Hindawi
Discrete Dynamics in Nature and Society
Volume 2020, Article ID 4163825, 9 pages
https://doi.org/10.1155/2020/4163825

*Research Article*

# Deep Learning-Based Network Security Data Sampling and Anomaly Prediction in Future Network

**Lan Liu**[1], **Jun Lin**[2], **Pengcheng Wang**[1], **Langzhou Liu**[1] **and Rongfu Zhou**[1]

[1]*Guangdong Polytechnic Normal University School of Electronic and Information Engineering, Guangzhou 510655, Guangdong, China*
[2]*China Electronic Product Reliability and Environmental Testing Research Institute, Guangzhou 510610, Guangdong, China*

Correspondence should be addressed to Jun Lin; linjun@ceprei.com

Received 16 March 2020; Accepted 23 April 2020; Published 17 May 2020

Guest Editor: Jianbiao Zhang



Based on the design idea of future network, this paper analyzes the network security data sampling and anomaly prediction in future network. Through game theory, it is determined that data sampling is performed on some important nodes in the future network. Deep learning methods are used on the selected nodes to collect data and analyze the characteristics of the network data. Then, through offline and real-time analyses, network security abnormal events are predicted in the future network. With the comparison of various algorithms and the adjustment of hyperparameters, the data characteristics and classification algorithms corresponding to different network security attacks are found. We have carried out experiments on the public dataset, and the experiment proves the effectiveness of the method. It can provide reference for the management strategy of the switch node or the host node by the future network controller.

## 1. Introduction

At present, people attach great importance to the research and application deployment of new technologies and new networks. Scientists are actively exploring the use of technologies such as IPv6, software-defined network (SDN), and 5G to build future networks that meet the requirements of high reliability, low delay, and wide coverage [1]. We need to pay attention to the new features of security events of future network.

5G has brought about massive communications and tens of billions of device access scenarios, all of which require flexible network architecture and high-performance networks. Software defined networking (SDN) is being strongly considered as the next promising networking platform [1, 2]. The logical centralization of network has brought new opportunities and challenges of the field of network security. In future network, the detection and prediction of network data anomaly caused by network malicious attack is an important problem to be solved. Research on the network data sampling strategy and the appropriate anomaly detection model of network security event in the future network has guiding significance for preventing future network. In this paper, we design and simulate a kind of network data sampling strategy of SDN using zero-sum game. After those steps, we can find out some important nodes to protected. And then, we intend to use the method of deep-learning to establish and analyze the network anomaly flow in future network.

The remainder of the paper is organized as follows. Section 2 summarizes the background and related work of deep learning-based network security data sampling and anomaly prediction in future network. In Section 3, we introduce the sampling model of SDN security data and the method of deep learning-based security flow anomaly prediction in detail. Experimental results and comparisons are presented in Section 4. Finally, conclusion is given in Section 5.

## 2. Related Work

*2.1. SDN Network Architecture and Security Data Sampling Model.* In recent years, Major mainstream manufacturers

have begun to deploy SDN networks. Many commercial cases have been applied. For example, Google built a B4 [3] network based on SDN to transform its network; Cimorelli [4] propose a distributed load balancing algorithm based on game theory to balance the traffic of the controller cluster. Abraxas of Switzerland adopted Huawei's SDN-based data center network solution to build a virtualized multitenant cloud data center network. In order to provide users with a better experience, Tencent use SDN to achieve differentiated path differentiation calculation and flow control. And, in the development of Internet communication technology in the coming decades, SDN also has broad prospects for development.

SDN is based on the granularity of data flow control, so that it does not understand the internal information of the data stream, which makes SDN vulnerable to attacks by Trojan, worms, spam, etc [5]. In order to ensure the security of the network, it is necessary to detect packets in the future network. Lan [6] propose a dynamic model with a time-varying community network, inspired by research models on the spread of epidemics in complex networks across communities. The results may help to decide the SDN control strategy to defend against network malware and provide a theoretical basis to reduce and prevent network security incidents.

Data packet sampling under limited network resources is necessary to reduce latency, improve the network bandwidth, and ensure network security of future network at the same time. Afek [7] present techniques for traffic sampling and large flows detection in SDN with OpenFlow. They make use of the sampling mechanisms for the development of an efficient method to detect large flows. Tang [8] propose an efficient sampling and classification approach with the two-phase elephant flow detection. They demonstrate their system can provide accurate detection with less sampled packets and short detection time. Aiming at the problem of existing flow statistical sampling in anomaly detection, the authors [9–11] analyze the distortion cause that packet sampling and time domain polymerization lead to flow record time series in theory. They propose different methods to solve it. Result shows that their methods can reduce impact of sampling rate on the signal to noise ratio and improve the performance of the anomaly detection.

