Anomaly Detection on Time-series Logs for Industrial Network

1th Lin Chen

Electric Power Research Institute, CSG
Guangzhou 510663, China
chenlin_temp@163.com

2th Xiaoyun Kuang

Electric Power Research Institute, CSG
Guangzhou 510663, China
kuangxiaoyun_temp@163.com

3th Aidong Xu

Electric Power Research Institute, CSG
Guangzhou 510663, China
xuaidong_temp@163.com

4th Siliang Suo

Key Laboratory of Guangdong electric power system network security enterprise
Guangzhou 510663, China
suosiliang_temp@163.com

5th Yiwei Yang

Key Laboratory of Guangdong electric power system network security enterprise
Guangzhou 510663, China
yangyiwei_temp@163.com

2020 3rd International Conference on Smart BlockChain (SmartBlock) | 978-1-6654-4073-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/SMARTBLOCK52591.2020.00022

***Abstract*—With the deep integration of industrialization and informatization, the network environment is becoming more and more complex, and security is facing a huge threat. Recently, the industrial control systems pose an open trend, so the strategy of preventing external attacks through "physical isolation" does not work anymore. The security threats in the traditional IT field gradually affect the security of industrial control networks. Recently, more and more researchers apply artificial intelligence algorithms and blockchain technology to industrial control network security. This paper aims to propose a new way of thinking, starting from two levels of physical topology and time series structure for a specific industrial control system, establish a graph data structure, and then use the graph neural network (GNN) algorithm to detect abnormal nodes. We evaluate our approach through comprehensive experiments and the results are promising.**

Keywords—Anomaly Detection, Time-series Log, Industrial Network.

1. INTRODUCTION

The Industrial Control System (ICS) is a kind of process management and control system with a complex topological structure. It includes not only various automatic control components but also a variety of process control components for real-time collection and response. For instance, the power dispatching automation system is a typical kind of Industrial control systems.

In recent years, Industrial control systems become the main target of attackers and intrusions such as Stuxnet, Blackenergy, etc. The early industrial control system is closed, which can isolate attacks from the outside. Therefore, in the early stage, "physical isolation" is an effective way to prevent network intrusions. However, with the integration of industrialization and informatization, the Internet-of-Things is also developing rapidly, and the industrial control system is evolving to open and changeable, thus the threat of the traditional IT field extends to the field of industrial control networks. The industrial control network exposed to risk also shows its vulnerability.

Faced with the increasingly severe network security situation, it is particularly important to detect the abnormal state and malicious behavior in the network environment. Therefore, anomaly detection is an essential task to build a safe and reliable computer system. Typically, system logs record system status and major events at key points to help developers debug performance issues and locate failures, and are a valuable resource for understanding system status [16][17][18]. In addition, such log data is generally available in almost all computer systems. The wealth of information and the ubiquity of logs make it possible to perform system monitoring and diagnostic tasks through log management analysis, such as identifying performance anomalies, analyzing usage statistics, diagnosing errors and crash causes, and ensuring application security [19][20].

The industry generally believes that as the cornerstone of trust in new infrastructure, blockchain helps to solve many problems faced in the development of the current industrial control network, such as realizing the trusted interconnection of machines, workshops, enterprises, and people, and improving industrial production efficiency, upgrade and optimization of data trustworthy exchange, service manufacturing upgrade, and industrial information security. More and more researchers apply artificial intelligence algorithms and blockchain technology to industrial control network security and the industrial Internet-of-Things.

Different from IT systems, there are various heterogeneous sensors, controllers, and other equipment in the industrial control systems. These components have complex interactions. The complex and coherent interaction makes each node of the industrial control network depend on each other and cooperate with each other, forming a structured network with strong data correlation between nodes. Traditional anomaly detection approaches on time-

978-1-6654-4073-8/20/$31.00 ©2020 IEEE
DOI 10.1109/SmartBlock52591.2020.00022

series data have not considered the correlation and interaction among the nodes of the industrial control network.

Many anomaly detection methods have been proposed to detect the abnormal state and behavior in the network. Based on the types of data involved and the techniques employed, there are two broad categories: supervised logging anomaly detection and unsupervised logging anomaly detection. Methods of supervised anomaly detection mainly include decision tree, logistic regression and support vector machine (SVM), etc., while methods of unsupervised anomaly detection mainly include principal component analysis (PCA), clustering, invariant mining, etc. The above supervised methods are all feature-based algorithms that need to be annotated in advance, and such methods rely on prior knowledge. It has good results when detecting known attacks, but it is difficult to identify unknown malicious behaviors. In contrast, the unsupervised method shows high accuracy in the face of unknown anomalies, but it also has a high false positive rate, and many normal events or logs are classified as abnormal events.

