Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 9233267, 9 pages
https://doi.org/10.1155/2022/9233267

Research Article

Integration of Edge Computing and Blockchain for Provision of Data Fusion and Secure Big Data Analysis for Internet of Things

Jingya Dong, [1,2,3,4] **Chunhe Song**, [1,2,3,4] **Tao Zhang**, [1,2,3,4] **Yuanjian Li**, [2,5] **and Hao Zheng** [2,5]

[1] *Key Laboratory of Networked Control Systems, Chinese Academy of Sciences, Shenyang 110016, China*
[2] *Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China*
[3] *Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China*
[4] *University of Chinese Academy of Sciences, Beijing 100049, China*
[5] *College of Information, Liaoning University, Shenyang 110036, China*

Correspondence should be addressed to Chunhe Song; songchunhe@sia.cn

Received 5 April 2022; Accepted 4 May 2022; Published 18 May 2022

Academic Editor: Jing Chai

Intelligent computing provides efficient, real-time, and secure data analysis services for the Internet of Things (IoT). As the number of IoT devices increases, IoT generates massive, diverse, and multisourcing datasets that can be used to improve IoT services further. Models trained by intelligent computing from a single system or sensor are often not global, and sending all data directly to the computing platform wastes network bandwidth and may cause network congestion and even privacy leakage. To ensure IoT applications' quality of service and privacy, we propose a framework that integrates edge computing and blockchain to provide lightweight data fusion and secure data analysis for IoT. We propose a lightweight data fusion method that can reduce the amount of data at the node level and prevent network congestion and bandwidth waste. Furthermore, we propose a hierarchical fuzzy hashing method to check and locate anomalies of IoT machine learning models to ensure the validity of IoT intelligent computing and the security of sensitive data. Finally, we demonstrate the effectiveness of the method proposed in this paper through experiments.

1. Introduction

The Internet of Things (IoT) refers to the interconnection and integration of things in the physical world and cyberspace through the Internet or other communication networks [1]. With the increasing number of inexpensive information sensing embedded devices, and sensors, such as smartphones, mobile devices, wireless sensor network devices, and ubiquitous communication devices, IoT has played an essential role in various practical systems. At present, IoT has penetrated many scenarios such as medicine, industry, transportation, and agriculture, to provide intelligent services and applications [2].

With the addition of a large number of sensor devices, IoT generates massive, multisource, heterogeneous, and sparse diverse datasets [2]. The collected datasets can help improve IoT services and enhance smart services. However, it is unwise to directly send massive and multisource heterogeneous IoT data to the cloud, which will waste network bandwidth and even cause network congestion and packet loss [3]. Therefore, the original data collected by the node needs to be processed by data fusion before it can be transmitted to prevent network congestion or packet loss. Data fusion is performed locally by nodes to discard multiple copies of the same data, reduce the size and dimension of data, and optimize data quality to facilitate the extraction of useful information. Therefore, data fusion can promote a vital technology to mine essential data from extensive and complex data to improve quality and facilitate decision-making.

To provide efficient, real-time, and secure data analysis services for the IoT, the intelligent computing platform for the IoT has gradually entered the research field [4]. However, current intelligent computing platforms often suffer from two flaws. (1) IoT data obtained from discrete embedded devices or a single sensor alone often does not have global properties and even introduces abnormal modeling and training noise. (2) Centralized machine learning models may cause risks to the privacy of training data [5–7], and sensitive IoT data may cause disclosure of the actual physical process of industrial manufacturing sites.

To solve the above problems, we propose a distributed framework integrating blockchain and IoT to provide lightweight data fusion and secure big data analysis services for IoT. The edge nodes train the fused data to obtain a local model and transmit it to the cloud server. The cloud server obtains the global model by merging the local models. Our approach to training machine learning models in a distributed fashion consisting of edge nodes reduces the risk of data privacy breaches (as data does not leave the local training nodes). At the same time, we use the blockchain to ensure the consistency of the local model and the global model and propose a hierarchical fuzzy hashing method to detect and locate the abnormal position of the model. The main contributions of this paper are as follows:

(1) Propose a distributed structure combining edge computing and blockchain to provide lightweight data fusion and secure big data analysis for the IoT

(2) Propose a lightweight data fusion method, which uses adaptive algorithms and maximum and minimum functions to identify and delete redundant data to achieve node-level data fusion

(3) Propose a hierarchical fuzzy hashing method to check and locate the anomaly of the IoT machine learning model and ensure security

The rest of this paper is organized as follows. In Section 2, we outline the work related to our proposed method. In Section 3, we describe our proposed framework and algorithm in detail. The experimental results and performance evaluation are summarized in Section 4. Finally, we conclude in Section 5.

2. Related Work

2.1. IoT and Data Fusion. Many mathematical theories are used for data fusion in IoT. [8] introduced a method based on probability distribution to express the dependence between random variables. However, the method in [8] is difficult to obtain the density function and prior probability, and its performance is limited when processing complex multivariable data simultaneously. Hong et al. [9] propose an evidence-based reasoning method that introduces beliefs to represent uncertainty in the world. Song et al. [10] proposed a lightweight data analysis method. However, the quality function problem [9] limits its practical application. Chen et al. [11] proposed a multisensor data fusion method for load monitoring. Pan et al. [12] used the fault diagnosis method to fuse the data of the wind turbine gearbox. Lin et al. [13] proposed a data fusion strategy based on transfer learning for industrial networking. Dina et al. [14] used a spatiotemporal fusion method for the real-time fusion of IoT data. However, the methods of papers [11–14] often require prior knowledge or prior modeling and cannot be applied at the node level.

At present, some work has focused on the privacy protection of data fusion. Shang et al. [15] used probability density function values and distribution factors to fuse different sensor data but lacked attention to robustness and verifiability. Wang et al. [16] used differential privacy combined with deep learning to achieve privacy protection but did not consider robustness. Lin et al. [17] proposed a privacy-enhanced data fusion strategy for medical data, but this method is less versatile.