Zero-sum game is a concept of game theory and it is a noncooperative game. As its model is relatively simple, a zero-sum game model can be built between the attacker and the defender in network attack and defense [12]. When the attacker attacks successfully, the attacker gains positive scores, while the defender gains negative scores, and the sum of the two is zero. In network attack and defense model, both attack and defense resources are limited. By quantifying network nodes and allocating the corresponding profit value, the game model of attack and defense is established, and we can improve the defense capability, reduce the attack loss, and find a reasonable packet sampling strategy in the future network.

*2.2. Deep Learning and Anomaly Detection.* As an important subfield of machine learning, deep learning has made breakthroughs in many artificial intelligence fields, such as speech recognition, computer vision, autonomous driving, and natural language processing [13]. Data flow in future network is usually high dimensional and heterogeneous. Deep learning can learn different levels of features from a large number of raw network data streams, and these automatic learning features do not require the domain knowledge of human experts, saving a lot of labor and time costs. We take these learned important features as the input of machine learning algorithm to complete the classification task, which can solve the problem of false alarm rates (FAR) and false positives (FP) of the intrusion detection system (IDS) in the future network security and realize the identification of network traffic [14].

In recent years, some scholars have introduced the method of deep learning into the field of network security [15–18]. They used convolutional neural networks (CNNs) to learn the spatial characteristics of network traffic and used the method of image classification to identify malicious network traffic. Recurrent neural network (RNN) is used to learn the temporal characteristics of network traffic and identify the traffic characteristics to improve the detection rate.

In the deep learning [19–21], CNNs have obtained good performance and wide application in the field of computer vision, and the recognition of handwritten numbers has achieved an extremely low false positive rate on the MNIST test set. The long short-term memory (LSTM) improves the original RNN algorithm [22–24], solves the problem of gradient disappearance or gradient explosion after training of time series modeling, and conducts deep learning through long-term state preservation and forward calculation and uses the back-propagation algorithm to train time series to establish the prediction model [25, 26].

## 3. Models and Methods

When the network is attacked in future network, we need to have a certain strategy, as soon as it is possible to find the existence of the attack and obtain the attack category and location information. The SDN controller is used to allocate defense resources according to the importance of nodes under the condition of limited defense resources to reduce network losses.

For the important nodes selected from the model, the spatial-temporal characteristics of network traffic are learned by combining CNN and LSTM in deep learning, so as to realize abnormal detection of network traffic. The processing process consists of three parts. First, the advantages of CNN in spatial feature extraction of image processing are utilized, and the spatial feature training is carried out after the network traffic data are processed graphically to form a traffic spatial classification model. Secondly, the traffic vectors processed by CNN are processed in time series, and the time characteristics of the traffic are learned through LSTM to form a traffic time feature recognition model. Then, combining spatial classification model and temporal feature recognition model, the current network traffic is classified. The model is shown in Figure 1.