To address the above issues, we propose a GNN-based anomaly detection approach for detection anomalies in industrial networks. The main contribution of this paper is as follows:

- The industrial control network is abstracted into a graph model from both the physical topology structure and time series logs. In this model, each device is abstracted as a node in the graph.
- We propose a GNN-based anomaly detection approach which could identify anomaly nodes (such as broken sensors or manipulated controller) in the industrial network.
- We conduct comprehensive experiments using well-known benchmark datasets. The results verify the effectiveness of our system.

The remainder of this paper is structured as follows. We describe the background and related work in Section II. Then we give the design and implementation of our approach in Section III. Section IV demonstrates some experiments to evaluate the effectiveness and efficiency of our approach. We close our paper with the conclusion in Section V.

2. RELATED WORK

The security methods of ICS can be roughly divided into two types: one is to extract the abnormal or attack pattern from the abnormal behavior, and then identify the observed values which match the pattern. The other is to establish the model of normal network behavior from the perspective of normal network behavior to determine whether the characteristics of behavior are within the scope of conventional behavior, to detect abnormal behavior.

The normal behavior of industrial control networks is explained in research work [1]. In the specific attack mode, although some packets conform to the network protocol specification, they are contrary to the current operation situation of the industrial control network, so it is necessary to warn of this behavior. In this way, we can identify whether the behavior of the industrial control network is running within the scope of normal behavior model, and then detect the external network attack or the abnormal situation inside the system. In theory, this method can identify network abnormal behavior accurately and warn the of system abnormal situations in time. However, the difficulty of this method is to establish an accurate and available behavior model. It is necessary to analyze the behavior patterns of network attacks and system anomalies to find out the frequent behavior patterns of a normal network. This will involve a comprehensive and profound understanding of the specific system and a large amount of work. Moreover, in the actual industrial production environment, the anomaly detection system is often implemented by an embedded platform with low computing power, which is limited by the requirements of computer computing power and real-time performance of the industrial system. Under the trade-off, this method cannot achieve the optimal performance.

Literature [2] designed and implemented an intrusion detection system and event monitoring environment for power grid system using SCADA system. The specific method is to record the information of all devices in the system in the form of an XML graph, use the program to parse the file and generate snort IDS features, and then detect the abnormal network behavior. The advantage of this method is that it can automatically model network abnormal behavior according to the network structure, improve the efficiency and accuracy of network feature mining, and reduce the negligence and omission caused by manual network behavior feature mining. However, this method can only resolve the relatively simple network behavior patterns. For those network behavior patterns which are relatively complex in principle and process, they still need to be manually configured. It can be inferred that the ability of this method to detect unknown network attacks or system anomalies is relatively low.

The anomaly detection method in literature [3] is based on the periodic characteristics of the industrial control network traffic. In the normal operation of the industrial control network, there are control command transmission and real-time status reports of field equipment to the control system. Due to the characteristics of industrial systems, network traffic often bursts at a certain time, which is closely related to the normal operation of industrial equipment. Therefore, there is a relatively fixed cycle and interval change of network traffic state with the operation of industrial system. For systems with abnormal conditions, network attacks or device anomalies often cause changes in network traffic state, such as changes in the burst period of network traffic, changes in the number of network packets, and changes in network noise. This method can achieve a relatively strict anomaly detection system to improve the detection rate only from the point of view of network traffic data characteristics. However, the change of network condition may be affected by many factors. Manual intervention in the industrial control system for equipment operation and instruction release will also change the flow status in the control network, and result in a high false alarm rate.

The above research works have some common problems:

- All network abnormal modes or normal network behavior patterns are fixed, which are set by a human in advance, and have weak detection ability for unknown attacks.
- The detection index of anomaly detection in the model is too single and leads to high false alarm

rate. Therefore, it needs many human efforts to carry out fault events. Besides, the process of building the database of network behavior patterns is very complex.

