Scholar Publishing Group
Machine Learning Theory and Practice
https://doi.org/10.38007/ML.2022.030307
ISSN 2790-0983 Vol. 3, Issue 3: 52-59

Cloud Monitoring Data Analysis and Simulation Based on Machine Learning

Rui Li*

Philippine Christian University, Philippine

WYLIRUI123@163.com

**corresponding author*

Keywords: Machine Learning, Cloud Monitoring, Massive Data, Analysis and Simulation

Abstract: With the rapid development of Internet technology, cloud computing has gradually been widely used in people's lives. However, due to the huge and complex data volume, limited storage space and maintenance difficulties, there are still some problems in cloud monitoring data analysis. This paper intends to use the combination of machine learning to study the analysis and experimental simulation of cloud monitoring data, in order to improve the data processing ability of the monitoring system. This paper mainly uses the experimental analysis method and information entropy to study the cloud monitoring system and its data processing and analysis capabilities. The experimental data show that the error of computer tools is less than 6%, which shows that the method is feasible and worthy of application.

1. Introduction

With the development of cloud monitoring platform, more and more data mining systems based on the analysis function and management under the network layer are introduced into its application field. At present, cloud monitoring system has two main development directions. One is an extensible platform based on the Web server, and the other is a network server (CSS) and desktop oriented development architecture. With the development and application of cloud monitoring system, after solving a series of problems such as data collection, transmission and processing, more and more people begin to research and use network architecture technology to achieve data sharing.

There are many research theories on machine learning in cloud monitoring data analysis. For example, some scholars designed a cloud monitoring data analysis platform integrating machine learning to further improve the accuracy and efficiency of data analysis on the cloud monitoring platform [1-2]. Some scholars research and design big data analysis platform based on cloud computing, hoping to provide reference for the construction of big data analysis platform [3-4]. Some scholars believe that machine learning platform can flexibly use existing computing resources to improve data processing efficiency [5-6]. Therefore, it is of great significance to use machine learning to simulate and analyze cloud monitoring data.

This paper first studies the data fusion analysis technology, and expounds the related concepts of data fusion and information entropy. Secondly, it analyzes the cloud monitoring platform integrating machine learning, and describes the relevant steps of cloud monitoring design. Then the theory and practice of the network monitoring management system are described. Finally, the relevant conclusions are drawn through simulation experiments and data analysis.

2. Cloud Monitoring Data Analysis Integrating Machine Learning

2.1. Data Fusion Analysis Technology

Data fusion analysis technology makes the analysis results of massive information more reliable and accurate through computers. Data fusion technology increases the reliability of target and reduces the uncertainty of information [7-8].

Data fusion analysis technology is mainly used to analyze and process the information obtained by distributed systems or multi-sensor. Decision level fusion belongs to the top level of three-layer fusion, which plays a vital role in the decision-making level. It coordinates and synthesizes the analysis results of a single sensor according to certain rules, and gives a comprehensive and perfect global optimal decision. The focus of feature level fusion analysis is feature extraction. The main step is to combine the obtained feature vectors and transmit them to the processing center for comprehensive analysis. The feature level belongs to the second level of three levels. Data level fusion is to process, synthesize and analyze the non preprocessed multi-source heterogeneous information collected by multi-sensor. It is the lower level of the three levels. Its biggest feature is that it retains the original features of the data, can obtain more accurate information, and has higher processing accuracy [9-10].

Information entropy makes a huge share in the measurement of uncertainty, which is the basis of modern information theory. Assuming that M is a continuous random variable, Q (m) indicates the probability when M is taken by m_i, then the uncertainty of M can be expressed by entropy G (M):

$$G(M) = -\int \mathrm{q}m_i \ln q(m_i) cm \tag{1}$$

The information entropy formula corresponding to discrete random variable X is:

$$G(M) = -\sum_{\mathrm{m} \in d} qm_i \ln q(m_i) \tag{2}$$

According to the above formula, the probability distribution of variable M determines the corresponding entropy value of M. The stronger the randomness of the probability distribution of variable M, the greater the corresponding entropy value, the more information needed, and the lower the orderliness.

