3%. By doing this, a prediction model deployed on MIoT devices can be stolen by attackers. Overall, the contributions of our work are as follows:

(1) Create a new platform of surgical IoT for cybersecurity study in high-performance medicine

(2) Propose a model stealing attack on the intelligent medical platform

(3) Implement and evaluate the proposed intelligent medical platform and model stealing attack

This paper is organized as follows: In Section 2, we review the related works focusing on the cyberattacks using deep neural networks for the MIoT. The model stealing attack experiments are designed in the methodology part which is presented in Section 3. In the next section, the evaluation of the attack scheme on the medical platform was demonstrated and discussed. In the last section, we summarize the results and conclude this paper.

2. Related Work

2.1. VR for MIoT. The IoT application has been widely used in the medical industry. In recent years, it has become widespread to combine Virtual Reality (VR) technology with medical-related majors. The integration of the Internet of Things and VR technology in the education field can enable learners to combine their conceptual learning with practical experience in a novel way [18]. Coogan and He use Unity Software, combined with a brain-computer interface, to control the VR environment and MIoT devices [19]. To make the operation of the entire medical platform more transparent, we adopted the combination of VR technology and MIoT to correctly reproduce the prediction process of lung cancer complicated with pulmonary embolism through the medical platform.

2.2. Cyberattacks with Deep Neural Networks. Because the medical concept of the Internet of Things is based on the concept of the Internet of Things, we should also understand the concept of the Internet of Things which was put forward in 1995 by Bill Gates in *The Road Ahead* and in 1999 by Auto-ID who first proposed the "Internet of Things," after the Internet of Things in various fields had a corresponding application, including the medical field. In 2013, Hu and his team [20] had believed that based on the support and guarantee of the powerful Internet of Things technology, the personal networking platform in the medical field will have a strong background shortly. This becomes reality, in 2018, when Jagadeeswari et al. [21] proposed a healthcare monitoring system based on big data training on a powerful computing platform. This has proven that the Medical Internet of Things has become a reality. In 2020, due to more and more cyberattacks, Flynn et al. [22] provided a proof of concept that the MIoT device and its accompanying smartphone app are vulnerable to attacks. A recent survey on Android malware detection is provided in [23]. This provides a certain theoretical basis for our attack model. The emerging deep learning techniques have shown impressive performance in various fields, from tasks like speech and object recognition to natural language processing (NLP), and even to cybersecurity tasks such as bug and vulnerability detection [24, 25]. Nevertheless, the deep learning technologies can easily be fooled by crafted adversarial examples, which have brought considerable attention since 2014 when Szegedy et al. [26] and follow-up studies [27, 28] showed that imperceptibly perturbed input images could successfully fool deep networks. Subsequently, Dalvi et al. [29] and Lowd and Meek [30, 31] investigate the carefully crafted adversarial samples which can fool linear classifiers in the context of spam email detection. In 2006, Barreno et al. [32] pointed out that machine learning algorithms can be targets of a malicious adversary, and deep learning algorithms are no exception. When it comes to the investigation of attacks to deep models using grey-box models, Papernot et al. [33] applied a grey-box target deep neural network (DNN) using the MNIST database. They use crafted adversarial samples against the target DNN, aiming to craft adversarial examples by approximating the decision boundaries of the target DNN. Subsequently, Bapiyev et al. [34] have demonstrated that one of the most promising approaches to the development of detection systems of network cyberattacks improved their software by application of modern models based on deep neural networks. And the results of model testing have shown that the accuracy of the basic variant is comparable

Table 1: Three different types of network attacks (white box attack, grey box attack, and black box attack) have been compared in detail. The enumerated expressions D represent the training data, the feature set x, the learning algorithm f, and its trained parameters/hyperparameters ω.

Name	The knowledge of the attacker	Formula expression
White box	Perfect knowledge	$\theta = (D, x, f, \omega)$
Grey box	Limited knowledge	$\theta = (\widehat{D}, x, f, \widehat{\omega})$
Black box	Zero knowledge	$\theta = (\widehat{D}, \widehat{x}, \widehat{f}, \widehat{\omega})$

with the accuracy of modern detection systems of network cyberattacks.

Table 1 shows different D, x, f, ω in different formula expressions, which represents different levels of knowledge of the attacker. Compared with white-box attacks, grey-box attacks show differences in enumerated expression D and trained parameters/hyperparameters ω, which are understood in the literature as unknown parameters. It can be concluded from the formula of a black-box attack that we do not know everything about the original network when carrying out the black-box attack. In our attack network, a grey-box attack is adopted. Based on the same data set selection interval, relatively reasonable data labels can be obtained by doing so while ensuring accuracy.

In this paper, we examine a copy attack using a CNN (which we call a copycat network, a grey-box attack) to copy information from another CNN (the target network) in a disease prediction scenario. By leveraging a small number of data labeled the target network, the copycat network could obtain similar performance compared with the target network, showing that the MIoT-based prediction model is vulnerable to adversarial attacks.

