*Turkish Journal of Computer and Mathematics Education Vol.12 No.9 (2021), 3089-3097*

*Research Article*

# Optimization Of Heterogeneous Iot Information Using Blockchain-Based Multiple Polynomials To Minimize Data Delay In Cloud Edge Environments

**Seung-Soo Shin**

Professor, Department of Information Security, Tongmyong University, 428, Sinseon-ro, Nam-gu, 48520, Busan

shinss@tu.ac.kr

Corresponding author :

**Article History**: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 20 April 2021

**Background/Objectives**: The information of IoT devices processed in a cloud environment has a costly problem in optimizing information of IoT devices due to different types of information generated by IoT devices and large-scale information processing. In the cloud environment, reducing costs and increasing transmission capacity remain the biggest issues to efficiently manage information generated by heterogeneous IoT devices.

**Methods/Statistical analysis**: The various parameters used for simulation in the proposed technique performance evaluation based on the mean of Monte Carlo simulations. Among the performance evaluation results of the proposed techniques derived in this environment, IoT processing time averaged 11.85% improvement since IoT processing time is classified into multi-subset sized network regions and then generated cumulative use of transactions through similarity between IoT information.

**Findings**: In this paper, we propose an optimization management technique for heterogeneous IoT information using hierarchical distributed polynomials to minimize the cost of processing information generated by IoT devices. The proposed technique not only enables batch distribution processing of IoT information by deep learning IoT information, but also performs multi-dimensional distribution processing of IoT information hierarchically in the process of sending and receiving IoT information.

**Improvements/Applications**: The proposed technique can synchronize the frequency of use according to the number of IoT information by applying the n-order distribution of IoT information to manage IoT information as efficiently as possible. The proposed technique improves bandwidth and processing time over existing techniques in the process of sending and receiving large amounts of IoT information in a short time when distributing heterogeneous IoT information during the IoT information linkage process.

***Keywords*:** Artificial IoT, Cloud Edge, Low Latency, Information collection, Distribution Polynomial.

## 1. Introduction

With the development of communication technology, various intelligent IoT technologies combined with IoT and artificial intelligence are developed, and in a distributed cloud environment, edge computing plays a role instead of cloud servers (or data centers) to collect/process/analyze information collected from IoT devices. Many changes are also being made in IoT information processing because edge computing can minimize the fast processing of collected data and the overhead of the network [1-3]. However, since information collected from heterogeneous IoT devices distributed and used in distributed networks is widely used in cloud environments, the demand for low processing costs and management efficiency of IoT information is steadily increasing [3,4].

In cloud edge networks, IoT devices can have various problems due to their surroundings and IoT self-malfunctioning, but they are solving these problems using blockchain technology [5]. Blockchain is used in various fields (economic, social, etc.) as well as in the IT sector, and one of the reasons why blockchain is popular in various fields is that it maintains a consistent ledger so that it can trade with each other without a central server.

In edge cloud environments, complexity and reliability are guaranteed by distributing complex and energy-consuming machine learning processes between edges and clouds, but latency and energy consumption between IoT devices operating in edge environments are still one of the challenges. Intelligent IoT can choose bandwidth allocation, offload, and relays to reduce latency of IoT networks and energy consumption of IoT devices. However, network optimization methods considering the loss of resources within IoT devices should be constantly studied. In the intelligent IoT environment, research is underway to improve the analysis evaluation time and accuracy of IoT collection information. However, although intelligent IoT requires applicable models at different compute layers, the prediction accuracy of IoT data may vary accordingly because different models are used in the cloud environment where IoT is being used whenever new events occur.

In this paper, we propose an optimization management technique for IoT information that can dynamically send and receive IoT information to minimize the cost of processing information sent and received from heterogeneous devices. The proposed technique applies blockchain to optimize the information of IoT devices using hierarchical distributed polynomials. In order to minimize IoT information loss, the proposed technique uses a method to synchronize IoT information frequency after n-order classification of IoT information according to the frequency of use. Furthermore, the proposed technique deep learning IoT information to optimize IoT information and then distributing IoT information in batches so that it is hierarchically multi-step structure. Through this process, the proposed technique has features that enable efficient management of large-scale IoT information while lowering the cost of processing IoT information.

The composition of this paper is as follows. Chapter 2 analyzes the environment and existing studies related to IoT information processing, and Chapter 3 proposes an information optimization management technique for heterogeneous IoT devices using hierarchical distributed polynomials. In Chapter 4, existing techniques are analyzed and concluded in Chapter 5.