2.2. Cloud Monitoring Platform Integrating Machine Learning

Because the division of each function module of the cloud monitoring system is relatively vague, each sub module is quantized in the design process of this paper. It includes data collection, data analysis and management. Collect the analog signal output by the sensor and transmit it to the external network or send the information to the control node through wired to the PC. After obtaining the digital sequence code, the desired result is parsed through a series of operations such as preset placement, and stored in the server for subsequent use or retrieval of other service resources [11-12].

The data collection function is provided by cloud monitoring, cloud computing and other servers. The data collection function refers to integrating various sensors, storage servers and other related devices in the cloud monitoring system on a platform to achieve on-site information and real-time monitoring. By building this hardware, users can obtain the required original parameters more conveniently and quickly. At the same time, it can reduce some unnecessary costs. The sensor in the sensor module detects the existence of the monitored object and sends the information to the host through the wireless network. When the camera takes a picture of the target, it is necessary to pay attention to whether there is overlap and intersection between lenses. If the image is stratified, refocus and process again. If the same color area or multiple video clips are collected, they will be displayed in blue to ensure the normal operation of the monitoring system [13-14].

Based on the design of cloud monitoring system, it is necessary to divide its functions according to different application environments, user needs and other factors. It includes data management module, sensor acquisition module and network transmission processing module. The sensing information is divided into three categories: perceptual feature data and learning feature pattern recognition information. In this paper, we use one of the more mature classifier algorithms in machine learning. ROSM is used to realize fusion operation. This method can well solve the problem that the traditional single machine model cannot accurately distinguish multiple individuals at the same time [15-16].

2.3. Network Monitoring Management System

The system as a whole is a network monitoring system, which monitors the running state of the target network and analyzes the behavior of the target network. Understand the network monitoring system as a whole. The network environment of the network monitoring system is a LAN composed of four two-layer switches interconnected by two. Each switch of the network has about 50 ports, so the network has about 200 terminal devices at most. The four switches in the network are divided into front-end network and back-end network according to their positions. The front-end network and back-end network each have two switches, and the two switches at the same end form the active standby relationship. The HSRP protocol is used to ensure the active/standby working relationship in a normal environment and complete the active/standby switching when necessary [17-18].

The network monitoring system is divided into two parts in terms of deployment: network monitoring configuration item and network packet capturing configuration item. The network monitoring configuration item realizes the functions of regular polling detection of network status, obtaining and analyzing network status data, visual display of network topology status, alarm of network fault status, storage of status and events, generation of query data chart, etc. The network monitoring module is deployed on the monitoring management node, and the network packet capture module is deployed on the packet capture nodes at the front and rear ends and controlled by the monitoring manager. The network monitoring module is responsible for obtaining the status data and other information of the switch, forming events and reporting them, and finally displaying them on the interface, as well as monitoring netflow, syslog and other information. The network packet capturing module uses NDIS middle layer driver programming to obtain the switch image port data. After real-time analysis, the data is stored in the local hard disk for the monitoring node's post analysis. The monitoring module remotely controls the packet capture module by using the internal protocol, and obtains the real-time protocol analysis and statistics results of the packet capture module through the internal protocol for display on the monitoring interface.

First of all, in the process of data acquisition, it is necessary to preprocess the analog signal output by the sensor. Noise interference signals may be generated due to different measurement environments and test objects. The second is to ensure that the sampling cost is large enough to get accurate results. The third is to find a suitable model from the computing model to achieve accurate estimation and high real-time algorithm efficiency. In cloud monitoring, the first step is to determine the location of data collection and classify and integrate the collected information. Then, basic parameters such as sensors and nodes required for cloud monitoring are obtained through different types of data sources. Due to the high complexity of the current network environment and the large demand for network bandwidth, these key points can not be effectively used. On the other hand, there is a problem of delay in the transmission of radio frequency devices. In cloud monitoring, data processing is the core part. A large amount of information collected is directly connected with it to form a set of complete, stable and accurate data. This structure can effectively improve the running speed of the algorithm and reduce the complexity of the algorithm. At the same time, due to its good portability and the ability to change according to different environments, the application scope has been expanded. In cloud monitoring data processing, it is first necessary to display the collected real-time and abnormal working status. This requires the integration of all raw data information obtained. There may be noise interference, temperature and other external factors in the sensing environment of the sensor, which may lead to certain effects inside the sensor.