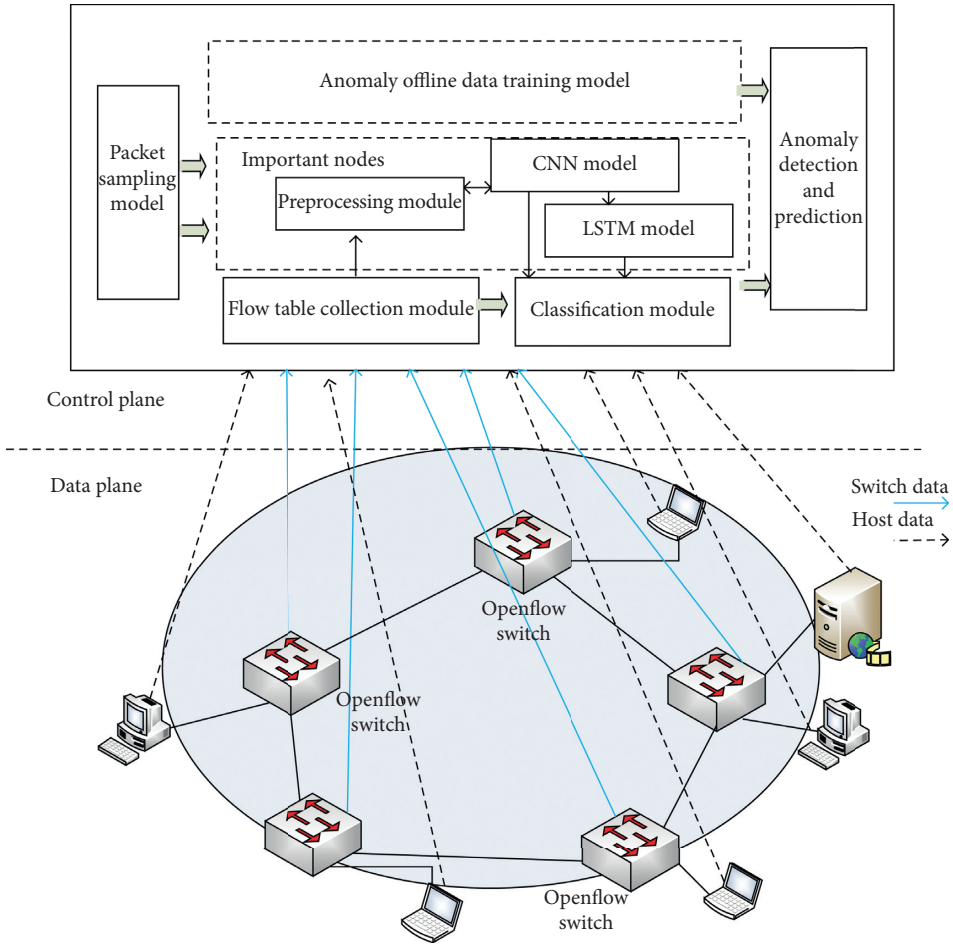


Figure 1: Deep learning-based network security data sampling and anomaly prediction diagram.

*3.1. Packet Sampling Model of Future Network.* For the sampling and classification of future network security data, the analysis is carried out from the aspects of SDN attack loss calculation method, node importance calculation, attack and defense strategy game model analysis, etc.

*3.1.1. Attack Loss Score Calculation.* The attacker's behavior can be seen as sending attack packets from a controlled computer to one or more network devices. When the defender nodes checked the attack packet by sampling strategy, the attack will fail and the defender will get a positive score; otherwise, the defender will get a negative score. The attacker sends packets from one network device to one or more network devices; if the packet is not intercepted by the defender, the attack is successful and the score is positive; otherwise, the attack is considered as a failure and the score is negative.

Based on the above background, the following hypotheses are considered:

*Hypothesis 1.* Under the limited defensive resource constraints, the probability that a defender detects a packet is directly proportional to its importance.

*Hypothesis 2.* Attackers always pursue maximum revenue, so they will prioritize attacks on network devices of high importance.

In the process of attack and defense game, both the attacker and the defender will use the optimal strategy to maximize their own benefits, and the SDN packet sampling problem will be simulated as a zero-sum game in which both sides of the attack and defense participate.

The SDN network is constructed into an undirected graph, and the set of vertices is $V$, the graph of the edge set $E$ is recorded as $G = (V, E)$, and the number of vertices and the number of edges of $G = (V, E)$ are, respectively, $|V|$ and $|E|$. Connect two vertices $u$, and the edges of $v$ are denoted as $e = (u, v)$.

When an attacker launches an attack, the probability of sending an attack packet is proportional to the importance. It is assumed that $k$ packets are extracted for every $n$ packets of the network device of importance $x$ and $m$ packets are included in the $n$ packets. Then, the probability of extracting $k$ out of $n$ packets in $n$ packets is $c_{n-m}^{k}/C_{n}^{k}$; then this is the probability that no attack packets are detected.

For the attacker, the benefit score is

$$U_a = \frac{c_{n-m}^k}{C_n^k} * x. \quad (1)$$

For the defender, the benefit score is

$$U_X = -\frac{c_{n-m}^k}{C_n^k} * x. \quad (2)$$

*3.1.2. Node Importance Calculation.* When an attacker successfully attacks the network node $v_t$, the score that can be obtained is based on the importance $\varphi(v_t)$ corresponding to the node $v_t$, and the attacker tends to attack the higher-priority nodes in the network to cause greater impact on the network. The network node value is quantified according to the importance of the network node, and the higher value is given to the more important network nodes. The nodes in the network are divided into switch nodes $S_k \in$ S, and the host nodes $H_k \in$ H, S, and H are included in N. For the normal operation of the network, the importance of the switch node (Switching device) is equal to the sum of importance value of all the host nodes (Terminal devices) connected to it. The importance of different switches in future network may be different, such as the core switch is more important than the edge switch; there is no difference between hosts. In summary, Theorem 1 and Theorem 2 are proposed.

**Theorem 1.** *The importance value of each S node is divided into direct importance value and indirect importance value.*

**Theorem 2.** *The direct importance value of a S node is equal to the sum importance value of the H nodes which it is connected, and the indirect importance value is equal to the direct importance value of the S node which it is connected.*

**Theorem 3.** *The importance value of each S node may be different, and the importance value of each H node is equal.*

According to Theorem 1–3, the importance value of the S node and the H node is divided. The importance SI value of the switch node is often higher than the importance HI value of the host node. The specific values can be used to represent different network nodes according to different network scenarios, for example, we may set HI value as 1. When SI value and HI value are set, attention is paid to the size relationship between them, that is, the value of S node $SI = \sum_{i=1}^{n} HI$, where $n$ is the H node connected to the S node.