3. New Platform for Mobile and Intelligent Medicine

3.1. MIoT System Design. Unity Software is a multiplatform integrated game development tool that allows players to easily create interactive content such as 3D video games, architectural visualization, and real-time 3D animation. This is a fully integrated professional game engine. The core code of the Unity engine itself is written in the underlying language C/C++. The image, sound, and physics engines are all compiled in C++. The dynamic link library DLL file encapsulates a series of methods and classes. C#, Python, and other programs call corresponding methods and classes through DLL files to build the game flexibly and with superior performance. Unity can run across platforms, such as Android, IOS, PC, and Web. This article is for the Android platform. Unity will publish the APK file of the VR project to the Android device and then display it through the headset. Unity will publish the APK file of the VR project to the VR headset and display this scene. The VR headset uses Pico G2 (Beijing Bird-Watch Technology Co., Ltd.) mobile VR headset, which has a field of view of 101°, refresh rate of 90 Hz, and resolution of 3K, providing the wearer with immersive medical VR application scenes (Figure 1).

As shown in Figure 1, the whole MIoT system consists of two parts: The left part is the construction of a three-dimensional lung model, in which three-dimensional voxel segmentation was performed on CT images of patients (lung cancer with pulmonary embolism), and the lesions were marked. The right part processes the patient's textual data and uses LSTM and RNN deep learning model algorithms to predict and classify the data, respectively. A safety module is then added to make up the MIoT system (Visual-Haptic Navigation System).

3.2. A Deep Neural Model for PE&LC Prediction. In this part, we use a CNN to perform the prediction of lung cancer with pulmonary embolism (LC&PE).

As we can see from Figure 2, our CNN-Net architecture contains two 1D convolution layers and two full-connection layers and connects to a sigmoid activation layer. Every 1D convolution layer is equipped with a kernel the size of which is 3, followed by a LeakyReLU activation layer and a max pool layer with a stride of 2 to downsample the text. Between two full-connection layers (one has the input size of 320 and the output size of 120; another one has the input size of 120 and the output size of 2), there is a LeakyReLU activation layer. Finally, we use a sigmoid neuron as a classifier.

We use the convolution layer to extract features from the data. The output value of the layer with input size (N, C_{in}, L) and output (N, C_{out}, L) can be precisely described as

$$\text{out}(N, C_{\text{out}}) = bias(C_{\text{out}}) + \sum_{k=0}^{C_{\text{in}}} weight(C_{\text{out}}, k) * input(N, k) \tag{1}$$

where N is the batch size, C denotes the number of channels, and L is a length of the signal sequence.

When $\text{groups} = \text{in}_{\text{channels}}$ and $\text{out}_{\text{channels}} = k * \text{in}_{\text{channels}}$, where k is a positive integer. This kind of operation is also called deep convolution in the literature.

For an input of size $(N, C_{\text{in}}, L_{\text{in}})$, depth convolution with depth multiplier can be constructed by parameters $C_{\text{in}} = C_{\text{in}}$, $C_{\text{out}} = C_{\text{in}} * k, \cdots, \text{groups} = C_{\text{in}}$ input: $(N, C_{\text{in}}, L_{\text{in}})$, output: $(N, C_{\text{out}}, L_{\text{out}})$ where

$$L_{\text{out}} = \frac{L(L_{\text{in}} + 2 * \text{padding} - \text{dilation} * (\text{kernel.size} - 1) - 1)}{\text{stride}} + 1. \tag{2}$$

4. Model Stealing Attack to the New Platform

4.1. Overview of the Threat Model. As we can see (Figure 3), the MIoT structure consists of three layers (the perception layer, the network layer, and the application layer). Healthcare data with a variety of devices have been mainly collected in the perception layer. The network layer is composed of a wireless system, which processes and transmits the input obtained by the perception layer with the support of the

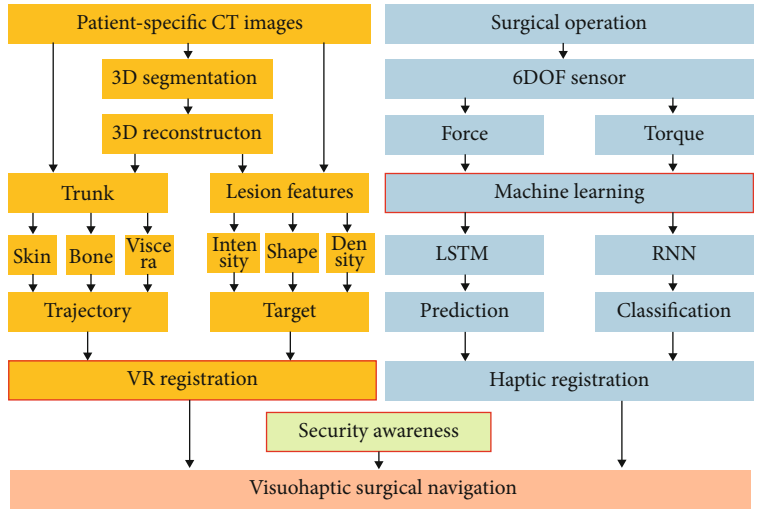

Figure 1: The structure and workflow of the proposed medical platform.