In recent years, artificial intelligence algorithms are gradually extended to the field of anomaly detection in industrial control networks. Some machine learning and deep learning algorithms have been applied. The K-means network anomaly detection method proposed in reference [4] is a semi-supervised machine learning algorithm based on incomplete data. The use of machine learning algorithm can help to establish a normal network behavior profile and detect abnormal behavior. This method uses labeled data to train the model, considers the situation when the seed information of K-means is not complete, and considers the selection of K value, the influence of noise points and outliers on the model, so that the model can maintain a high detection rate. At the same time, due to the strong generalization ability of the machine learning algorithm, the model has a strong learning ability. Furthermore, it can be effectively detected in the case of unknown network attacks or system anomalies. In reference [5][6][24], convolution neural network is used to solve the problem of anomaly detection in the industrial control networks. This method will collect the network traffic.

As an emerging technology, blockchain shows its effectiveness in anomaly detection. Blockchains are immutable digital ledger systems implemented in a distributed fashion and usually without a central authority. At their most basic level, they enable a community of users to record transactions in a ledger that is public to that community, such that no transaction can be changed once published [25].

The blockchain system is composed of data layer, network layer, consensus layer, incentive layer, contract layer, and application layer. Among them, the data layer encapsulates the underlying data block, the related basic data and basic algorithms such as data encryption and time stamp. The network layer includes distributed networking mechanism, data transmission mechanism, and data verification mechanism, etc. The consensus layer mainly encapsulates various consensus algorithms of network nodes. The incentive layer integrates economic factors into the blockchain technology system, mainly including the issuing mechanism and distribution mechanism of economic incentives. The contract layer mainly encapsulates all kinds of scripts, algorithms, and smart contracts, which is the basis of the programmable characteristics of blockchain. The application layer encapsulates various application scenarios and cases of blockchain. In this model, the time-stamp based chain block structure, the consensus mechanism of distributed nodes, the economic incentive-based on consensus computing power, and the flexible and programmable smart contract are the most representative innovation points of blockchain technology.

Blockchain had been widely used in various applications. Qiu and Gai et al. had applied blockchain techniques to enhance the security and protect the privacy of Internet of Things [11] and smart grid [12]. Other applications with blockchain had also been reported, such as digital evidence systems [13] and finance systems [14]. Qiu et al. also illustrated how to use dynamically scalable blockchain to enhance the security of communication systems [15].

3. Design and Implementation

In this section, we will first introduce the testbed used in this paper, then present the design of our approach.

3.1 An Industrial Control Testbed

SWaT [10][21] is a simulated industrial control environment Designed by the Singapore University of Technology and Design which is used to support the research and development of cyber-physical systems (CPS). The testbed is simulating a water treatment system. SWaT testbed consists of six Processes. Process 1 (P1) is responsible for water intake, storing the raw water into the water tank for further treatment. Process 2 (P2) is responsible for the pretreatment work, measuring the content of NaCl, HCl, and other substances through conductivity, acidity, and alkalinity. Process 3 (P3) is responsible for ultrafiltration, i.e Using a fine filtration membrane to remove impurities. Process 4 (P4) is the dechlorination stage, using ultraviolet lamp dechlorination. Process 5 (P5) is the reverse osmosis stage,

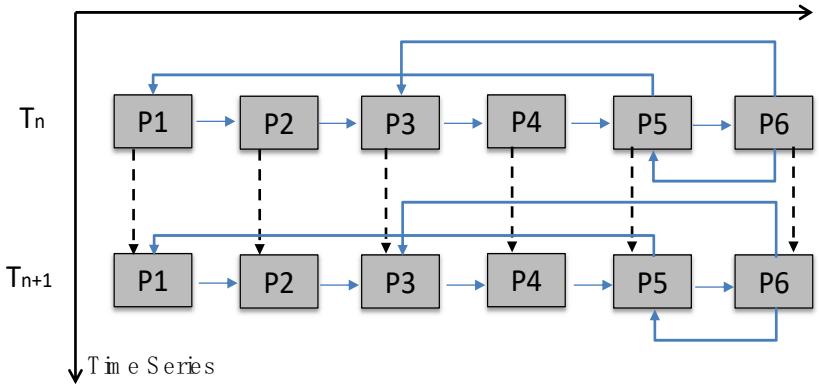

Figure. 1. The graph model of Swat

mainly responsible for removing inorganic impurities. Process 6 (P6) plays an intermediate storage role, if the water quality meets the discharge requirements, the water will be discharged, otherwise, the water will be sent back to the P3 ultrafiltration stage to start the cycle again. In the process of the test-bed running, there are sensors and brakes in each process stage. The sensors send the field equipment status data to the upper management or decision-making system, and then the upper system will control the command to reach

the brake after analysis. The whole process is completed through industrial control network. When the equipment sends and receives signals and instructions through the industrial control network, it can capture the communication process and equipment status information in the network.