3. Design and Implementation of Monitoring System

3.1. System Composition

The monitoring system is mainly composed of three parts: Client, Server, and SuperServer. The Client is responsible for the monitoring of a single node, the Server is responsible for the data integration in a data center, and the SuperServer is the data integration of the entire cloud and provides remote access functions. See Figure 1 for details:

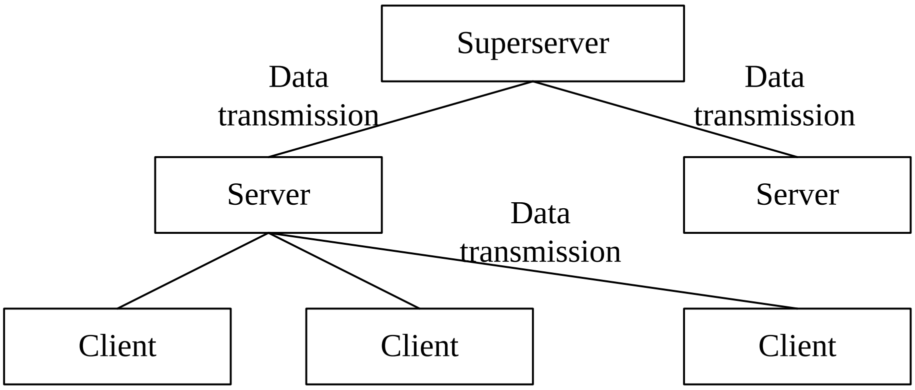

Figure 1. Overall structure of the monitoring system

The design requirement of the monitoring client is to obtain the information of a single node and send it to the server. For the server, although it is configured on an independent server, it still needs to know some information about the SuperServer. Similarly, it is also configured through xml information. The SuperServer side is responsible for monitoring the computing resources of the entire cloud, receiving client monitoring information from the server, and providing remote monitoring methods, including administrators' remote viewing of computing resource status, user request queuing, and server power consumption.

3.2. System Development Environment

The cloud computing platform monitoring system is based on the openstack platform for monitoring. Therefore, the monitoring system is deployed on the openstack cluster and uses the ceilometer monitoring framework for secondary development. The focus and difficulty of the work are the compilation of ceilometer monitoring plug-ins, the change of monitoring data processing process, and the formulation and formatting of analysis rules in the analysis engine. The implementation of this scheme involves the following development environment: data transmission

Openstack: mitaka

Ceilometer: Openstack monitoring framework, which cannot be inherited and deployed during openstack deployment, needs to be manually deployed in the cloud environment.

Esper: CEP (complex event process) analysis and processing engine, written in java language, analyzes data streams in real time and stores analysis rules.

JDK1.6: Java language software development kit.

Pycharm: openstack is written in python language.

3.3. Cloud Monitoring System Function Test

The function test includes the administrator's logical operations on the data center: adding, deleting data centers, clusters, servers, and viewing the hardware information and utilization information of specified servers. Tests show that these operations can be completed well. It is also necessary to test the real-time transmission of monitoring information.

4. Analysis of Experimental Test Results

4.1. Power and CPU Utilization

Due to the shortage of server resources, the test in this paper is based on an ordinary PC. The configuration is CPU: Intel Pentium E5400 3.6Ghz, memory: 4G. Java programs are used to control the CPU utilization. The statistical results are shown in Table 1:

Table 1. The relationship between server power consumption and CPU utilization

CPU Utilization	Power	CPU Utilization	Power
0	68.4	60	85.4
20	73.9	80	91.7
40	80	100	97

Use matlab to draw points and fit the data of the table. As shown in Figure 2, we can see that when CPU utilization changes regularly, its power also changes. According to Figure 2, the changes are positively correlated. In addition, the accuracy of the system tools has also improved with the increase of utilization.