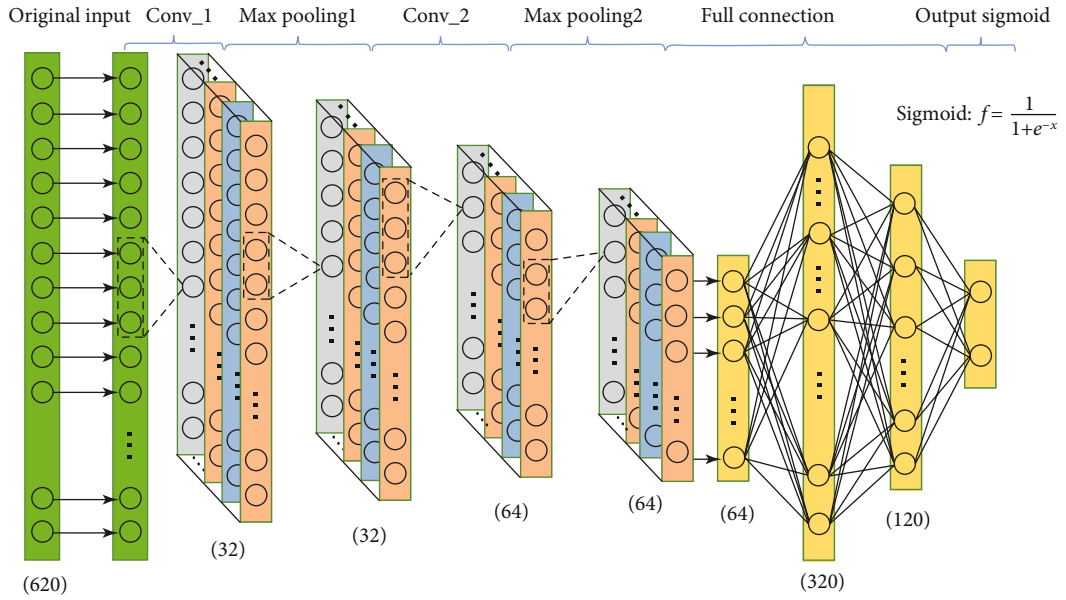

Figure 2: The Convolutional Neural Network (CNN) with 4 layers.

technology platform. According to the actual situation and service needs of the target population, the medical information resources are integrated at the application layer to provide personalized medical services to meet the needs of end users.

Dividing MIoT into these three levels enables a more thorough analysis of where the network is at risk. In the perception layer, Wang et al. put forward the concept of the input formed by applying small but intentionally worst-case perturbations to examples in the data set; by doing this, they can output an incorrect answer with high confidence [10]. In the network layer, we can steal the model already trained by others for higher business value, which can greatly reduce the investment in the early stage of research and development and obtain higher profits.

4.2. Theoretical Description of the Model Stealing Attack. In this part, we will introduce how to build our imitation network (copycat network) using data stolen from an existing target network (CNN in this case). The whole process of stealing is mainly to use random natural data to steal a network of imitators from the existing target network. It mainly includes two steps, creating pseudo training data and training a network of imitators. In the first step, a target network is used as a grey box to mark random natural data to generate a pseudo data set. Then, this pseudo data set is used to train an imitation network to replicate the property of the target network.

A data set is needed to train the imitation network (Figure 4). We recommend using pseudo data sets extracted from the target network (including text data related to or

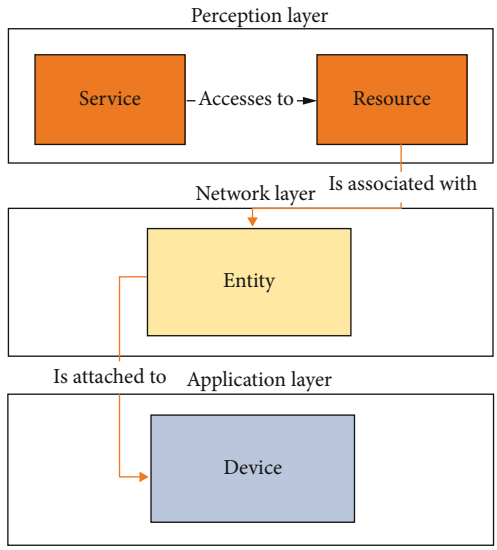

Figure 3: Structure of Medical Internet of Things.

not related to the problem domain (PD)). Therefore, the pseudo data set is completely different from the original data set. When performing a steal operation, the target network receives text data as input and affords class tags as output. The data set can be composed of the same PD as the target network, or it can be composed of random natural text data. First, we assume that the attacker has text data in the same PD as the training target network. Second, we suppose that the attacker can only access publicly available large-scale data sets, but in our research, the original labels are considered irrelevant. When automatically labeling these data sets (PD and/or nonproblem domain (NPD)), the target network is used by the attacker. Another type of network can be trained with labeled pseudo data sets, hoping to capture the nuances of the characteristic regions, to achieve property close to the target network. Achieving this hypothesis is mainly based on adding imperceptible noise to the input text data of CNN to obtain an answer from the network in a certain direction. The NPD can be achieved from the Internet for free. Then, when disposing of small databases (for example, PD data sets), the data expansion process can help increase the size of the database to obtain better results.

Once a pseudo data set is obtained, the simulation network can start training. Firstly, a model architecture must be chosen as an attacker to mimic. Note that the attacker performing the replication may not know the target network's model architecture, but it makes no difference. We use a well-known architecture (CNN architecture) to compare with the original network. CNN is created for classification, so its output layer can be set according to specific problems. For the attacker, this may also be the case of the chosen architecture, i.e., imitating the target network. So, the output of the selected model must be adapted to the target network's PD; the output number of the replicator must match the number of classes processed by the derivation of the target network.