2.2. IoT with Blockchain and Edge Computing. As a paradigm shift, blockchain transforms the IoT field by providing a decentralized environment [18]. Yang et al. discussed the impact of blockchain on IoT in terms of security, privacy, scalability, etc. [19]. Lin et al. [20] pointed out the benefits that blockchain provides to smart cities and provided a research route for smart city computing resource transactions. Li et al. [21] investigated the benefits of blockchain decentralization for microgrid energy supply. Duan et al. [22] proposed a communication network architecture for the Internet of Vehicles with the introduction of edge computing. [23] discussed the integration trend of IoT and blockchain and introduced future research directions.

Asante et al. [24] pointed out that physical devices on the blockchain are often unavailable storage and computing resources, which requires edge computing to solve this problem. Wang et al. [25] utilized edge computing and blockchain to exchange medical IoT data efficiently. Guo et al. [26] used blockchain technology to solve the problem of information silos between IoT platforms. Jindal et al. [27] leveraged blockchain and edge computing to secure vehicle-to-grid energy transactions. Liu et al. [28] proposed a new wireless blockchain framework supporting mobile edge computing to offload intensive mining figures from edge computing nodes. Sun et al. [29] proposed a system combining federated learning, edge computing, and blockchain to achieve this model sharings.

3. Data Fusion and Security Data Analysis for IoT

Sensor nodes for various applications in IoT transmit their data to cloud data centers through network gateways and edge servers. Unstructured data and spatiotemporal stream data collected by sensors can affect data center processing speed and decision-making behavior, potentially causing bandwidth bottlenecks, latency, and throughput degradation for time-critical and latency-sensitive applications. Large amounts of redundant data will lead to packet loss, longer delays, and network congestion [30]. At the same time, a single centralized database solution is often not suitable for complex IoT systems, so we consider integrating the distributed structure of blockchain and IoT to provide secure big data analysis services. This section discusses our proposed data fusion method and security protection strategy to eliminate redundant data and provide secure and effective data analysis services for IoT applications.

3.1. IoT Framework Integrating Edge Computing and Blockchain. In IoT applications, if the data collected by the IoT application layer is transmitted to the cloud layer directly or through the edge, it may cause congestion and significant delay at the cloud service layer. Therefore, we add data fusion at the nodes for optimization. IoT often involves private data, such as electricity consumption data in smart grids, medical and health data, and other private data that should not be leaked. At the same time, the IoT network requires real-time monitoring, forecasting, and other services, such as traffic forecasting, fire alarms, etc., and the data often needs to be processed locally. Given the above, in our model, edge nodes will perform local model training on the data locally to protect the privacy of the data. After the edge node transmits the local model parameters to the cloud server, it will train the global model and pass it to the edge node. The edge node will use the data analysis results or model parameters of the cloud server to process the data in time to meet the timeliness of the IoT network. At the same time, in the proposed architecture, the consistency and validity of local data and cloud data transmission verified by blockchain are used to prevent poisoning attacks. Figure 1 shows the overall framework of this paper.

3.2. Lightweight Data Fusion Method. To eliminate correlated and redundant data in IoT, we use a lightweight data fusion method at the node level. The proposed method utilizes adaptive distance exponent and minimax functions at the node level to identify and remove redundant data [31, 32]. Our method can be flexibly extended to any application. Through our method, on the one hand, redundant data can be identified and removed, and on the other hand, the changing trend of the data can be keenly captured to ensure the quality of the data. At the same time, due to the existence of the adaptive distance index, our method can automatically fuse the data even without prior knowledge.

The parameters of the fusion algorithm, including the maximum and minimum values, the distance index, the fusion index, and two lists, are stored in the node's buffer. The maximum value, minimum value, distance index, and fusion index are denoted by max, min, γ, and α, respectively. The fusion index α is a constant, and $\alpha > 0$. The larger the fusion index is, the lower the fusion rate is; the smaller the fusion index is, the higher the fusion rate is. The two lists are the distance interval list and the data packet list, respectively, denoted by L and D. At time i, the sensor value is denoted as N_i. In the beginning, when a new sensor value N_0 is sensed by a node, then max = min = N_i, $\gamma = (\max - \min)/\alpha$. The parameter changing method is shown in

$$f(t_i) = \begin{cases} \min = N_i, \gamma = \dfrac{\max - \min}{\alpha}, \text{if } N_i < \min, \\ \max = N_i, \gamma = \dfrac{\max - \min}{\alpha}, \text{if } N_i > \max. \end{cases} \quad (1)$$

The parameters max and min are used to capture the magnitude of the data change and draw a fixed point on the data change curve. It is assumed that the sampling rate of each node is Sr packets per second, where $Sr \geq 1$. The node sends data to the gateway at intervals T. In our method, data fusion is performed at each time interval T period. After the sensor value is entered into the node, the value will first update the min, max, and γ values. N_i and its distance from the previous packet are added to the distance list. We use the updated γ value to detect the distance list, delete the data packets that do not meet the requirements of the data list, and update the distance list. The data fusion steps are shown in Algorithm 1.

Our method automatically adjusts the distance index to fit the characteristics of the data according to the varying range of the data. According to the distance index, the data is fused while retaining the extreme points of the data. Our method perfectly preserves the data features and reduces the fusion loss. At the same time, the fusion rate is used to flexibly adjust the sensitivity of the data to adapt to different scenarios. The fused data significantly reduces data redundancy, packet collision probability, and network delay and reduces network congestion.

3.3. Secure and Privacy-Preserving Anomaly Detection and Localization Algorithms. After data fusion, each node transmits the fused data to the cloud data center through network gateways and edge servers. The gateway acts as a relay node to monitor IoT devices and offload data to edge servers. The collaborative work of the edge service layer and the cloud service layer provides efficient, accurate, and real-time data processing and decision support for IoT applications [33].