3.2 Graph Model

From the aspect of spatial analysis, the SWaT testbed is presented in all stages of the sensors and brake to cooperate with each other, phase there is mutual restriction between interactional relationship. When the equipment in the network is in normal operation, the data in the related equipment has a linkage change relationship. For example, the periodic change of the water flow sensor data in the pipeline reflects the water flow change, which affects the periodic change of the water level sensor data in the water tank. When one device in a network is attacked, the effect can also be reflected in other sensors or brakes when it is not working properly. Therefore, it is easy to establish a graph model that conforms to the graph neural network input according to the water flow path. The independent stages of each water treatment process are taken as individual nodes and adjacent stages are connected by undirected edges. In addition, it should be noted that the water treatment process of the testbed is not a one-way chain. Since water may be sent back to the earlier stage of the process for re-treatment under certain conditions in some stages, edges between (P1, P5), (P3, P6) and (P5, P6) should also be added, as shown in Figure 1.

From the perspective of time, the data characteristics of the SWaT data set are related not only to the spatial structure of the testbed, but also to the state of the testbed at a certain period of time. During the normal operation of the test platform, the state of each experimental equipment is in a relatively stable periodic change, and the state of the previous stage regularly affects the current state of the equipment to a certain extent. When attacked, the device status data cycle is often destroyed, which is also an important feature of industrial control network anomaly detection. Therefore, it is necessary to extend the model and create state nodes at different times, that is, create different nodes for the same processing process at different time points. On the basis of ensuring that an adjacency relationship is maintained between different nodes at the same time, an adjacency relationship is also established between nodes at adjacent moments of the same process, as shown in Figure 1.

3.3 GNN-based Anomaly Detection

Some researchers have applied machine learning or deep learning algorithms to anomaly detection for SWaT. Inoue J et al. [22] proposed and evaluated the application of unsupervised machine learning to anomaly detection for SWaT. They compared two methods: Deep Neural Networks (DNN) adapted to time series data generated by SWaT, and one-class Support Vector Machies (SVM). For both methods, they first train detectors using a log generated by SWaT operating under normal conditions. Then, they evaluate the performance of both methods using a log generated by SWaT operating under 36 different attack scenarios. They find that DNN generates fewer false positives than one-class SVM while SVM detects slightly more anomalies. They summarized and compared the advantages and disadvantages of the two methods.

Different from other common machine learning or deep learning algorithms, graph neural network [23] is a kind of neural network that performs tensor operation directly on the graph structure. During the training process of the graph neural network model, different nodes will exchange data along the edges connected between nodes. After exchanging data, each node carries out independent calculation according to the data exchanged by nearby nodes and updates the node state until the node state reaches a certain stable value, that is, the node state updates less and less until it reaches a certain critical value. After the state of the node has stabilized, each node calculates the output of the node based on the current state data.

In GNN [8][9] model, the update function of node state and output function of node state are the key points of constructing a graph neural network model. The state update function of the node takes the current state vector x_n of the node, the current eigenvector l_n of the node, the eigenvector $L_{NE\,[N]}$ of the neighbor node, and the eigenvector $L_{CO\,[N]}$ of the adjacent side as factors to calculate the state vector x_n of the node in the next round. In the data structure modeling of this graph, there is no feature between nodes, so the feature vector $L_{CO\,[N]}$ of adjacent edges will not be included in the calculation of the node update function. That is:

$$x_n = f_\omega\left(l_n, x_{ne[n]}, l_{ne[n]}\right), n \in N$$

Graph neural network does not do any constraint g_w to the output function, so you can directly use a simple feed-forward neural network for the output. The input dimension is the dimension of node state vector x_n and node feature vector l_n, and the output dimension is the number of node label types. Since the goal of this task is to judge the state of the node, and the state of the node is divided into two classes: abnormal and normal, therefore the output dimension is 2.