The purpose of this simulated network is to evaluate whether the proposed method can replicate the target model with a small set of text data set in the same PD. In this case, we assume that the attacker can access a small amount of data in the same domain but without labels. Therefore, the samples of this data set contain text data set of the same PD as the original data set but are marked by the target network.

The transferability of adversarial samples is accurately defined. We suppose an opponent is interested in producing a misclassified adversarial sample $\vec{x^*}$ that is different from the class assigned to the legal input $\vec{x}$ by the model. This can be achieved by solving the following optimization problem:

$$\vec{x^*} = \vec{x} + \delta_{\vec{x}} \text{ where } \delta_{\vec{x}} = \arg\min_{\vec{z}} f\left(\vec{x} + \vec{z}\right) \neq f\left(\vec{x}\right). \quad (3)$$

To mislead the sample $\vec{x^*}$, the model f was calculated deliberately. However, as mentioned earlier, such adversarial samples are often misclassified by models f' other than f in practice. To facilitate discussion, we formalize the concept of transferability of adversarial samples as

$$\Omega_X\left(f, f'\right) = \left|\left\{f'\left(\vec{x}\right) \neq f'\left(\vec{x} + \overrightarrow{\delta_{\vec{x}}} : \vec{x} \in X\right)\right\}\right|. \quad (4)$$

The set X represents the expected input distribution of the tasks solved by model f and model f'. We divide the adversarial sample transferability into two variables to characterize the pair of models (f, f'). First is the transferability within technology, which defines transferability between training models of the same machine learning technology with different parameter initializations or data sets (for example, f and f' are both neural networks or both decision trees). Second is crosstechnology transferability, which considers using models trained by two technologies (for example, f is a neural network and f' is a decision tree).

4.3. Discussion on the Specific Medical Scenario and the Attack. Lung cancer with pulmonary embolism accounts for a large proportion of medical mortality, a large part of which is due to errors in the diagnosis of patients with lung cancer with pulmonary embolism. Our system, after several training steps, can predict accurately whether a lung cancer patient will have pulmonary embolism at the same time.

This would allow doctors to have an accurate diagnosis of the patient and develop a suitable plan to reduce the mortality rate. The system is of great value both medically and economically. However, this system can be vulnerable to attacks. The attack we designed was to steal a trained model. In today's increasingly important intellectual property, attacks of such kind can severely damage the profit of the model owner, causing the leak of patients' privacy. In this paper, we implement a copycat model to steal a trained model for predicting lung cancer with a pulmonary embolism network and demonstrate the feasibility of successfully copying the performance of a trained model.

The data set we use consists of 179 lung cancer patients with pulmonary embolism, 1372 lung cancer patients without pulmonary embolism, and 71 samples randomly collected from natural data which have been used to create the original data set (the size of which is 1622). Among the total

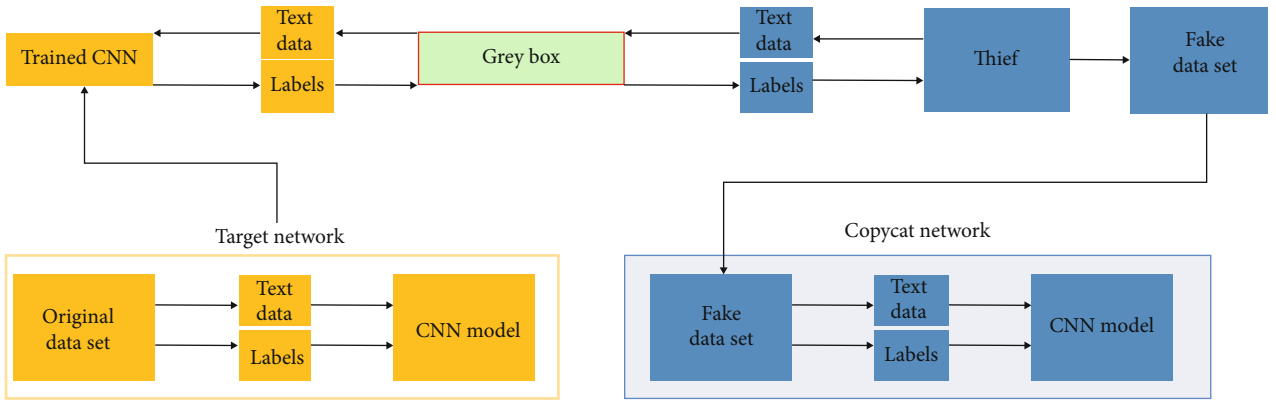

Figure 4: On the left side, the target network is trained by an original data set and is available as an API, input text data, and output class labels. The right side shows the process of obtaining stolen tags and creating pseudo data sets: sending a random natural text data set to the API to obtain tags. Then, this pseudo data is used set to train the imitation network.