## 2. Prelliminaries

### 2.1. Cloud Edge Computing vs. Blockchain

Cloud edge computing is one of the networks that came out to address the challenges of the cloud (such as high bandwidth and latency). Cloud edge computing does not pass information collected from IoT devices to cloud servers (or data centers), but it places intelligent IoT between IoT devices and cloud servers (or data centers) to handle what cloud servers (or data centers) need to handle. Intelligent IoT refers to a convergence technology that can be utilized according to the IoT environment and characteristics by combining Internet of Things technology with artificial intelligence technology [6]. Since the development of cloud technology, the speed and amount of data in the network have increased exponentially, resulting in massive amounts of data sent and received between IoT devices and cloud servers (or data centers) causing various problems in cloud operations (database capacity excess, data processing time delay, DDoS security attacks, etc.). These challenges pose many challenges for cloud services to operate normally, especially the lack of memory resources for limited computing and intelligent IoT devices [7].

Figure 1 shows the environment and process of intelligent IoT operating in various smart environments (smart cities, smart homes, smart industries, smart healthcare, wearable, etc.). To meet the capabilities of intelligent IoT operating in Fig. 1, efficient management of data processed in environments such as cloud computing platforms and intelligent IoT applications is required. Furthermore, information protection and network delay of massive amounts of data sent and received between intelligent IoT devices and cloud servers (or data centers) must be addressed.

Recently, there has been a growing number of platforms providing cloud services that apply blockchain to distribute and process information collected from IoT devices. This is because blockchain can effectively handle data distributed processing, which distributes and stores data for all users participating in the network. Blockchain has the structure of a chain connected sequentially over time after forming blocks, so it can contrast and verify information held by all users using the service, minimizing data protection and delay.

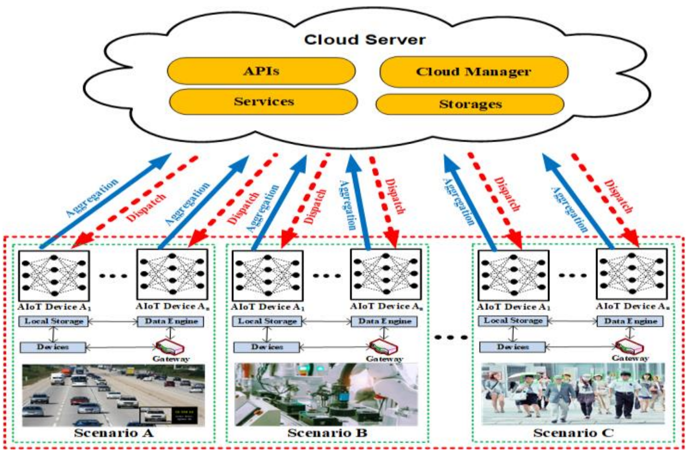


**Figure 1. An architecture of AIoT and cloud server**

Blockchain is difficult to falsify or alter distributed networks because it can split/store multiple data in multiple environments, such as Figure 1. If a distributed network is to be falsified, network hacking is virtually impossible because all users who participate in the blockchain must be attacked. The biggest feature of the blockchain is that it does not need a central administrator and distributed storage of data. Because of these characteristics, blockchain is used in industries where reliability and security are guaranteed. Table 1 presents characteristics by classifying blockchain technology by generation. Since 2009, when blockchain was known, blockchain has been widely used in three generations.

**Table 1: Classification of blockchain technology by generation**

| Generation | Characteristics | Period (Year) |
|---|---|---|
| First-generation | .Start with Bitcoin (System)<br>.Public blockchain<br>.Limited application (virtual integration) | 2009 to 2015 |
| Second-generation | .Start with Etherium (platform, smart contract)<br>.Support for a variety of applications based on smart contracts<br>.The emergence of private blockchain<br>.An effort to overcome a thresholds | 2015 to 2018 |
| Third-generation | .Scalability & Security<br>.Support for interoperability between blockchain<br>.Governance | After 2018 |

Like Table 1, blockchain is complementing and improving problems by generation as technologies are developed by generation. In particular, since blockchain became known to the public, the biggest problems it has (transaction speed, processing capacity, interoperability, etc.) are trying to improve performance indicators in various forms in various industries and applications.