The edge service consists of edge nodes, and the cloud service layer consists of peer cloud servers. In our model, the edge node is responsible for training the local model, and the cloud server merges the local model to obtain the global model. The edge node transmits the trained local model parameters to the cloud server, and the cloud server trains the global model according to all collected local models. Our strategy addresses the defect that local data cannot represent global attributes in IoT and guarantees local data privacy. Machine learning scenarios in IoT, such as smart medical care and smart grid, may attack machine learning models [1]. Here, we consider the use of blockchain technology to build defense mechanisms. That is to use the blockchain to protect the consistency and immutability of the local model and the global model. This method can improve the scalability of the IoT network, eliminate single points of failure, and ensure the model's validity. Figure 2 shows the basic idea of the hierarchical fuzzy hashing algorithm. The local models of edge nodes differ, as illustrated in Figure 2, due to the diverse types of sensors used by edge nodes.

We introduce the blockchain to record the parameters of local and global models to process and check the erroneous input of information. The variability of the data recorded by the blockchain can prevent attackers from changing the model parameters after training. That is, the blockchain constitutes a defense tool to model attacks. Referring to [4, 34],

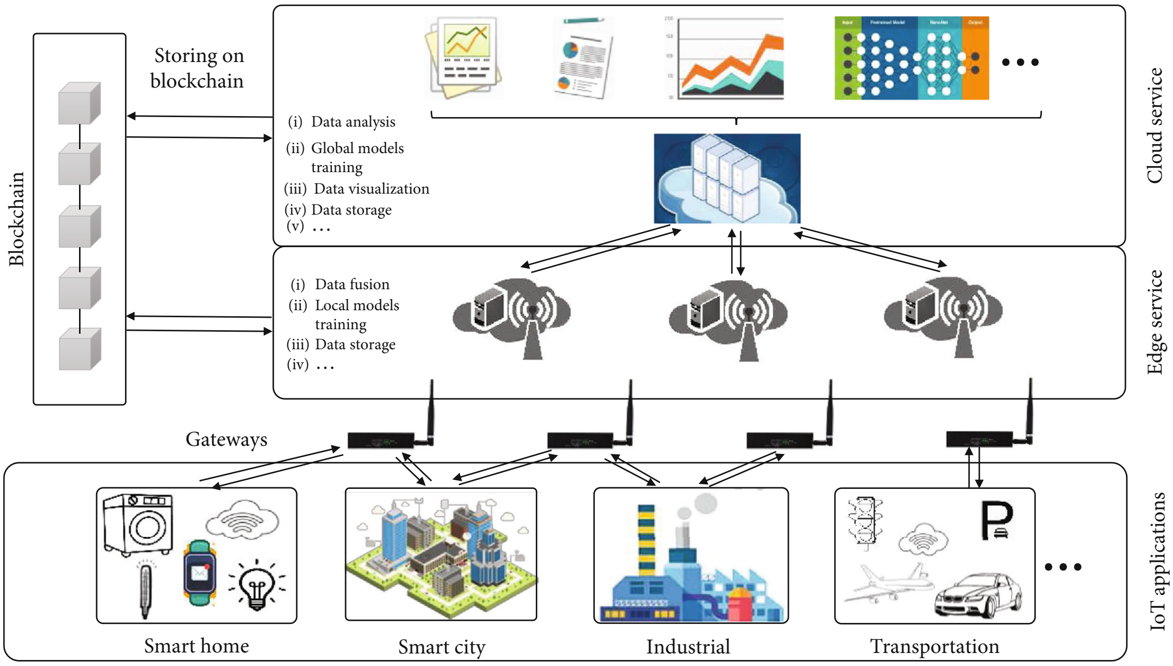


Figure 1: Overall framework.

```
**Input:** max, min, $\gamma$, $L = [\,]$, $D = [\,]$, $N_i$, $T$, $\alpha$
**Output:** $D = [D_1, D_2, \cdots]$
  for $i$ in range $(0, T)$:
    if $i = 0$:
      max = min = $N_i$
      $D$.append $(N_i)$
    else:
      $D$.append $(N_i)$
      $L$.append $(\|D_i - D_{i-1}\|)$
    if $N_i >$ max: max $= N_i$
    if $N_i <$ min: min $= N_i$
  $\gamma = (\max - \min)/\alpha$
  $n = \mathrm{length}(L)$
  for $l$ in range $(0, n)$:
    if $L[l] < \gamma$:
      if $D[l+1] ==$ max or $D[l+1] ==$ min:
        $D = \mathrm{del}\,(D[l])$
        break
      else:
        $D = \mathrm{del}\,(D[l+1])$
        break
    end if
  end for
end for
return $D$
```

Algorithm 1: Data fusion method.

we consider using a hierarchical fuzzy hash function to replace the information record structure of the traditional blockchain.

We propose the method of hierarchical fuzzy hashing, mainly considering the contingency of IoT attacks. The amount of data transmitted in the network through hierarchical fuzzy hashing is reduced, and data security is guaranteed. We divide the blockchain into two levels of information, the local chain and the global chain. The underlying information consists of a fuzzy hash function, and the fuzzy hash value can detect abnormal changes in model parameters. To improve the transmission efficiency, we save the hash of the fuzzy hash value in the global information. Figure 3 shows the basic idea of distributed anomaly detection and location algorithm combined with blockchain.

We use a hierarchical fuzzy hashing algorithm to reduce data transmission loss by preserving subinformation of information. The global model is recorded in the global chain using a hashed hash. The edge node hashes the record information of the global chain and compares it with the record information of the local chain. An abnormality is detected using the fuzzy hash value detection of the global chain and local chain records when there is an abnormality. We use the fuzzy hash value to calculate the Hamming distance between the models, determining the degree of attack on the model and locating the attacked position. Our hierarchical fuzzy hashing method and anomaly detection algorithm improve the efficiency of the IoT network and ensure the model's security under the condition of guaranteeing anomaly detection.