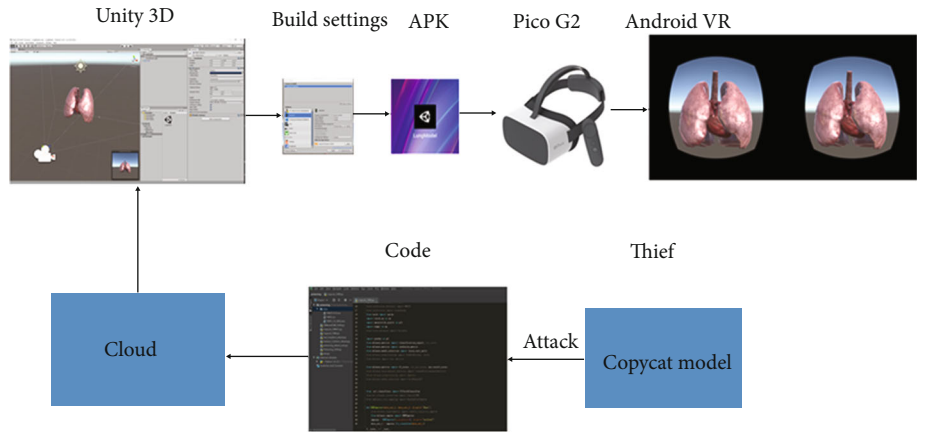

Figure 5: It describes how to steal the target network by using the existing model and introduces the steps of 3D reconstruction of the segmented CT images by using the Unity 3D platform.

number of 1622 patient samples, 60% of the samples were used as a training set and 40% as a test set. As a result, our system predicted lung cancer with pulmonary embolism with a precision of 79.43%.

5. Experiments and Results

5.1. Implementation of the Platform and the Attack. Unity's release of the VR project to the Android platform process is shown in Figure 5. As shown in Figure 5, the overall display is the process of a copycat model attacking the medical prediction model of lung cancer with pulmonary embolism. In this process, the copycat model plays the role of a thief. The prediction model of lung cancer with pulmonary embolism established by us is stored in the cloud. First, we determine the network model used by a copycat, build the model through code compilation software, and then reuse the input following the original model input requirements of the data set, stealing useful labels for us to use to generate the copycat network. To make the whole prediction result more convenient for observation, we used the Unity 3D platform for 3D modelling to generate a 3D lung. First, we used the code to isolate the lesion area in the CT image of the patient and generated a file in the form of OBJ, which was imported into the Unity 3D platform for modelling. The upper part of the figure shows the 3D modelling process. In contrast, the lower part shows the whole process of the copycat model attacking the prediction model of cloud lung cancer combined with pulmonary embolism. The whole framework shows the process of the copycat model attacking MIoT.

The prepared data set has been imported into the target network stored in the cloud; at the same time, the label corresponding to our data set is also output together. The network we selected was trained through data sets and stolen tags. During the training, the parameters and hyperparameters in the network were constantly fine-tuned so that the copycat network and the target network were continuously fitted to achieve similar effects, which meant that our attack was successful.

5.2. Performance of Intelligent Medical Platform. We use the confusion matrix as the evaluation standard of the intelligent medical platform. In the prediction analysis, the confusion table, sometimes called a confusion matrix, is a two-row,

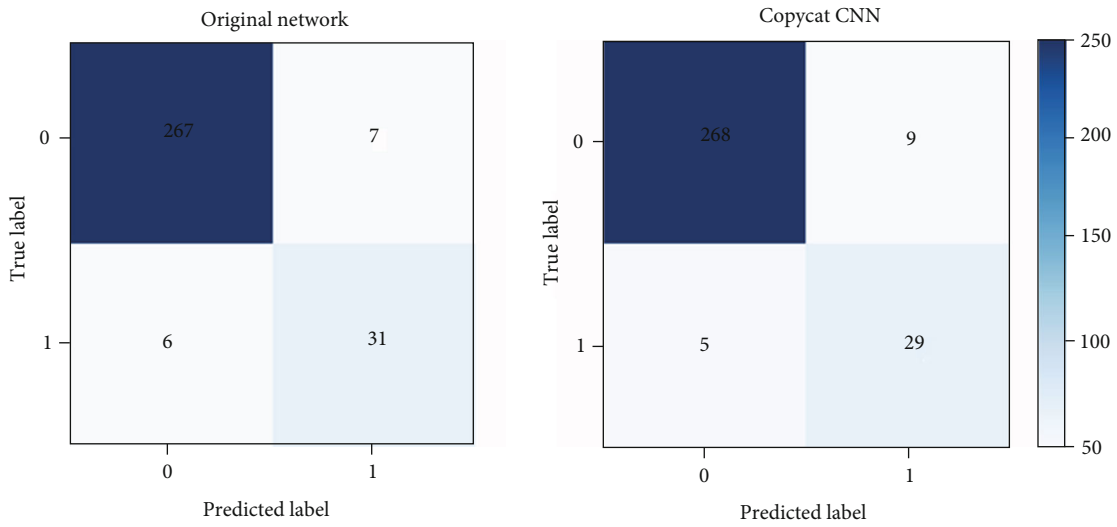

Figure 6: The confusion matrix of LC&PE's prediction.

Table 2: Values of different indicators based on the source model.

Object	Precision	Recall	F1_score
JC	0.97	0.98	0.98
JC&PE	0.84	0.82	0.83
Macro avg	0.91	0.90	0.90
Weighted avg	0.96	0.96	0.96
Accuracy	—	—	0.96

Table 3: List of the performance metrics of the Copycat CNN.