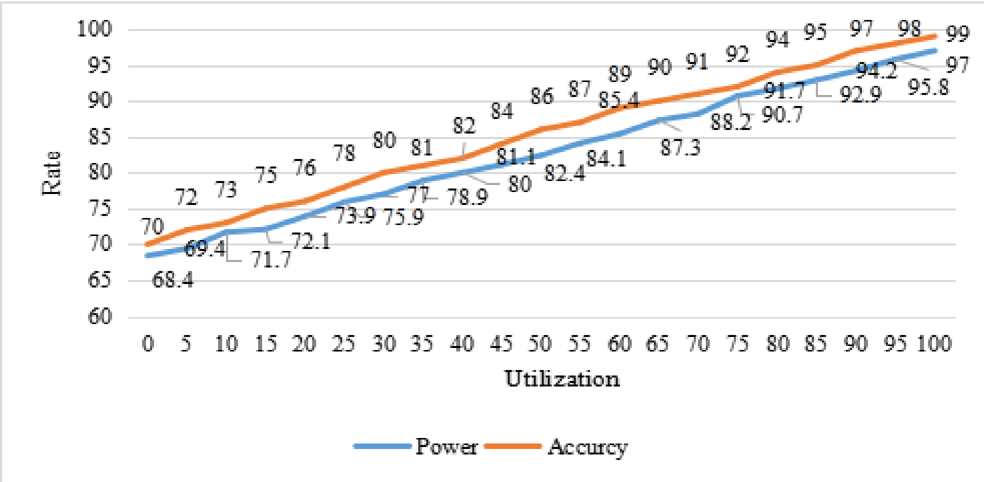

Figure 2. The relationship between server power consumption and CPU utilization

4.2. Calculation and Analysis of Power Consumption Data

Use the monitoring system to obtain the utilization rate of the test computer for 1 hour, and then calculate the theoretical value of power consumption according to the formula. At the same time, the computer power consumption of this hour is obtained through the power monitor. Finally, the theoretical value and the actual value are compared to verify the correctness of the monitoring system's power consumption acquisition method. The statistical results are shown in Table 2:

Table 2. Power consumption data calculation and analysis

	Calculated value	Actual test value	Error
1	0.081	0.08	2.4%
2	0.073	0..07	5.5%
3	0.085	0.09	4.3%
4	0.084	0.08	5.4%
5	0.081	0.08	2.3%