According to Theorem 2, assuming that the importance value of each H node is 1, then the direct importance value of a S node is equal to the sum of all the H nodes connected to it. And, the indirect importance value the S node is equal to the sum of all S nodes connected to it. We add the two values when we calculate the importance value of S node.

*3.1.3. Zero-Sum Game Model of Attack and Defense Strategies.* For an attacker, there are two main attack strategies:

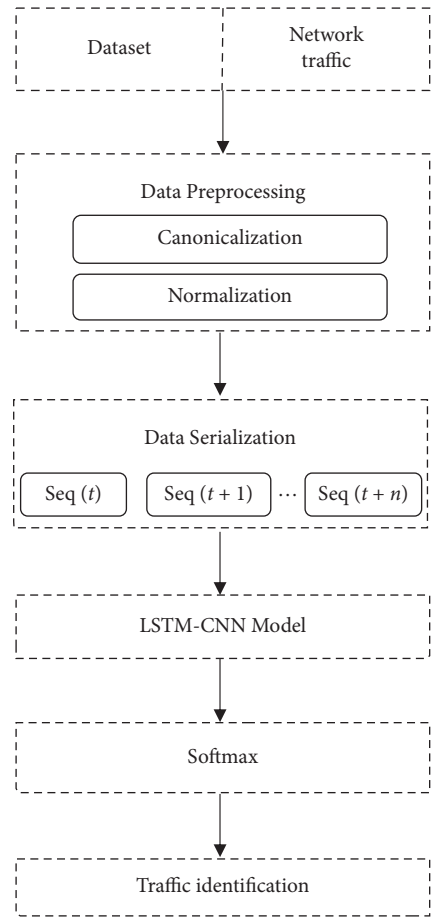


Figure 2: The network anomaly detection model based on deep learning in future network.

Table 1: Network topology.

| | Topology 1 | Topology 2 | Topology 3 | Topology 4 |
|---|---|---|---|---|
| Number of switches | 3 | 1 | 5 | 3 |
| Number of hosts | 4 | 4 | 5 | 5 |
| Number of links | 7 | 4 | 9 | 7 |
| Topology | Tree type | Star type | Line type | Hybrid type |

(1) Sends attack packets to the defender network device on average if the importance of network nodes is unknown

(2) Sends attack packets to the defender network device by its proportion if the importance of network nodes is known.

Suppose an attacker uses attack strategy 1 to distribute attack packets evenly to $n$ network devices, this $n$ is exactly equal to the number of defender network devices. It is assumed that when an attacker uses an attack strategy, it may be randomly assigned to attack a network device of high

Table 2: Defender's detection success rate of attack and score of the attacker.

| Experimental method | | Defensive detection success rate | Attacker score |
|---|---|---|---|
| Topology 1 | Attack strategy 1 vs. defensive strategy 1 | 0.41 | 7.2 |
| | Attack strategy 1 vs. defensive strategy 2 | 0.75 | 6.8 |
| | Attack strategy 2 vs. defensive strategy 1 | 0.75 | 6.3 |
| | Attack strategy 2 vs. defensive strategy 2 | 0.98 | 5.7 |
| Topology 2 | Attack strategy 1 vs. defensive strategy 1 | 0.62 | 3.1 |
| | Attack strategy 1 vs. defensive strategy 2 | 0.96 | 2.1 |
| | Attack strategy 2 vs. defensive strategy 1 | 0.96 | 2.2 |
| | Attack strategy 2 vs. defensive strategy 2 | 0.99 | 2.4 |
| Topology 3 | Attack strategy 1 vs. defensive strategy 1 | 0.19 | 16.1 |
| | Attack strategy 1 vs. defensive strategy 2 | 0.29 | 17.1 |
| | Attack strategy 2 vs. defensive strategy 1 | 0.29 | 17.1 |
| | Attack strategy 2 vs. defensive strategy 2 | 0.39 | 13.7 |
| Topology 4 | Attack strategy 1 vs. defensive strategy 1 | 0.41 | 8.5 |
| | Attack strategy 1 vs. defensive strategy 2 | 0.75 | 7.3 |
| | Attack strategy 2 vs. defensive strategy 1 | 0.75 | 7.3 |
| | Attack strategy 2 vs. defensive strategy 2 | 0.98 | 6.1 |

importance value of defender network or may be randomly assigned to attack a network device of low importance value defender network.

When defenders deal with attackers, there are two main defense strategies:

(1) The probability of network device packet detection is equal

(2) The probability of network device packet detection is directly proportional to its importance value

*3.2. Network Anomaly Detection Based on Deep Learning.* After describing the sampling model in Section 3.1, we find the secure nodes that need sampling in future network. On these nodes, we use the network traffic anomaly detection method based on deep learning and combine CNN and LSTM to detect and classify network security data. The spatial-temporal characteristics of network traffic can be obtained through training, which has great potential to improve the overall performance of network traffic detection technology in future network. The algorithms analyze the possible security events and submit them to the controller of SDN for further analysis and optimization of the whole network.