4. Experimental Results

In this section, we evaluate the efficiency of our proposed lightweight data fusion and the effectiveness of the hierarchical fuzzy hashing security strategy based on experimental metrics.

4.1. Data Fusion Algorithm Test. For data fusion, we perform data fusion at the node level. We use the data collected by

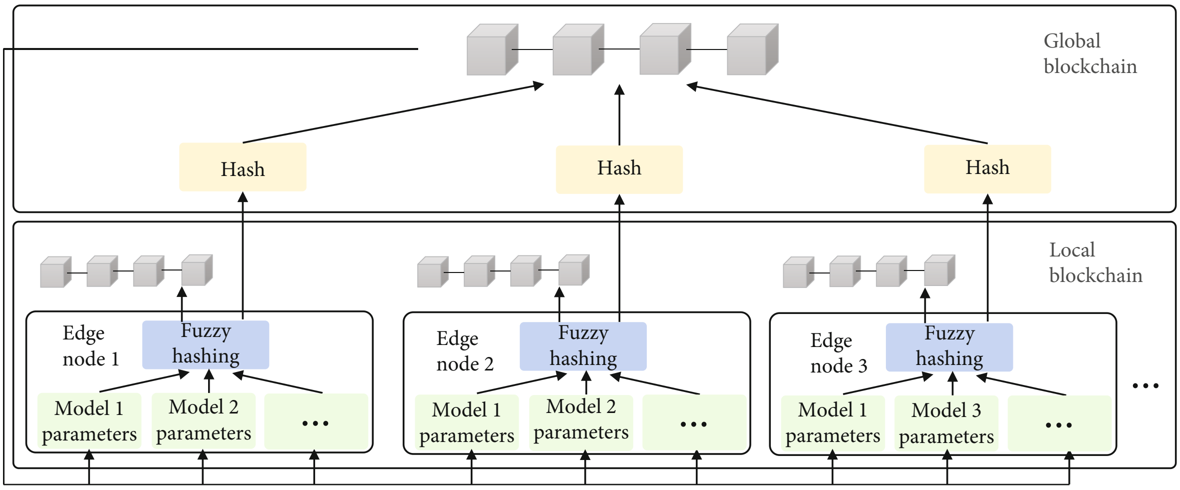


Figure 2: Hierarchical fuzzy hashing algorithm.

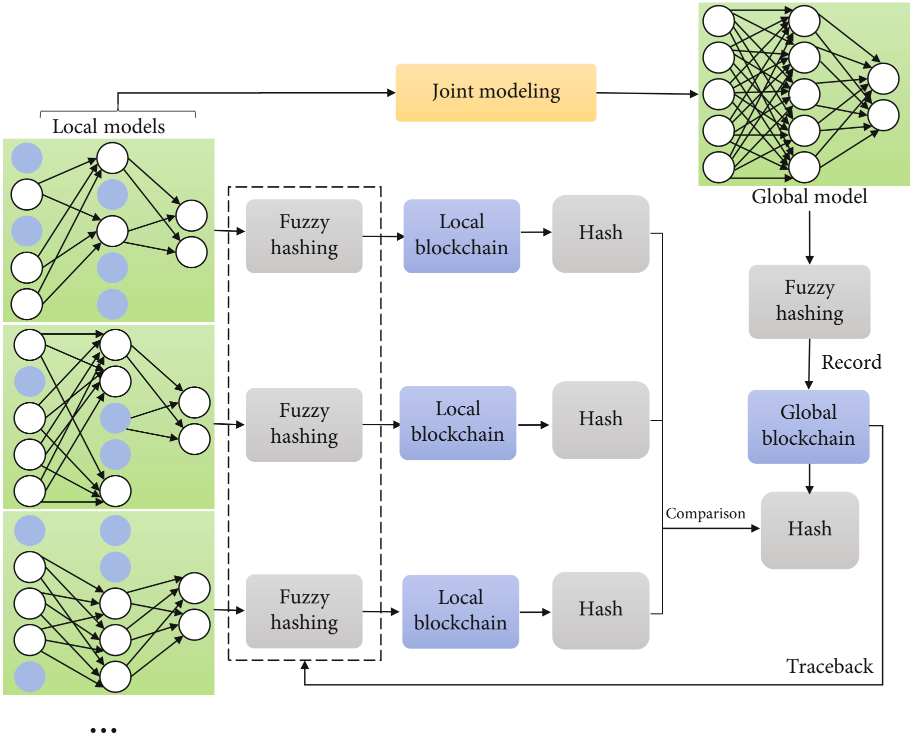


Figure 3: Distributed anomaly detection and localization algorithm combined with blockchain. (1) The node compares the hash value of the local chain to the hash value of the global chain to see whether there is an abnormality. (2) After the abnormality occurs, the node locates the anomaly by comparing the local model's fuzzy hashing value to the global chain's fuzzy hashing value.

the Intel Berkeley Research Lab [35] to test the data fusion algorithms' performance and fusion rate.

We first measured the capture of data features by the lightweight fusion algorithm. Taking the temperature sensor data of a particular day as an example, we take one hour as the fusion period, and the fusion result is shown in Figure 4. Figure 4 shows the data packets transmitted by our proposed lightweight data fusion algorithm under different fusion indexes. As seen from the figure, our proposed algorithm dramatically reduces the number of packets. At the same time, we can find that our algorithm can reduce the number of packets while ensuring the characteristics and quality of the data. Experimental results demonstrate that our method reduces data redundancy, packet collision probability, and reduces network congestion.