### 2.2. *Previous research*

Various studies are underway to address the challenges of cloud edge computing (such as high bandwidth and latency). [8-18]. However, cloud edge computing cannot completely address the multiple services being used in cloud edge computing, as various requirements in different environments are also causing various problems. Recent work has shown the following studies for minimizing latency, resource loss, communication costs, etc. in cloud edge computing.

Foroohifar et al. proposed an approach using self-awareness concepts to ensure the complexity and reliability of data processing in cloud environments. This approach ensures complexity and reliability by distributing complex and consuming machine learning between cloud edges and clouds [8]. However, this approach is a disadvantage of not improving latency and energy consumption between devices operating in cloud edge environments, but only improving complexity and reliability. Kang et al. proposes a lightweight scheduler that minimizes DNN operations between intelligent IoT devices and cloud servers (or data centers) to minimize latency and energy consumption between devices operating in cloud edge environments [9]. This scheduler is characterized by distributing DNN operations between intelligent IoT devices and cloud servers (or data centers) to minimize the cost of DNN computation. However, one of the challenges in cloud edge environments is that latency has not improved significantly compared to existing techniques. Zhao et al. proposed a framework for reducing energy consumption of IoT devices in industrial environments using cloud services [10]. This framework features a three-layer optimization of the framework to minimize latency of IoT networks and energy consumption of IoT devices. However, the disadvantage of this framework is that it is optimized for mobile edge computing environments, and off-road and relay are selected and used. Yang et al. proposes a framework for DL task offload of unmanned aerial vehicles [11]. The framework allows offloading by separating the phase CNN model into upper and lower layers to enable hierarchical DL tasks. Wang et al. proposes an algorithm to minimize the resources of massive amounts of data sent and received in a cloud environment [12]. The algorithm focuses on minimizing the resources of IoT devices, minimizing the loss of resources between local updates and global parameter aggregation. Mills et al. proposed a decentralized optimized Adam format to minimize the cost of communicating devices operating in a cloud environment [13]. However, this approach is characterized by accelerating the integration of distributed datasets and communication costs without performing the process of collaborating with the cloud for runtime reduction. Figurnov et al. proposed a method for deep learning using massive amounts of data sent and received in a cloud environment [14]. This method is characterized by running DNNs dynamically and not activating the entire layer. Furthermore, the method has the characteristic of performing classification first with a simple example before activating the entire layer. Bolukbasi et al. proposed an early DNN model termination method to improve model evaluation time and accuracy by exploiting massive amounts of data sent and received from the cloud [15]. The method is characterized by performing DNN model evaluations using only a fraction of the data, rather than applying the massive amount of data sent and received to the entire layer. Figurnov et al. proposes a DNN layer tuning method using images among the massive amounts of data sent and received [16]. This method is characterized by manually adjusting the number of layers in the image region rather than automatically. Teerapittayanon et al. proposed an architecture in which cloud environments deploy DNNs on edge and end devices at different layers [17]. While this method has the advantage of being able to deploy DNNs in different cloud environments, it has the disadvantage of having to change all models to initialize the model. If all models are not initialized in the model initialization, the prediction accuracy is drastically reduced.

## 3. Optimization Management Techniques for Intelligent IoT Data

### 3.1. *Overview*

Recently, the cloud environment has been supporting services in various forms to provide diverse services to users, and their methods have changed greatly from centralized to distributed. The reason why the cloud environment has changed from centralized to distributed is to analyze the massive amount of data sent and received between IoT and cloud servers (or data centers) while freely expanding the infrastructure that makes up the cloud. Furthermore, the cloud environment uses blockchain to ensure that the massive amount of data sent and received between IoT and cloud servers (or data centers) can be securely protected at low cost.

As cloud services are steadily increasing around IoT devices, the edge cloud environment requires an integrity process of massive amounts of data sent and received between IoT and cloud servers (or data centers). The edge cloud environment seeks to apply blockchain to cloud services by recognizing the importance of the integrity of the massive amounts of data sent and received between IoT and cloud servers (or data centers). However, blockchain-distributed and traceable cloud services are being utilized because they do not guarantee integrity for both large volumes of data sent and received between IoT devices and cloud servers (or data centers).