The network anomaly detection model based on deep learning in future network is shown in Figure 2.

For the data on the important nodes found by the sampling model, the data are firstly preprocessed, including numerical coding and normalization. Then, the preprocessed data were input into the LSTM-CNN model, and the spatial and time feature learning of network traffic were carried out. Finally, the two kinds of neural networks were combined, and the output was classified by Softmax and the attack events were classified.

The experimental process is as follows:

Step 1 open IDS datasets or simulated attacks are used as training datasets, and real-time network traffic is collected as test data

| Network traffic types in CICIDS2017 | |
|---|---|
| Benign | 2273097 |
| DoS hulk | 231073 |
| Portscan | 158930 |
| DDoS | 128027 |
| DoS goldeneye | 10293 |
| FTP patator | 7938 |
| SSH patator | 5897 |
| DoS slowloris | 5796 |
| DoS slowhttptest | 5499 |
| Bot | 1966 |
| Web attack brute force | 1507 |
| Web attack XSS | 652 |
| Infiltration | 36 |
| Web attack sql injection | 21 |
| Heartbleed | 11 |
| *Total* | *2830743* |

Figure 3: Dataset statistics.

Step 2 data preprocessing is carried out, and the flow data after feature extraction is numerically coded and feature normalized

Step 3 the preprocessed data were coded with one-hot coding, the matrix was converted into $m \times m$ traffic images, and the image data were classified through the CNN neural network

Step 4 the preprocessed data were divided into time series and trained by LSTM neural network to obtain the abnormal flow probability of the next period.

Finally, the two training models are combined to predict and identify the current network traffic and realize the real-

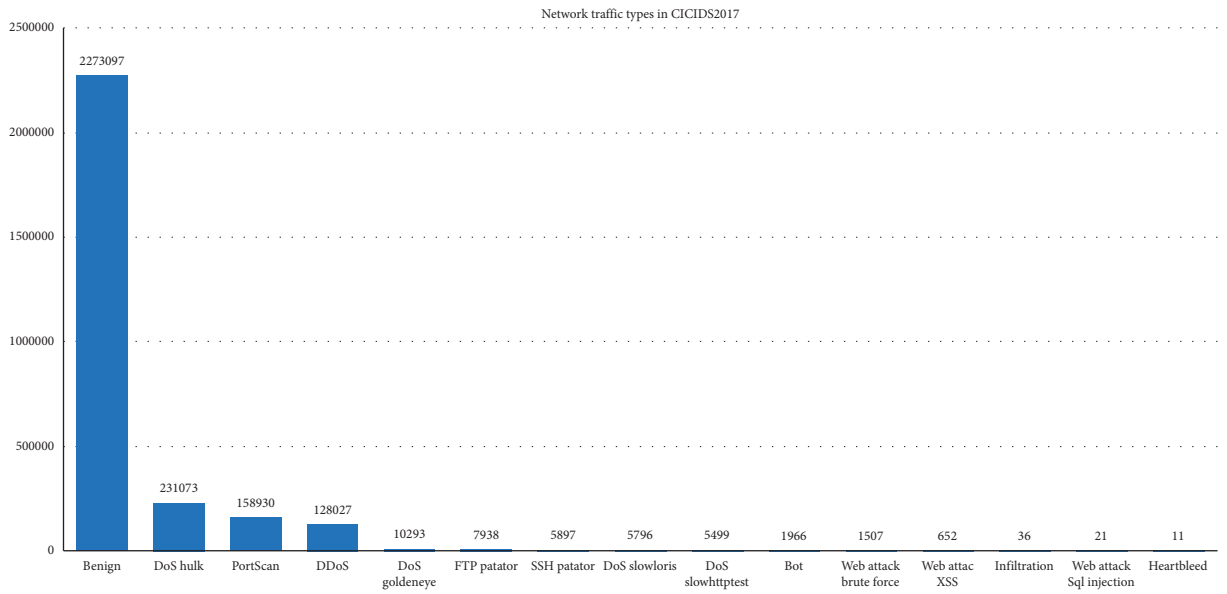


Figure 4: Network traffic types in CICIDS2017.