We tested the fusion performance of the data fusion algorithm. We compare existing data fusion methods, including stratified sampling [3] and EECC [36]. Figure 5 shows the proportion of data transmitted to the gateway by different data fusion methods. In Figure 5, the horizontal

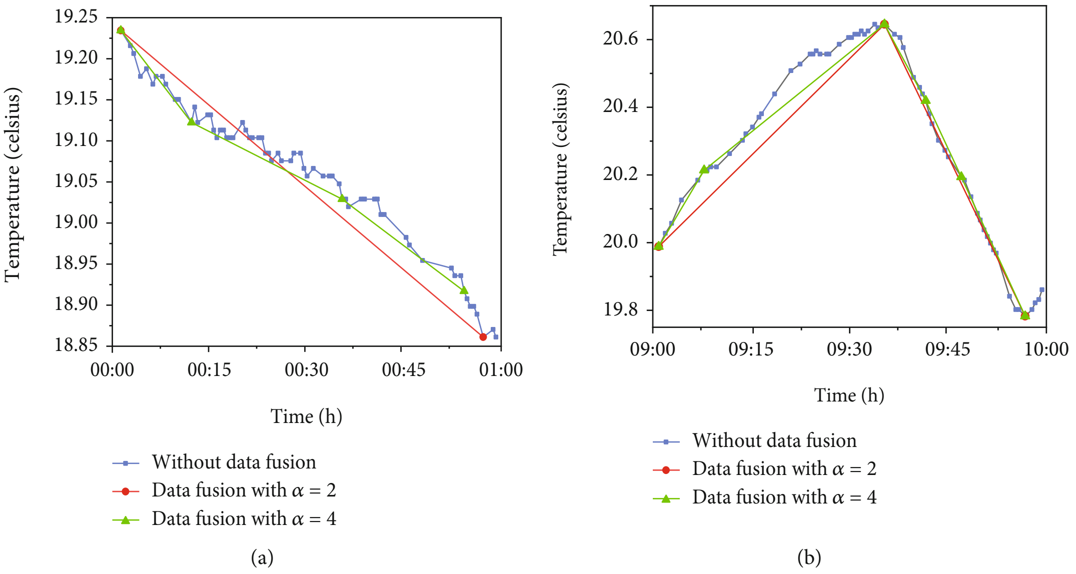


Figure 4: Data fusion in different periods. (a) 00: 00-01: 00 stage, data fusion situation. (b) 09: 00-10: 00 stage, data fusion.

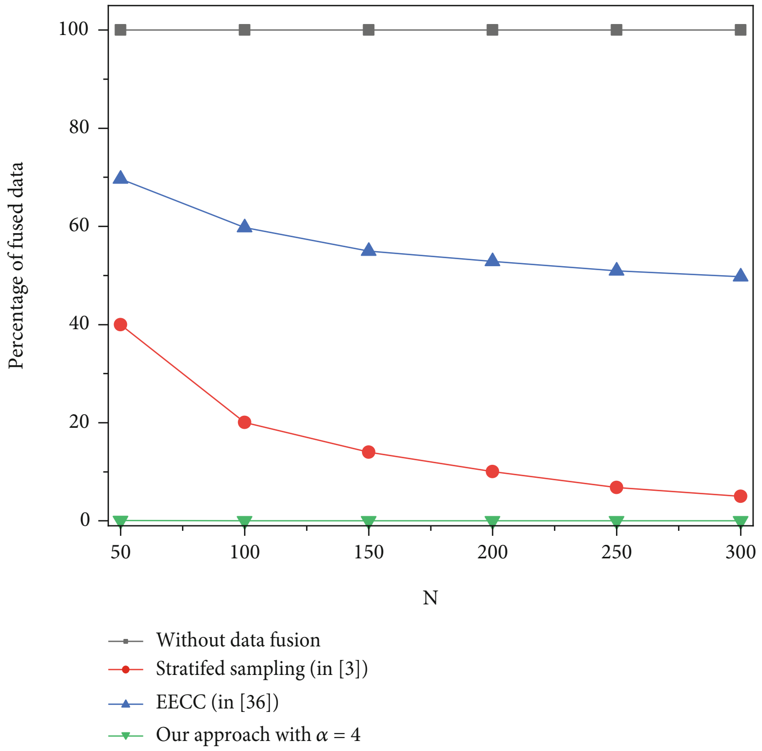
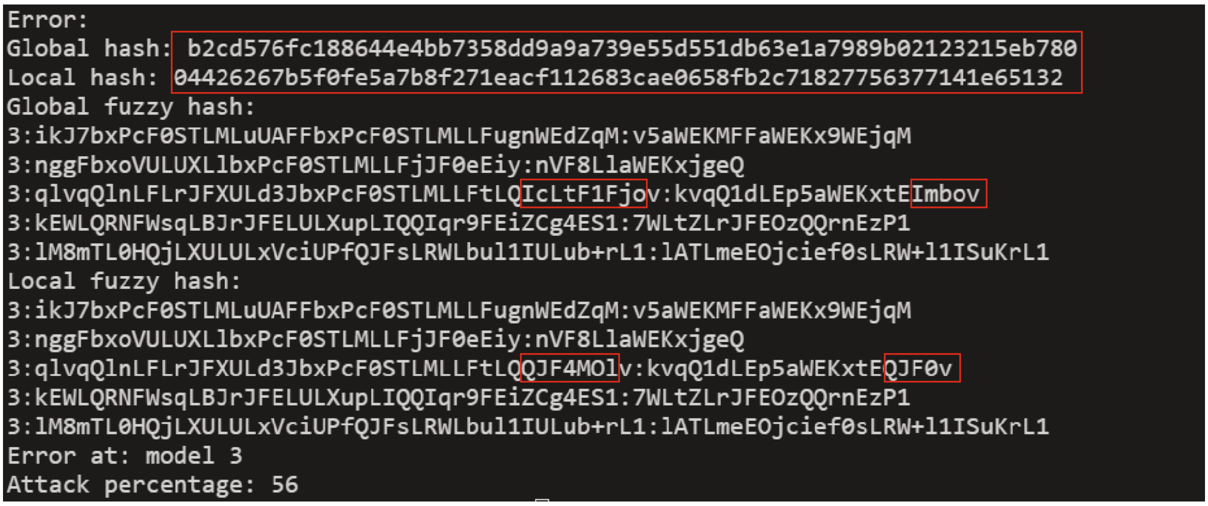


Figure 5: Data fusion rate.

axis N represents the total number of data packets, the vertical axis is the percentage of transmitted data packets $n = N'/N$, and N' is the number of compressed data packets. In Figure 5, our method significantly outperforms other algorithms in compression performance. During the fusion process, stratified sampling takes the extreme value of each layer after stratification. The stratification leads to a higher percentage of fused data delivered by the gateway, while EECC increases the number of datagrams due to the transmission of multiple copies. This experiment proves that our method has a better data fusion effect than existing algorithms.