Object (copycat)	Precision	Recall	F1_score
JC	0.97	0.99	0.98
JC&PE	0.91	0.79	0.85
Macro avg	0.94	0.89	0.91
Weighted avg	0.96	0.96	0.96
Accuracy	—	—	0.96

two-column table composed of TP (True Positive), FN (False Negative), FP (False Positive), and TN (True Negative). It allows us to do more analyses, not just to get it right. The following expressions are the application of different parameters in the obfuscation matrix:

$$\begin{aligned} \text{Acc} &= \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}, \\ \text{Rec} &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{Pre} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ f_1 &= 2 * \frac{\text{Pre} * \text{Rec}}{\text{Pre} + \text{Rec}}. \end{aligned} \tag{5}$$

In the predictive classification model, the quantity of TP and TN is large, while the quantity of FP and FN is small, which means the prediction accuracy is higher (which can be seen from Figure 6). However, what is counted in the confusion matrix is the number. Sometimes, faced with a large amount of data, it is difficult to measure the number of models by counting. Therefore, the confusion matrix is an extension of the secondary and tertiary indicators in the basic statistical results (obtained by adding, subtracting, multiplying, and dividing the lowest indicators).

Table 4: List of the absolute values indicating the performance variation between the original network and the imitator network after training.

Object (copycat)	Precision	Recall	F1_score
JC	0	0.01	0
JC&PE	0.07	0.03	0.02
Macro avg	0.03	0.01	0.01
Weighted avg	0	0	0
Accuracy	—	—	0

Therefore, after we obtain the confounding matrix of lung cancer with pulmonary embolism, we need to see how many observed values correspond to the second and fourth quadrants, where the value (267 + 31 = 298) takes up a large proportion in the total (311), which means that our prediction model is effective.

Macro average means to average the recall of class 1 and the recall of class 0. The weighted average is calculated using the proportion of samples as the weight. From the table above, our model has high prediction accuracy. From Table 2, we can see that our model has achieved a very high precision.

5.3. Effectiveness of Model Stealing Attack. We trained a CNN to predict LC&PE, using an adaptive learning rate of 1*e*-4, which is then reduced based on the smooth behavior of the verification loss. Other hyperparameters include the batch size of 8, the number of instances (T) set to 200 (unless otherwise specified), the Adam optimizer with a weight of 0.01,

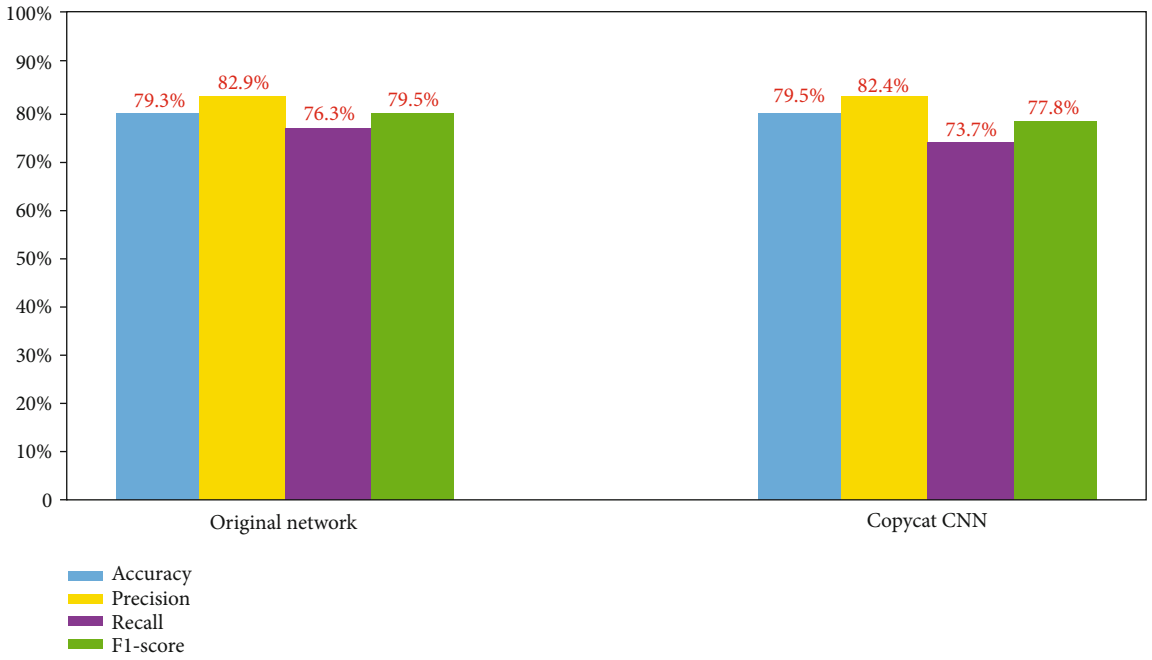

Figure 7: Comparison of different results between the original prediction model and the copycat model. As shown in the figure, in terms of the precision/recall, the performance variations between the Copycat CNN and the original network range from 2.6% to 0.3%.