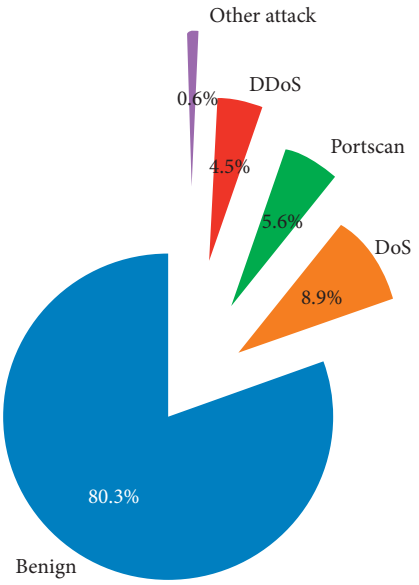


Figure 5: Complete CICIDS2017 dataset distribution.

time automatic monitoring of future network traffic anomaly detection function.

## 4. Experimental Results and Analysis

*4.1. Experimental Method of Packet Sampling.* In order to verify the difference of sampling strategy described in 3.1, Matlab and graph theory were used to build the model and construct the network topology and node sampling function. Four kinds of topology structure and four kinds of attack and defense strategies were used to carry out 16 groups of simulation, each group of simulation was repeated 10 times, and then the average value of each group of data was calculated. Under different combinations of attack strategies and topologies, SDN packet sampling strategy based on zero-sum game is compared with random sampling strategy. The experimental topology is shown in Table 1. The

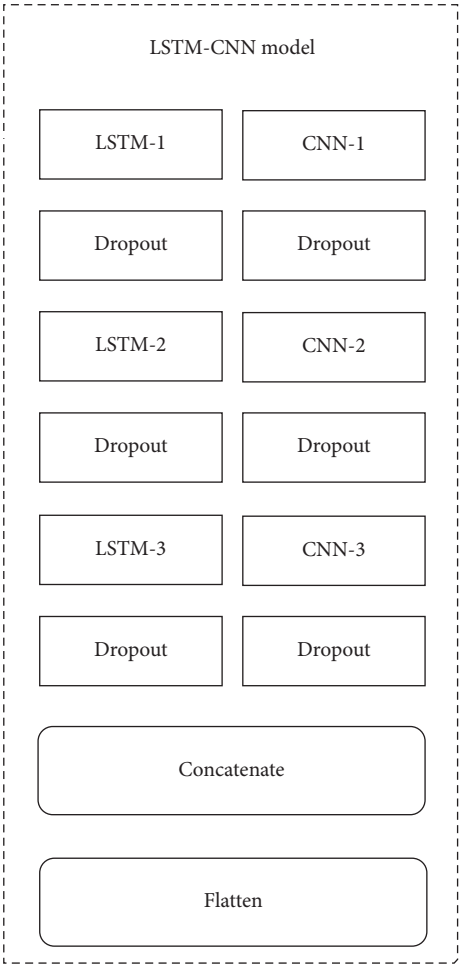


Figure 6: LSTM-CNN model.

detection success rate of defenders against attacks and the scores of attackers are shown in Table 2.

In the experiment, there is only one attack host and the attack data sent to the network every second is 20 per second. Each network device can receive 20 packets per second. The total number of sampling nodes per second for network devices in the entire topology is 20, and the attack score is reserved to decimal.

Experimental data show that, compared with attack strategy 1, attack strategy 2 can improve the defense success rate and reduce the attack score, which indicates that, in network attack, increasing the power of sending attack packets to the target host will make the target host easy to detect the attack and take active defense. Compared with defensive strategy 1, defensive strategy 2 can improve the detection success rate and reduce the attack score. The reason is that SDN packet sampling strategy based on zero-sum game tends to protect important nodes, so this strategy is effective.

*4.2. Datasets and Experimental Methods of Anomaly Detection.* In this section, the mentioned network traffic anomaly detection method is tested. All models of this method are designed and verified on the Google Colab platform, and the TPU accelerator provided by Google Colab is used. The framework of deep learning selects Keras based on TensorFlow 2.1 and CICIDS2017 [27] as the dataset for anomaly detection.

We use CICIDS2017 as the dataset for anomaly detection, published by the Canadian Institute for Cybersecurity. CICIDS2017 is a dataset for simulating real attacks and contains the necessary features for common network events. Among them, the traffic data are captured by packet and extracted by CICFlowMeter. Each data contains more than 80 dimensions of network traffic characteristics.

Before the experiment, we first conducted data statistics on CICIDS2017, and its traffic types is shown in Figure 3 and its traffic distribution is shown in Figure 4. It can be seen that there are 15 types of traffic, including normal traffic and 14 types of attack traffic.

Then, we carried out numerical normalization and traffic label coding on the dataset, and the numerical normalization was mapped by the MinMax method. In the process of traffic label coding, according to the traffic distribution characteristics of CICIDS2017, we can see that the normal traffic occupies more than 80%, and the attack traffic is mainly DOS, PortScan, and DDoS. Therefore, we divided 15 types of traffic into Benign, DOS, DDoS, PortScan, and other attacks, as shown in Figure 5; it make our experimental training and statistics more convenient.