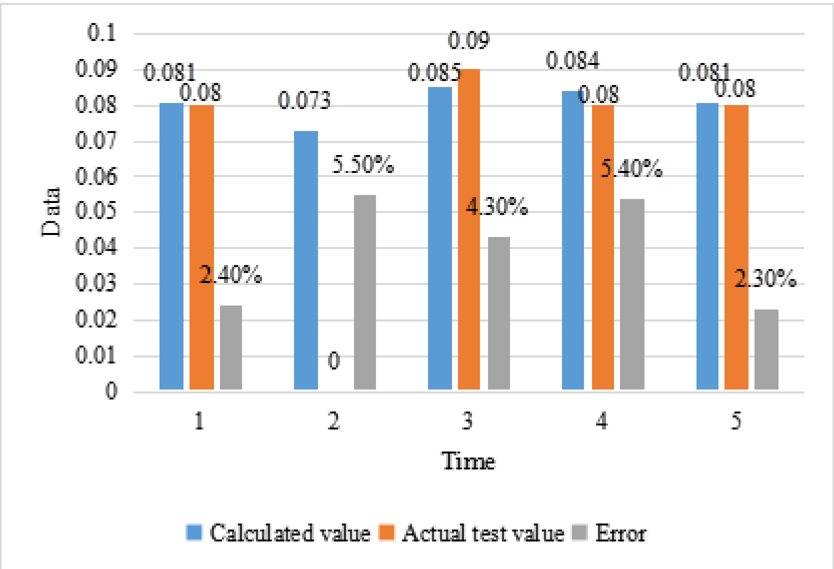

Figure 3. Power consumption data calculation and analysis

As shown in Figure 3, we can find that the error between the current power consumption calculation result and the actual detection value is small. However, it is feasible and stressful for cloud data center to use this method to monitor system power consumption. There are various servers in the cloud data center, and P (U) cannot be applied in a fixed way. For different servers, it is necessary to calculate their utilization and power consumption.

5. Conclusion

In this paper, we mainly study the cloud monitoring data mining algorithm based on machine learning. Through the analysis and processing of the collected sensor information, the data with high recognition accuracy that is most suitable for use in different environments can be obtained. Machine learning is a kind of information processing, data analysis and decision-making work by imitating human intelligence. Compared with the traditional algorithm, the system has the advantages of good real-time performance and strong adaptability. However, there are still some problems in the actual operation, which need further improvement and perfection to better serve our daily production and life.

Funding

This article is not supported by any foundation.

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] Ranya M. M. Salem, M. Sabry Saraya, Amr M. T. Ali-Eldin: An Industrial Cloud-Based IoT System for Real-Time Monitoring and Controlling of Wastewater. IEEE Access 10: 6528-6540 (2022). https://doi.org/10.1109/ACCESS.2022.3141977

[2] Madanjit Singh, Sarveshwar Bharti, Harjit Kaur, Vaibhav Arora, Munish Saini, Manevpreet Kaur, Jaswinder Singh: A Facial and Vocal Expression Based Comprehensive Framework for Real-Time Student Stress Monitoring in an IoT-Fog-Cloud Environment. IEEE Access 10: 63177-63188 (2022). https://doi.org/10.1109/ACCESS.2022.3183077

[3] Kashif Mehboob Khan, Junaid Arshad, Waheed Iqbal, Sidrah Abdullah, Hassan Zaib: Blockchain-Enabled Real-Time SLA Monitoring for Cloud-Hosted Services. Clust. Comput. 25(1): 537-559 (2022). https://doi.org/10.1007/s10586-021-03416-y

[4] Munish Bhatia, Sapna Kumari: A Novel IoT-Fog-Cloud-based Healthcare System for Monitoring and Preventing Encephalitis. Cogn. Comput. 14(5): 1609-1626 (2022). https://doi.org/10.1007/s12559-021-09856-3

[5] Berken Utku Demirel, Islam Abdelsalam Bayoumy, Mohammad Abdullah Al Faruque: Energy-Efficient Real-Time Heart Monitoring on Edge-Fog-Cloud Internet of Medical Things. IEEE Internet Things J. 9(14): 12472-12481 (2022). https://doi.org/10.1109/JIOT.2021.3138516

[6] Ali Reza Shamshiri, M. B. Ghaznavi-Ghoushchi, Ali Reza Kariman: ML-Based Aging Monitoring and Lifetime Prediction of IoT Devices with Cost-Effective Embedded Tags for Edge and Cloud Operability. IEEE Internet Things J. 9(10): 7433-7445 (2022). https://doi.org/10.1109/JIOT.2021.3116065

[7] Dennis A. Martillano, Marco C. Iligan, Algerica Raeven R. Ramos, Allan Daraman Jr. March Fernan H. Abadines: Wearable Tool for Breathing Pattern Recognition and Exacerbation Monitoring for COPD Patients via a Device-to-Cloud Communication Model. J. Commun. 17(6): 423-433 (2022). https://doi.org/10.12720/jcm.17.6.423-433

[8] Maad M. Khalill, Anatoly D. Khomonenko, M. D. Matushko: Measuring the Effect of Monitoring on a Cloud Computing System by Estimating the Delay Time of Requests. J. King Saud Univ. Comput. Inf. Sci. 34(7): 3968-3972 (2022). https://doi.org/10.1016/j.jksuci.2021.02.001