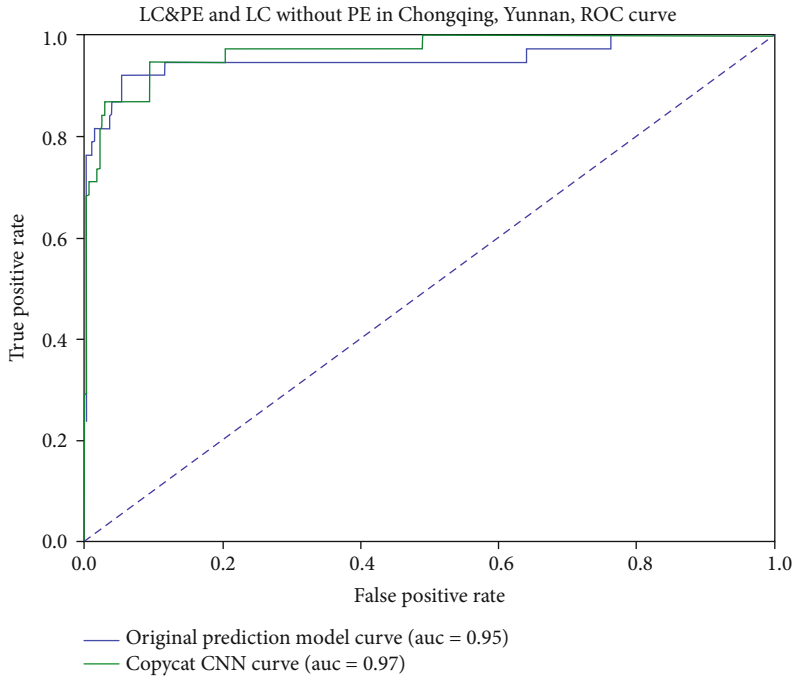

Figure 8: The ROC curve about LC with PE and LC without PE in Chongqing, Yunnan.

and binary crossentropy loss. The implementation is based on Pytorch and uses NVIDIA GTX 1070 GPU.

The Receiver Operating Characteristic (ROC) curve shows the detection capabilities of the trained CNN model and the imitated CNN under different classification thresholds. The abscissa of the plane is the false positive rate (FPR), and the ordinate is the true positive rate (TPR). For the classifier, we can get the TPR and FPR point pairs according to the performance of the classifier on the test sample.

As can be seen, Table 3 lists the performance metrics of the Copycat CNN and Table 4 lists the absolute values indicating the performance difference variation between the original network and the imitator network after training. Combined with the data in Tables 3 and 4, we can see that

the copycat model can achieve high accuracy in stealing the prediction model of lung cancer with pulmonary embolism, which is almost the same. And from the figure, we can see that Figure 7 describes the absolute value of the difference between the original network and the imitator network after training, and in terms of the precision/recall, the performance variations between the Copycat CNN and the original network range from 2.6% to 0.3%. Figure 8 shows the ROC curve about LC with PE and LC without PE in Chongqing, Yunnan. Almost the same bar chart and ROC curve close to 1 prove that the copycat network built by us is a model with functions close to the original network with facts. Above, the performance difference between the network stolen from the target medical platform model through the copycat model and the original network is not evident. This means that we can successfully use deep learning models to steal the target network with a small amount of labeled data.

Through the comparison of the data in the experiments we obtained, we can see that the copycat is generally low in various scales with the original network, which, in the prediction accuracy of lung cancer and f1 appeared on the score difference of 0, shows that we can steal out of the network and the gap with the original network has become very small, thus proving that our guess is correct. We may conclude that the prediction results of the copycat model are 99% identical to those of the original model.

6. Conclusions

In this paper, we establish a new platform based on surgical IoT for cybersecurity study. On the established intelligent medical platform, we propose a CNN for lung cancer with pulmonary embolism prediction. To demonstrate the attack to an established model on the surgical IoT platform, we implemented a random selection model that mimics CNN training using a small number of labeled samples. Experimental results show that the replication model can successfully replicate the performance of the target CNN, achieving minor performance variance (less than 3%). The success of the attack shows that intellectual property such as the trained AI model using private and sensitive information can be stolen. How to effectively prevent attacks of such kind from happening is an open question for researchers from the fields of cybersecurity, MIoT, and deep learning.

Data Availability

The data supporting the results of this study can be obtained from the corresponding author.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Authors' Contributions

Liqiang Zhang and Gunjun Lin contributed equally to this paper.

Acknowledgments

We thank Professor Jun Peng of the Yunnan First People's Hospital for the helpful data processing guidance and Xuejuan Wang and Shangjin Lv for collecting the data together. This research is funded by the National Natural Science Foundation of China (61741516) and the National Science Foundation of Yunnan Province, China (ZD2014004) of Yunnan Key Laboratory of Optoelectronic Information Technology, Kunming, China.

References

[1] W. Zhou, Y. Jia, A. Peng, Y. Zhang, and P. Liu, "The effect of IoT new features on security and privacy: new threats, existing solutions, and challenges yet to be solved," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1606–1616, 2019.

[2] A. Sheth, "Internet of things to smart IoT through semantic, cognitive, and perceptual computing," *IEEE Intelligent Systems*, vol. 31, no. 2, pp. 108–112, 2016.

[3] V. Tan and S. A. Varghese, "IoT-enabled health promotion," in *Proceedings of the First Workshop on IoT-enabled Healthcare and Wellness Technologies and Systems*, pp. 17-18, New York, NY, USA, 2016.

[4] S. Vicini, S. Bellini, A. Rosi, and A. Sanna, "Well-being on the go: an IoT vending machine service for the promotion of healthy behaviors and lifestyles," in *International Conference of Design, User Experience, and Usability. Springer*, Berlin, Heidelberg, 2013.

[5] M. Abomhara, Department of Information and Communication Technology, University of Agder, Norway, G. M. Køien, and Department of Information and Communication Technology, University of Agder, Norway, "Cyber security and the internet of things: vulnerabilities, threats, intruders and attacks," *Journal of Cyber Security and Mobility*, vol. 4, no. 1, pp. 65–88, 2015.