We input the serialized preprocessed data into LSTM and CNN neural network, where LSTM predicts the temporal characteristics of the traffic sequence and CNN learns the spatial characteristics of the network traffic sequence. This experiment of deep learning framework using Keras LSTM and neural network (CNN) in the model structure as shown in Figure 6, including CNN and LSTM three-layer neural network, is adopted, and each layer neural network

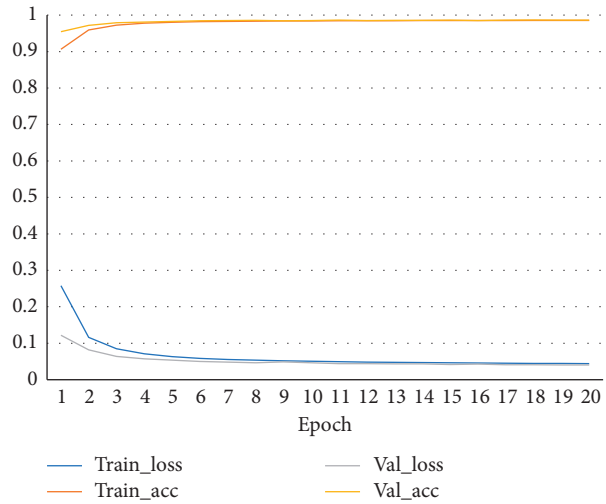


Figure 7: Loss-accuracy change rate.

Table 3: Classification report.

| Types | Precision | Recall | F1-score |
|---|---|---|---|
| BENIGN | 1.00 | 0.98 | 0.99 |
| DoS | 0.99 | 1.00 | 0.99 |
| PortScan | 0.95 | 1.00 | 0.98 |
| DDoS | 1.00 | 1.00 | 1.00 |
| Other attacks | 0.89 | 0.94 | 0.92 |
| Total | 0.966 | 0.984 | 0.976 |

using the dropout discard part features, to prevent overfitting, on the fourth floor, LSTM is combined with CNN through the flatten layer for dimension reduction and finally the output was sorted through the softmax layer.

After experimental tests, as the number of epochs increased, we obtained the variation trend of loss value and accuracy value in the traffic classification of the LSTM-CNN model. It can be seen from Figure 7 that when the epoch reached 7.5 times, the performance of this model tended to be stable. The loss value was 0.0441, and the accuracy value was 0.9853 when the epoch was 20 times.

After training the data, we tested the model and the accuracy reached 0.966, with a better recognition rate of network attacks. Table 3 shows the comparison of detection rates between normal traffic and attack traffic using CNN-LSTM. The evaluation criteria include precision, F1Score, and recall.

Through the experiment, DDOS can achieve 100% successful detection, and the average F1-score of normal traffic and other attack traffic can reach 97.6%, indicating that this method has excellent performance in the future network anomaly detection.

## 5. Conclusion

The global view and centralized control of the future network make the network traffic control in the big data environment convenient and effective, but most of the anomaly traffic detection often needs to be detected through a large

number of data samples and the number of abnormal traffic explosive growth, resulting in a decline in detection efficiency.

This paper proposes a sampling and classification prediction model of anomaly traffic of future networks based on game theory and deep learning. The defense performance of network is improved by protecting important nodes. The experimental platform has been built, and we also use public datasets to test our method. The results show that the sampling strategy of SDN packets based on zero-sum game and the method of deep learning analysis for the selected important nodes are effective. In the future, further research can be carried out on the game model, different types of deep learning methods, and super-parameter selection.

## Data Availability

The data used to support the findings of this study are available at https://www.unb.ca/cic/datasets/ids-2017.html.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research was supported by the National Natural Science Foundation of China (61972104), the Special Project for Research and Development in key areas of Guangdong Province (2019B010121001), and the Special Fund for Science and Technology Innovation Strategy of Guangdong Province (2020a0332).

## References

[1] D. B. Rawat and S. R. Reddy, "Software defined networking architecture, security and energy efficiency: a survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 325–346, 2017.

[2] J. Ren, A. Hussain, H. zhao et al., "Advances in brain inspired cognitive systems," in *International Conference on Brain Inspired Cognitive Systems*, vol. 11691 of Lecture Notes in Computer Science, Springer, Berlin, Germany, 2020.

[3] S. Jain, A. Kumar, S. Mandal et al., "B4," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 3–14, 2013.

[4] F. Cimorelli, F. D. Priscoli, A. Pietrabissa, L. R. Celsi, V. Suraci, and L. Zuccaro, "A distributed load balancing algorithm for the control plane in software defined networking," in *Proceedings of the 2016 24th Mediterranean Conference on Control and Automation (MED)*, pp. 1033–1040, Athens, Greece, June 2016.