4.2. Anomaly Detection and Localization Algorithm Test. We tested the model anomaly detection and anomaly localization methods. We consider the case where an attacker produces anomalous models by retraining the model with additional data containing Trojan triggers. The main



```
Error:
Global hash:  b2cd576fc188644e4bb7358dd9a9a739e55d551db63e1a7989b02123215eb780
Local hash: 04426267b5f0fe5a7b8f271eacf112683cae0658fb2c71827756377141e65132
Global fuzzy hash:
3:ikJ7bxPcF0STLMLuUAFFbxPcF0STLMLLFugnWEdZqM:v5aWEKMFFaWEKx9WEjqM
3:nggFbxoVULUXLlbxPcF0STLMLLFjJF0eEiy:nVF8LlaWEKxjgeQ
3:qlvqQlnLFLrJFXULd3JbxPcF0STLMLLFtLQIcLtF1Fjov:kvqQ1dLEp5aWEKxtEImbov
3:kEWLQRNFWsqLBJrJFELULXupLIQQIqr9FEiZCg4ES1:7WLtZLrJFEOzQQrnEzP1
3:lM8mTL0HQjLXULULxVciUPfQJFsLRWLbul1IULub+rL1:lATLmeEOjcief0sLRW+l1ISuKrL1
Local fuzzy hash:
3:ikJ7bxPcF0STLMLuUAFFbxPcF0STLMLLFugnWEdZqM:v5aWEKMFFaWEKx9WEjqM
3:nggFbxoVULUXLlbxPcF0STLMLLFjJF0eEiy:nVF8LlaWEKxjgeQ
3:qlvqQlnLFLrJFXULd3JbxPcF0STLMLLFtLQQJF4MOlv:kvqQ1dLEp5aWEKxtEQJF0v
3:kEWLQRNFWsqLBJrJFELULXupLIQQIqr9FEiZCg4ES1:7WLtZLrJFEOzQQrnEzP1
3:lM8mTL0HQjLXULULxVciUPfQJFsLRWLbul1IULub+rL1:lATLmeEOjcief0sLRW+l1ISuKrL1
Error at: model 3
Attack percentage: 56
```

Figure 6: An example of anomaly detection and localization based on hierarchical fuzzy hashing.

purpose of our method is to check whether the model is poisoned by comparing it with a benign model and then using the local fuzzy hash to locate the abnormal location and give the poisoning ratio of the model.

We test the feasibility and effectiveness of our proposed method on anomaly detection and localization of parameters using a set of machine learning models. We focus on how the model detects and locates anomalies. We consider the global model to be benign. The experiments assume that the attacker leads to the generation of abnormal models by retraining the model with additional data containing Trojan triggers. We first use the two-layer hash value comparison of the local and global models to check whether the model is poisoned. When the model poisoning is detected, we use the fuzzy hash value to locate the abnormal location and give the model poisoning ratio.

Figure 6 shows the hierarchical fuzzy hashing algorithm's anomaly detection and localization process after model poisoning. As shown in Figure 6, after the global hash and the local hash are entirely different, the fuzzy hash value will be backtracked (when the global hash is consistent with the local hash, this step is not performed). It can be seen from Figure 6 that the poisoning model is not completely inconsistent with the correct model. According to the difference of fuzzy hash value, we can locate the poisoning position and poisoning degree of the model. This experiment demonstrates our proposed hierarchical hashing, checking anomaly models and locating anomalous locations.

This experiment mainly provides an extensive evaluation of the various performances of our proposed hierarchical hash function. To compare the performance of the hierarchical hash function on the blockchain system, we will first take the method that does not use machine learning and hierarchical fuzzy hashing as the basic method (the edge nodes directly transmit all data to the cloud server for processing). Meanwhile, we use the fuzzy hashing method used in the paper [5] as another model for comparison. We use data from the same sensor for performance testing.

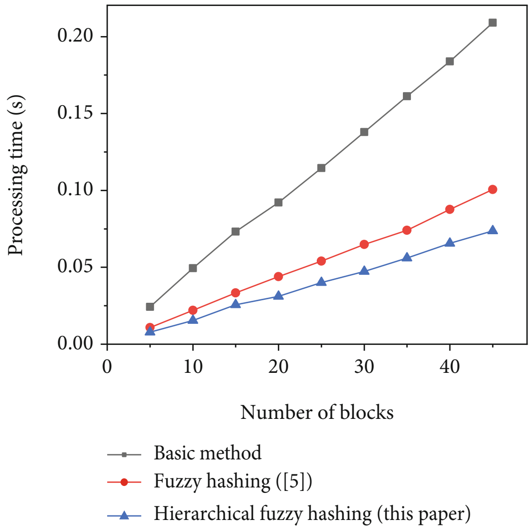


Figure 7: Overall performance evaluation.

We contrast the proposed layering with the paper's base method and the method [5]. At the same time, for simplicity, we use the central node as the verification node to improve the system's efficiency. That is to say, after the central node (cloud service) has passed the operation, it can be written to the global chain, and each edge node maintains its local chain and periodically transmits encrypted data to the cloud service. In this experiment, there are five edge nodes and one central server. Figure 7 shows the relationship between the number of processing blocks of a node and the required time.

We also tested end-to-end latency. We simulate the process of storing and exception queries (as shown in Figure 8). Compared with other methods, our proposed hierarchical fuzzy hashing algorithm requires less time on average to process packets. As shown in Figure 8, the hierarchical fuzzy hashing proposed by us significantly improves the processing power of the blockchain system.