In this paper, we propose an optimized IoT information management technique that can efficiently manage only IoT information that can be dynamically sent and received from information of massive amounts of data sent and received in edge cloud environments. The proposed technique uses hierarchical distributed polynomials to build links between IoT information while processing IoT information distributedly in edge cloud environments. Furthermore, the proposed techniques link large amounts of dynamically sent and received data to blockchain-based multiple hash chains to ensure the integrity of IoT information. To minimize IoT information loss, the proposed technique classifies IoT information according to the frequency of use of IoT information into n-order sizes and then synchronizes IoT information frequency. The proposed technique can improve IoT information efficiency while lowering the cost of processing IoT information when distributing IoT information in batches so that IoT information is optimized to a hierarchical multi-step structure.

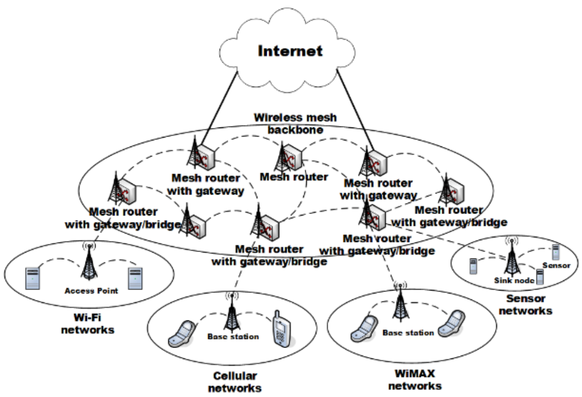


**Figure 1. IoT Component of Proposed Scheme in Clould Environment**

The proposed technique is scalable to a multi-level structure of the n-layer so that information sent and received from heterogeneous IoT devices in distributed cloud environments, such as Figure 1, can have a hierarchical structure. The reason why the proposed technique has a hierarchical distributed path, such as Figure 1, is because of two purposes: First, the proposed technique requires IoT information to send and receive queries of a certain size in order to dynamically send and receive IoT information of this species. Second, the proposed technique can minimize the cost of processing IoT information because it synchronizes IoT information with each other according to the frequency of use of IoT information.

The proposed technique links the path of previous and subsequent blocks of IoT information to blockchain through blockchain-based multiple hash chains to determine only the desired information among the massive amounts of data sent and received in the cloud environment. The proposed technique constructs heterogeneous IoT devices operating in distributed clouds hierarchically and uses the similarity of IoT information blocks stochastically to ensure that the subgroups of hierarchical structures respond flexibly without variation. Furthermore, the proposed technique verifies the hash values stored in the blockchain according to whether to add IoT block information to the blockchain connected to the lower group, ensuring synchronization as well as reliability of IoT information.

### 3.2. IoT Data Gathering Method

IoT information in the proposed technique can be fused at the subnet cluster level to allow local management of IoT information management as well as fast results of IoT information in the IoT information collection process, such as Figure 2. Furthermore, the proposed technique can further deform IoT information in XML, JSON, or RDF form considering edge cloud scalability. In addition, IoT information collected from massive amounts of data sent and received between IoT devices and cloud servers (or data centers) can be analyzed by converging high interoperability JSON into a data connection and conference-oriented RDF format.

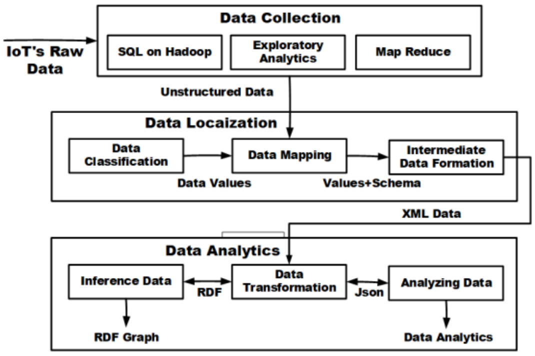


**Figure 2. Process and Analysis for IoT Data Gathering in Clould Environment**

The enormous capacity of IoT information sent and received from heterogeneous devices such as Figure 3 can efficiently manage IoT information by enclosing IoT information into order of polynomials granted to IoT information blocks after being encoded in each packet. Furthermore, the proposed technique can be dynamically operated according to the distributed edge cloud environment situation by classifying it in the form of a polynomial of n-1 order according to the weights of the IoT information without the overlap of the polynomial n-order and IoT information.