Table 1. Precision in different parameters

No.	Learning Rate	Vector Dimension	Precision
1	0.01	4	84.3%
2	0.01	8	89.4%
3	0.01	16	87.7%
4	0.1	8	88.9%
5	0.1	16	86.6%

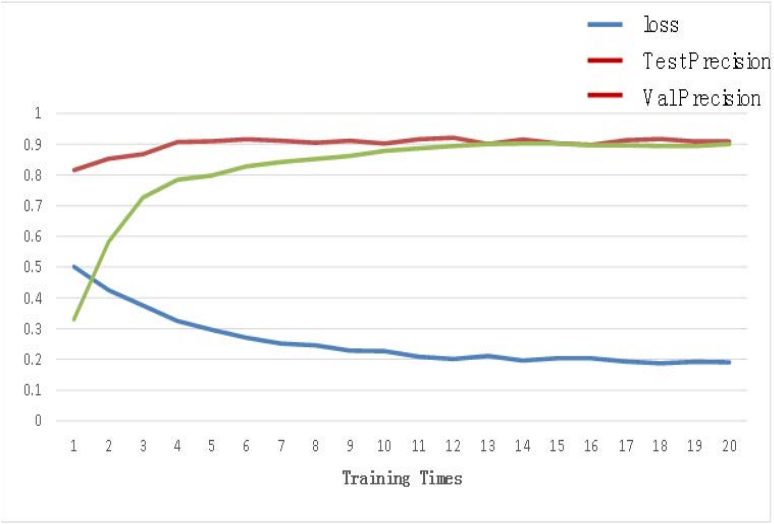

Figure. 2. Train and Testing Results

The state transition function and output function are shared by all nodes. In the process of the model training of each loop, each node computes according to the above formula and iteratively updates the state vector of each computing node. In order to achieve the optimized state vector, the purpose of stabilizing output function and then use the node according to the node state vector and feature vector to forecast the node label. In the process of loop-back propagation, the Adam optimizer is used and the cross-entropy is used as the loss function.

4. EXPERIMENTS

In this section, we will introduce the experimental results of our approach.

4.1 Dataset

The researcher of Swat run the testbed for 11 days and collect all the test data. Among the 11 days, the first 7 days run normally. And the following 4 days, operators launched 36 attacks, and collect the normal run and attack in the process of traffic data in the industrial control network and the physical equipment status data (sensor, brake, etc.), during a sampling per second, collected more than 100 gigabytes of data. The data used in this paper is only part of the data of the physical device network, including 496800 pieces of normal operation data and 449919 pieces of abnormal operation data. Each piece of data contains timestamps and data on the state of the sensors and brakes in each process.

4.2 Experimental Results

The normal data and abnormal data are divided into training sets and test sets respectively with a ratio of 5:1. The training set is then divided in a 1:1 ratio, one of which is used as a verification set. When the number of cycles of the set node transition function is 3, the dimension of the state eigenvector is 8, and the learning rate is 0.01. After 20 training iterations, it was found that the loss of the training set gradually stabilized at about 0.2 after the 12th iteration, the accuracy of the test set reached 92.1%, and the accuracy of the verification set reached 89.4%. In subsequent training, the accuracy of the test set and the verification set fluctuated around 90%, achieving good results.

After multiple parameter adjustments, the accuracy training results of the verification set are shown in Table 1. It can be seen that the model performs well when the state vector dimension is set to 8. When the dimension of the state vector is set to 4, it may be because the model cannot get sufficient training due to too few states, and the model cannot fully explore the state information of the graph data structure. However, when the state vector dimension is set high, overfitting may occur, so the performance of the model on the validation data set will be worse. As for the setting of the learning rate, in the larger case, it affects the convergence speed of the model. In most cases, the model converges within 50 training cycles.

5. CONCLUSION

In this paper, we provide a new thinking, starting from two levels of physical topology and time series structure for a specific industrial control system. We establish a graph data model to simulate SWaT, an industrial control system testbed, and then use the graph neural network (GNN) algorithm to detect abnormal states. We evaluate our approach through comprehensive experiments and the results are promising.

6. ACKNOWLEDGEMENT

The authors gratefully acknowledge the anonymous reviewers for their helpful suggestions.