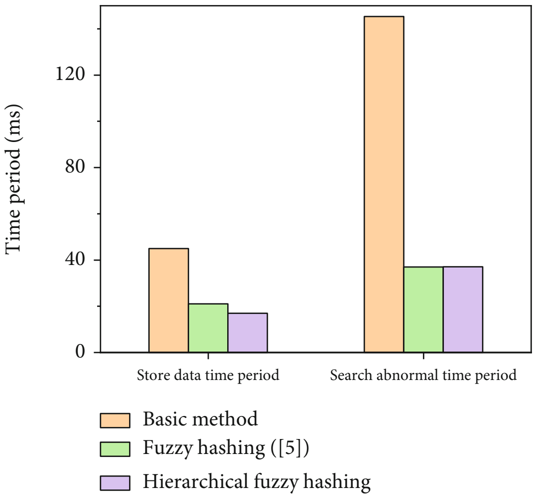


Figure 8: End-to-end delay testing.

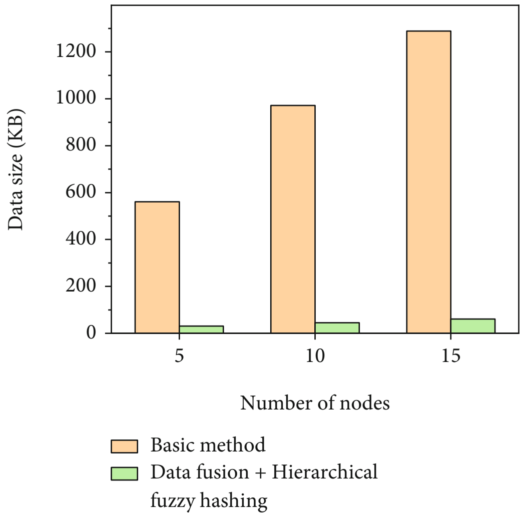


Figure 9: Data packet overhead before and after data fusion.

4.3. Data Packet Overhead. We finally test a comparative experiment of the overhead of data transfer packets for the overall experiment. The purpose of the experiment is to compare the impact of our proposed data fusion algorithm and hierarchical hash function on the data transmission overhead of IoT. We calculated the comparison of packet sizes transmitted by edge nodes to cloud servers in the case of 5, 10, and 15 nodes. As shown in Figure 9, the packet size processed by our method is much smaller than the packet size before processing. The above experiments demonstrate that the use of data fusion, machine learning, and hierarchical fuzzy hashing can positively impact IoT data analysis services, which can reduce time and transfer data volume while improving data confidentiality and fault tolerance.

5. Conclusion

In this paper, given the current situation of the rapid increase in the scale and quantity of IoT data, we propose a method to integrate edge computing and blockchain to provide data fusion and secure big data analysis for the IoT. We propose a node-level lightweight data fusion method and a hierarchical fuzzy hashing method, which are used to reduce the amount of data transmitted by IoT and ensure the security of IoT data. We reduce the amount of data transfer at the node level using a lightweight data fusion method. The edge node calculates the local model to reduce the pressure on the cloud server, and the cloud node is used to merge the local model to improve the representativeness of the model. We utilize blockchain and hierarchical fuzzy hashing to ensure the consistency and validity of local and global models while maintaining data privacy. Our experimental results show that our method dramatically reduces the amount of data transmitted by the IoT and improves the system's security. In the future, our goal is to solve the load imbalance problem of edge nodes to enhance the utilization of edge devices.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

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

Acknowledgments

This paper is supported by the Jiangsu Provincial Key Research and Development Program under Grant BE2020001-1, independent project of State Key Laboratory of Robotics (2022-Z03), and Liaoning Natural Science Foundation (2021-MS-030).

References

[1] Y. Li, Y. Zuo, H. Song, and Z. Lv, "Deep learning in security of Internet of Things," *IEEE Internet of Things Journal*, p. 1, 2021.

[2] C. Song, Y. Sun, G. Han, and J. Rodrigues, "Intrusion detection based on hybrid classifiers for smart grid," *Computers and Electrical Engineering*, vol. 93, p. 107212, 2021.

[3] M. Jan, M. Zakarya, M. Khan et al., "An AI-enabled lightweight data fusion and load optimization approach for Internet of Things," *Future Generation Computer Systems*, vol. 122, pp. 40–51, 2021.

[4] C. Song, G. Han, and P. Zeng, "Cloud computing based demand response management using deep reinforcement learning," *IEEE Transactions on Cloud Computing*, vol. 10, no. 1, pp. 72–81, 2022.

[5] D. Unal, M. Hammoudeh, M. A. Khan, A. Abuarqoub, G. Epiphaniou, and R. Hamila, "Integration of federated machine learning and blockchain for the provision of secure big data analytics for Internet of Things," *Computers & Security*, vol. 109, p. 102393, 2021.

[6] F. Zhu, J. Gao, J. Yang, and N. Ye, "Neighborhood linear discriminant analysis," *Pattern Recognition*, vol. 123, p. 108422, 2022.

[7] F. Zhu, J. Gao, C. Xu, J. Yang, and D. Tao, "On selecting effective patterns for fast support vector regression training," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3610–3622, 2018.

[8] C. Song, P. Zeng, Z. Wang, T. Li, L. Qiao, and L. Shen, "Image forgery detection based on motion blur estimated using convolutional neural network," *IEEE Sensors Journal*, vol. 19, no. 23, pp. 11601–11611, 2019.

[9] X. Hong, C. Nugent, M. Mulvenna, S. McClean, B. Scotney, and S. Devlin, "Evidential fusion of sensor data for activity recognition in smart homes," *Pervasive and Mobile Computing*, vol. 5, no. 3, pp. 236–252, 2009.