[9] Jerico Moeyersons, Sarah Kerkhove, Tim Wauters, Filip De Turck, Bruno Volckaert: Towards Cloud-Based Unobtrusive Monitoring in Remote Multi-Vendor Environments. Softw. Pract. Exp. 52(2): 427-442 (2022). https://doi.org/10.1002/spe.3029

[10] David Hästbacka, Jari Halme, Laurentiu Barna, Henrikki Hoikka, Henri Pettinen, Martin Larrañaga, Mikael Björkbom, Heikki Mesiä, Antti Jaatinen, Marko Elo: Dynamic Edge and Cloud Service Integration for Industrial IoT and Production Monitoring Applications of Industrial Cyber-Physical Systems. IEEE Trans. Ind. Informatics 18(1): 498-508 (2022). https://doi.org/10.1109/TII.2021.3071509

[11] Ankit Agarwal: Cloud Internet of Things Based Machine Monitoring Analysis of Energy Parameters Using Novel Techniques. Wirel. Pers. Commun. 124(2): 1789-1814 (2022). https://doi.org/10.1007/s11277-021-09431-x

[12] Jacob John, Mariam Sunil Varkey, Riya Sanjay Podder, Nilavrah Sensarma, Munuswamy Selvi, S. V. N. Santhosh Kumar, Arputharaj Kannan: Smart Prediction and Monitoring of Waste Disposal System Using IoT and Cloud for IoT Based Smart Cities. Wirel. Pers. Commun. 122(1): 243-275 (2022). https://doi.org/10.1007/s11277-021-08897-z

[13] Priscila Cedillo, Emilio Insfrán, Silvia Abrahão. Jean Vanderdonckt: Empirical Evaluation of a Method for Monitoring Cloud Services Based on Models at Runtime. IEEE Access 9: 55898-55919 (2021). https://doi.org/10.1109/ACCESS.2021.3071417

[14] Özgün Yilmaz, Levent Görgü, Michael J. O'Grady, Gregory M. P. O'Hare: Cloud-Assisted Mobile Crowd Sensing for Route and Congestion Monitoring. IEEE Access 9: 157984-157996 (2021). https://doi.org/10.1109/ACCESS.2021.3129932

[15] Chetan M. Bulla, Mahantesh N. Birje: A Multi-Agent-Based Data Collection and Aggregation Model for Fog-Enabled Cloud Monitoring. Int. J. Cloud Appl. Comput. 11(1): 73-92 (2021). https://doi.org/10.4018/IJCAC.2021010104

[16] Rajendra Kumar Dwivedi, Rakesh Kumar, Rajkumar Buyya: Secure Healthcare Monitoring Sensor Cloud with Attribute-Based Elliptical Curve Cryptography. Int. J. Cloud Appl. Comput. 11(3): 1-18 (2021). https://doi.org/10.4018/IJCAC.2021070101

[17] Joe-Air Jiang, Jen-Cheng Wang, Chao-Liang Hsieh, Kai-Sheng Tseng, Zheng-Wei Ye, Lin-Kuei Su, Chih-Hong Sun, Tzai-Hung Wen, Jehn-Yih Juang: An Alternative Body Temperature Measurement Solution: Combination of a Highly Accurate Monitoring System and a Visualized Public Health Cloud Platform. IEEE Internet Things J. 8(7): 5778-5793 (2021). https://doi.org/10.1109/JIOT.2020.3034024

[18] Tuan Anh Nguyen, Dugki Min, Eunmi Choi, Jae-Woo Lee: Dependability and Security Quantification of an Internet of Medical Things Infrastructure Based on Cloud-Fog-Edge Continuum for Healthcare Monitoring Using Hierarchical Models. IEEE Internet Things J. 8(21): 15704-15748 (2021). https://doi.org/10.1109/JIOT.2021.3081420