References:

[1] S. Zanero. Behavioral intrusion detection. Computer and Information Sciences-ISCIS 2004.Springer. 2004:657-666

[2] P. Oman, M. Phillips. Intrusion Detection and Event Monitoring in SCADA Networks. international conference on critical infrastructure protection, 2007: 161-173

[3] R. Barbosa, R. Sadre, and A. Pras. Towards periodicity based anomaly detection in SCADA networks. emerging technologies and factory automation, 2012: 1-4

[4] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai. Kitsune: an ensemble of autoencoders for online network intrusion detection. arXiv preprint arXiv:1802.09089, 2018

[5] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y Sheng. Malware traffic classification using convolutional neural network for representation

learning. 2017 International Conference on Information Networking (ICOIN). IEEE, 2017: 712-71

[6] D. Urbina, J. Giraldo, A. Cardenas, N. Tippenhauer, and H. Sandberg. Limiting the Impact of Stealthy Attacks on Industrial Control Systems. computer and communications security, 2016: 1092-1105

[7] J. Vavra, M. Hromada. Anomaly Detection System Based on Classifier Fusion in ICS Environment. soft computing, 2017

[8] F. Scarselli, M. Gori, A. Tsoi, Hagenbuchner M, &Monfardini G. The Graph Neural Network Model. IEEE Transactions on Neural Networks, 2009, 20(1): 61-80

[9] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, C. Li , and M. Sun. Graph Neural Networks: A Review of Methods and Applications. arXiv: Learning, 2018

[10] J. Goh, S. Adepu, K. Junejo, and A. Mathur. A Dataset to Support Research in the Design of Secure Water Treatment Systems. critical information infrastructures security, 2016: 88-99

[11] K. Gai, Y. Wu, L. Zhu, Z. Zhang, and M. Qiu, Differential Privacy-Based Blockchain for Industrial Internet-of-Things. IEEE Trans. Ind. Informatics 16(6): 4156-4165 (2020)

[12] K. Gai, Y. Wu, L. Zhu, M. Qiu, and M. Shen, Privacy-preserving Energy Trading Using Consortium Blockchain in Smart Grid, IEEE Transactions on Industrial Informatics, Vol. 15, No. 6, pp. 3548-3558, 2019

[13] Z. Tian, M. Li, M. Qiu, Y. Sun, and S. Su, "Block-DEF: A Secure Digital Evidence Framework using Blockchain," Information Science, Elsevier, Vol. 491, No.1, pp: 151-165, 2019

[14] W. Pan, M. Qiu, Application of Blockchain in Asset-Backed Securitization, 2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), Baltimore, MD, USA, 2020, pp. 71-76

[15] H. Qiu, M. Qiu, G. Memmi, Z. Ming, and M. Liu, A Dynamic Scalable Blockchain Based Communication Architecture for IoT, International Conference on Smart Block Chain (SmartBlock), LNCS 11373, pp. 159–166, Tokyo, Japan, Dec. 2018

[16] Chow M, Meisner D, Flinn J N, Peek D N, Wenisch T F. The mystery machine: End-to-end performance analysis of large-scale internet services. 11th {USENIX} Symposium on Operating Systems Design and Implementation({OSDI} 14), 2014: 217-231

[17] Nagaraj K, Killian C, Neville J. Structured comparative analysis of systems logs to diagnose performance problems. Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation. USENIX Association, 2012: 26-26

[18] Lee G, Lin J, Liu C, Lorek A and Ryaboy D. The unified logging infrastructure for data analytics at Twitter. Proceedings of the VLDB Endowment, 2012, 5(12): 1771-1780

[19] Yuan D, Mai H, Xiong W, Lin T and Pasupathy S. SherLog: error diagnosis by connecting clues from run-time logs. ACM SIGARCH computer architecture news. ACM, 2010, 38(1): 143-154

[20] Xu X, Zhu L, Weber I, Bass L and Sun D. POD-Diagnosis: Error diagnosis of sporadic operations on cloud applications. 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks. IEEE, 2014: 252-263

[21] Adepu S, Mathur A. An Investigation into the Response of a Water Treatment System to Cyber Attacks. The 17th IEEE International Symposium on High Assurance Systems Engineering (HASE) 2016. IEEE, 2016

[22] Inoue J, Yamagata Y, Chen Y, Poskitt, Christopher M and Sun J. Anomaly Detection for a Water Treatment System Using Unsupervised Machine Learning. 2017 IEEE International Conference on Data Mining Workshops (ICDMW). IEEE Computer Society, 2017

[23] Jia W . Overview of Graph Neural Network. Modern Computer. 2019

[24] Canizo M, Triguero I, Conde A and Onieva E. Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study. Neurocomputing, 2019, 363:246-260

[25] Dylan Y, Peter M, Nik R and Karen S. Blockchain Technology Overview. NIST Interagency/Internal Report (NISTIR)-8202, 2019