[6] L. Liu, O. de Vel, Q.-L. Han, J. Zhang, and Y. Xiang, "Detecting and preventing cyber insider threats: a survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 1397–1417, 2018.

[7] R. Coulter, Q.-L. Han, L. Pan, J. Zhang, and X. Yang, "Data-driven cyber security in perspective—intelligent traffic analysis," *IEEE Transactions on Cybernetics*, vol. 50, no. 7, pp. 3081–3093, 2020.

[8] R. Coulter, Q.-L. Han, L. Pan, J. Zhang, and Y. Xiang, "Code analysis for intelligent cyber systems: a data-driven approach," *Information Sciences*, vol. 524, pp. 46–58, 2020.

[9] G. Lin, S. Wen, Q. L. Han, J. Zhang, and Y. Xiang, "Software vulnerability detection using deep neural networks: a survey," *Proceedings of the IEEE*, vol. 108, no. 10, pp. 1825–1848, 2020.

[10] M. Wang, T. Zhu, T. Zhang, J. Zhang, S. Yu, and W. Zhou, "Security and privacy in 6G networks: new areas and new challenges," *Digital Communications and Networks*, vol. 6, no. 3, pp. 281–291, 2020.

[11] K. Fu, T. Kohno, D. Lopresti et al., "Safety, security, and privacy threats posed by accelerating trends in the internet of things," *Computing Community Consortium (CCC) Technical Report*, vol. 29, no. 3, 2017.

[12] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issues in internet-of-things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1250–1258, 2017.

[13] A. Boejen and C. Grau, "Virtual reality in radiation therapy training," *Surgical Oncology*, vol. 20, no. 3, pp. 185–188, 2011.

[14] S. C. Sethuraman, V. Vijayakumar, and S. Walczak, "Cyber attacks on healthcare devices using unmanned aerial vehicles," *Journal of Medical Systems*, vol. 44, no. 1, p. 29, 2020.

[15] A. Mohan and Cyber security for personal medical devices internet of things, "IEEE International Conference on Distributed Computing in Sensor Systems," *IEEE*, vol. 2014, pp. 372–374, 2014.

[16] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," *Icdar*, vol. 3, no. 2003, 2003.

[17] J. R. Correia-Silva, R. F. Berriel, C. Badue, A. F. de Souza, and T. Oliveira-Santos, "Copycat CNN: stealing knowledge by persuading confession with random non-labeled data," *2018 International Joint Conference on Neural Networks (IJCNN)*, 2018.

[18] F. Mohamed, J. Abdeslam, and E. B. Lahcen, "Towards new approach to enhance learning based on internet of things and virtual reality," in *Proceedings of the International Conference on Learning and Optimization Algorithms: Theory and Applications*, 2018.

[19] C. G. Coogan and B. He, "Brain-computer interface control in a virtual reality environment and applications for the internet of things," *IEEE Access*, vol. 6, pp. 10840–10849, 2018.

[20] F. Hu, D. Xie, S. Shen, and On the application of the internet of things in the field of medical and health care, "IEEE international conference on green computing and communications and IEEE Internet of Things and IEEE cyber, physical and social computing," *IEEE*, vol. 2013, pp. 2053–2058, 2013.

[21] V. Jagadeeswari, V. Subramaniyaswamy, R. Logesh, and V. Vijayakumar, "A study on medical internet of things and big data in personalized healthcare system," *Health information science and systems*, vol. 6, no. 1, p. 14, 2018.

[22] T. Flynn, G. Grispos, W. Glisson, and W. Mahoney, "Knock! Knock! Who is there? Investigating data leakage from a medical internet of things hijacking attack," in *Proceedings of the 53rd Hawaii International Conference on System Sciences*, 2020.

[23] J. Qiu, J. Zhang, L. Pan, W. Luo, S. Nepal, and X. Yang, "A survey of Android malware detection with deep neural models," *ACM Computing Survey*, 2020.

[24] J. Gu, Z. Wang, J. Kuen et al., "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.

[25] N. Sun, J. Zhang, P. Rimba, S. Gao, Y. Xiang, and L. Y. Zhang, "Data-driven cybersecurity incident prediction: a survey," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 2, pp. 1744–1772, 2019.

[26] C. Szegedy, W. Zaremba, I. Sutskever et al., "Intriguing properties of neural networks," *arXiv preprint arXiv*, 2013, 1312.6199.

[27] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv*, 2014, 1412.6572.

[28] S. M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. , 20162574–2582, 2016.

[29] N. Dalvi, P. Domingos, S. Sanghai et al., "Adversarial classification," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 99–108, 2004.

[30] D. Lowd and C. Meek, "Adversarial learning," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pp. 641–647, 2005.

[31] D. Lowd and C. Meek, "Good word attacks on statistical spam filters," *CEAS.*, vol. 2005, 2005.

[32] M. Barreno, B. Nelson, R. Sears, A. D. Joseph, and J. D. Tygar, "Can machine learning be secure," in *Proceedings of the 2006 ACM Symposium on Information, computer and communications security*, pp. 16–25, 2006.

[33] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "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.

[34] I. M. Bapiyev, B. H. Aitchanov, I. A. Tereikovskyi et al., "Deep neural networks in cyber attack detection systems," *International Journal of Civil Engineering and Technology (IJCIET)*, vol. 8, no. 11, pp. 1086–1092, 2017.