[10] C. Song, S. Liu, G. Han, P. Zeng, H. Yu, and Q. Zheng, "Edge intelligence based condition monitoring of beam pumping units under heavy noise in the industrial Internet of Things for Industry 4.0," *IEEE Internet of Things Journal*, p. 1, 2022.

[11] X. Chen, C. Song, and T. Wang, "Spatiotemporal analysis of line loss rate: a case study in China," *Energy Reports*, vol. 7, pp. 7048–7059, 2021.

[12] Y. Pan, R. Hong, J. Chen, J. Singh, and X. Jia, "Performance degradation assessment of a wind turbine gearbox based on multi- sensor data fusion," *Mechanism and Machine Theory*, vol. 137, pp. 509–526, 2019.

[13] H. Lin, J. Hu, X. Wang, M. Alhamid, and M. Piran, "Toward secure data fusion in industrial IoT using transfer learning," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 10, pp. 7114–7122, 2021.

[14] F. Dina, S. Moussa, and N. Badr, "The spatiotemporal data fusion (STDF) approach: IoT-based data fusion using big data analytics," *Sensors*, vol. 21, no. 21, p. 7035, 2021.

[15] W. Shang, J. Cui, C. Song, J. Zhao, and P. Zeng, "Research on industrial control anomaly detection based on FCM and SVM," in *2018 17th IEEE international conference on trust, security and privacy in computing and communications/12th IEEE international conference on big data science and engineering*, pp. 218–222, New York, NY, USA, Aug. 2018.

[16] Z. Wang, Y. Fu, C. Song, P. Zeng, and L. Qiao, "Power system anomaly detection based on OCSVM optimized by improved particle swarm optimization," *IEEE Access*, vol. 7, no. 1, pp. 181580–181588, 2019.

[17] H. Lin, S. Garg, J. Hu, X. Wang, M. J. Piran, and M. S. Hossain, "Privacy-enhanced data fusion for COVID-19 applications in intelligent Internet of Medical Things," *IEEE Internet of Things Journal*, vol. 8, no. 21, pp. 15683–15693, 2021.

[18] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke, "Blockchain technologies for the Internet of Things: research issues and challenges," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2188–2204, 2018.

[19] W. Yang, E. Aghasian, S. Garg, D. Herbert, L. Disiuta, and B. Kang, "A survey on blockchain-based internet service architecture: requirements, challenges, trends, and future," *IEEE Access*, vol. 7, pp. 75845–75872, 2019.

[20] X. Lin, J. Wu, S. Mumtaz, S. Garg, J. Li, and M. Guizani, "Blockchain-based on-demand computing resource trading in IoV-assisted smart city," *Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1373–1385, 2021.

[21] J. Li, Z. Zhou, J. Wu et al., "Decentralized on-demand energy supply for blockchain in Internet of Things: a microgrids approach," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1395–1406, 2019.

[22] W. Duan, J. Gu, M. Wen, G. Zhang, Y. Ji, and S. Mumtaz, "Emerging technologies for 5G-IoV networks: applications, trends and opportunities," *IEEE Network*, vol. 34, no. 5, pp. 283–289, 2020.

[23] S. Shivam, B. Bhushan, and M. Ahad, "Blockchain based solutions to secure IoT: background, integration trends and a way forward," *Journal of Network and Computer Applications*, vol. 181, p. 103050, 2021.

[24] M. Asante, G. Epiphaniou, C. Maple, H. Al-Khateeb, M. Bottarelli, and K. Z. Ghafoor, "Distributed ledger technologies in supply chain security management: a comprehensive survey," *IEEE Transactions on Engineering Management*, 2021.

[25] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Network*, vol. 33, no. 5, pp. 156–165, 2019.

[26] S. Guo, X. Hu, S. Guo, X. Qiu, and F. Qi, "Blockchain meets edge computing: a distributed and trusted authentication system," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 1972–1983, 2020.

[27] A. Jindal, G. Aujla, and N. Kumar, "SURVIVOR:A blockchain based edge-as-a-service framework for secure energy trading in SDN-enabled vehicle-to-grid environment," *Computer Networks*, vol. 153, pp. 36–48, 2019.

[28] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Computation offloading and content caching in wireless blockchain networks with mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11008–11021, 2018.

[29] Y. Sun, C. Song, S. Yu, Y. Liu, H. Pan, and P. Zeng, "Energy-efficient task offloading based on differential evolution in edge computing system with energy harvesting," *IEEE Access*, vol. 9, pp. 16383–16391, 2021.

[30] F. Zhu, Y. Ning, X. Chen, Y. Zhao, and Y. Gang, "On removing potential redundant constraints for SVOR learning," *Applied Soft Computing*, vol. 102, p. 106941, 2021.

[31] H. Zheng, Z. Shi, C. Zhou, M. Haardt, and J. Chen, "Coupled coarray tensor CPD for DOA estimation with coprime L-shaped array," *IEEE Signal Processing Letters*, vol. 28, pp. 1545–1549, 2021.

[32] C. Zhou, Y. Gu, S. He, and Z. Shi, "A robust and efficient algorithm for coprime array adaptive beamforming," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 2, pp. 1099–1112, 2018.

[33] C. Song, W. Xu, G. Han, P. Zeng, Z. Wang, and S. Yu, "A cloud edge collaborative intelligence method of insulator string defect detection for power IIoT," *IEEE Internet of Things Journal*, vol. 8, no. 9, pp. 7510–7520, 2021.

[34] J. Kornblum, "Identifying almost identical files using context triggered piecewise hashing," *Digital Investigation*, vol. 3, pp. 91–97, 2006.

[35] S. Madden, *Intel Berkeley research lab data*, 2003, http://berkeley, intel-research, net/labdata, html.

[36] S. Jan, M. A. Jan, R. Khan, H. Ullah, M. Alam, and M. Usman, "An energy-efficient and congestion control data-driven approach for cluster-based sensor network," *Mobile Networks and Applications*, vol. 24, no. 4, pp. 1295–1305, 2019.