[5] W. Zhang, X. Wang, S. Zhang, and M. Huang, "SDN data packet sampling strategy based on security game," *Journal of Zhengzhou University (Science Edition)*, vol. 50, no. 1, pp. 15–19, 2018.

[6] L. Lan, K. L. K. Ryan, R. Guangming, and X. Xu, "Malware propagation and prevention model for time-varying community networks within software defined networks," *Security and Communication Networks*, vol. 2017, Article ID 2910310, 8 pages, 2017.

[7] Y. Afek, S. A. Bremler-Barr, and L. SchiffLandau Feibish, "Sampling and large flow detection in SDN," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 5, pp. 345-346, 2015.

[8] F. Tang, L. Li, L. Barolli, and C. Tang, "An efficient sampling and classification approach for flow detection in SDN-based big data centers," in *Proceedings of the 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)*, Taipei, Taiwan, March 2017.

[9] J. Zhao, J. Sun, Y. Zhai, Y. Ding, C. Wu, and M. Hu, "A novel clustering-based sampling approach for minimum sample set in big data environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 32, no. 2, 2018.

[10] H. Grushka-Cohen, O. Biller, O. Sofer, L. Rokach, and B. Shapira, "Simulating user activity for assessing effect of sampling on DB activity monitoring anomaly detection," in *Policy-Based Autonomic Data Governance*, Springer, Berlin, Germany, 2019.

[11] Y. Yong-Qiang, S. Chao, and Z. Jian-Hui, "Research on impact of packet sampling on anomaly detection and its elimination method," *Computer Engineering*, vol. 39, no. 1, pp. 131–135, 2013.

[12] S. D. Bopardikar, A. Borri, J. P. Hespanha, M. Prandini, and M. D. Di Benedetto, "Randomized sampling for large zero-sum games," *Automatica*, vol. 49, no. 5, pp. 1184–1194, 2013.

[13] Z. Chiba, N. Abghour, K. Moussaid, A. El Omri, and M. Rida, "Intelligent approach to build a deep neural network based IDS for cloud environment using combination of machine learning algorithms," *Computers & Security*, vol. 86, pp. 291–317, 2019.

[14] X. Shao, M. Zhang, and J. Meng, "Data stream clustering and outlier detection algorithm based on shared nearest neighbor density," in *Proceedings of the 2018 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)*, Xiamen, China, January 2018.

[15] W. Huang and J. W. Stokes, "MtNet: a multi-task neural network for dynamic malware classification," *Detection of Intrusions and Malware, and Vulnerability Assessment*, Springer, Berlin , Germany, pp. 399–418, 2016.

[16] T. Bolukbasi, J. Wang, and O. Dekel, "Adaptive neural networks for efficient inference," in *Proceedings of the 34th International Conference on Machine Learning*, pp. 527–536, Sydney, Australia, August 2017.

[17] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, "Unsupervised real-time anomaly detection for streaming data," *Neurocomputing*, vol. 262, pp. 134–147, 2017.

[18] A. Wankhade and K. Chandrasekaran, "Distributed-intrusion detection system using combination of ant colony optimization (ACO) and support vector machine (SVM)," in *Proceedings of the 2016 International Conference on Micro-Electronics and Telecommunication Engineering, ICMETE 2016*, pp. 646–651, Ghaziabad, India, September 2016.

[19] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP)*, Madeira, Portugal, January 2018.

[20] P. Mishra, E. S. Pilli, V. Varadharajan, and U. Tupakula, "Intrusion detection techniques in cloud environment: a survey," *Journal of Network and Computer Applications*, vol. 77, pp. 18–47, 2017.

[21] L. Lan and L. Jun, "Some special issues of network security monitoring on big data environments," in *Proceedings of the 2013 IEEE 11th International Conference on*

*Dependable, Autonomic and Secure Computing*, Chengdu, China, December 2013.

[22] L. P. Dias, J. J. F. Cerqueira, K. D. R. Assis, and R. C. Almeida, "Using artificial neural network in intrusion detection systems to computer networks," in *Proceedings of the 2017 9th Computer Science and Electronic Engineering (CEEC)*, pp. 145–150, Colchester, UK, September 2017.

[23] Hu W., Tan Y., Black-box Attacks against RNN Based Malware Detection algorithms, 2017, https://arxiv.org/pdf/1705.08131.

[24] Grosse K., Papernot N., Manoharan P., Adversarial Perturbations against Deep Neural Networks for Malware classification, 2016, https://arxiv.org/pdf/1606.04435.

[25] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[26] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *Proceedings of the 2017 International Conference on Information Networking (ICOIN)*, pp. 712–717, Da Nang, Vietnam, January 2017.

[27] A. Boukhamla and J. C. Gaviro, "CICIDS2017 dataset: performance improvements and validation as a robust intrusion detection system testbed. International Journal of Information and Computer Security," vol. 9, 2018.