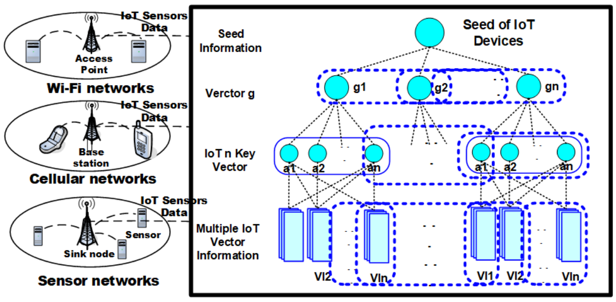


**Figure 3. Routing Information and Weight Processing of Proposed Scheme**

The proposed technique can minimize IoT information processing time by allowing dynamic linkage processing such as Figure 3. Furthermore, the proposed technique improves IoT information throughput by transmitting IoT information transmission interval differences instead of continuously transmitting all IoT information.

### 3.3. Efficient linkage between intelligent IoT information

The proposed technique efficiently links the processing of massive amounts of data sent and received to process IoT information in real time in an edge cloud environment, improving bandwidth and processing latency over existing environments. In addition, the proposed technique improves bandwidth and processing time over existing techniques in the process of sending and receiving large amounts of IoT information in a short period of time when distributing heterogeneous IoT information during IoT information linkage process. This is because the proposed technique dynamically multi-linking massive amounts of data sent and received between IoT devices and cloud servers (or data centers) into blockchain and then placing index transactions of IoT information in a DNN model, enabling rapid processing. Furthermore, the proposed technique minimizes the overhead of the network because it communicates analyzed and predicted IoT information over an overlay network, such as Figure 4, through a DNN model. The proposed technique dynamically limits the computing and memory resources that can arise from IoT edge computing, as IoT devices have different network ranges, such as Figure 4. The proposed technique obtains cumulative probability values by grouping each weight information for IoT groups in n regions into blockchain to collect IoT information as a time axis of the time series. Since the proposed technique weights the index information of IoT information analysis/forecasting through cumulative probability values, IoT information linkage is handled more efficiently than existing techniques.

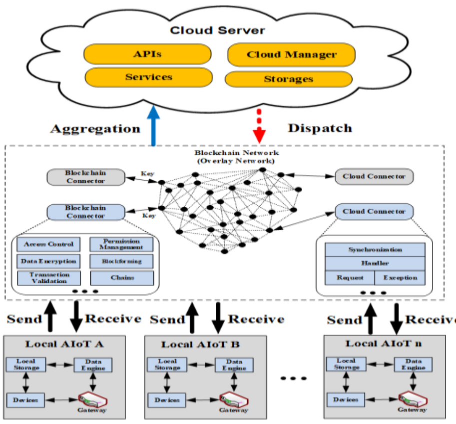


**Figure 4. IoT Information Process between IoT and cloud server**

The proposed technique was structured and processed like Figure 4, which allowed the following functions: First, in regional networks containing IoT devices, transmission and reception information has a distributed network structure to minimize bottlenecks. Second, we minimize the processing delay and network overhead of IoT information by allowing edge computing environments and data centers to be included, and link dynamically extracted information from overlay network environments to blockchain. Third, the collected IoT information can prevent data modulation and illegal access access in advance because the index information and transaction information are organized into blockchain using blockchain-based hashchains.

### 3.4. Processing IoT Information Association Using Time Series Information

The proposed technique allows transactions of IoT information to be grouped into multi-hashchain-based blockchain to link up massive amounts of data sent and received between IoT devices and cloud servers (or data centers). The linkage processing of IoT information using time series information is handled in three stages: IoT information collection stage, IoT information analysis stage, and IoT information delivery stage.

#### 3.4.1 IoT Information Gathering Stages

IoT information collection is selecting seeds to collect sensing information from numerous IoT devices of heterogeneous at high speed in distributed edge cloud environments and then dynamically process the collected information. The seeded information uses index values to improve the efficiency of the linkage between IoT information through analysis/predictions of IoT information. Furthermore, the proposed technique collects the collected information as time series information and then links it to ensure the fast processing and integrity of the IoT information. This is to minimize bottlenecks during the network transmission and reception process.

#### 3.4.2 IoT information analysis stage

The proposed technique performs a three-step analysis of IoT information as follows.

· Step 1: Collect the information of IoT devices distributed in the edge cloud environment in real time, group them into subsets, respectively, in a hierarchical distributed structure, and select seeds.

· Step 2: Obtain the correlation information between property information ($Att_n$ , $Att_m$) among the information grouped by each subset as expression 1.

$$\text{IoT\_Infor} = \{Att_x \mid Att_i \in \text{Att},\ 1 \leq i \leq L\} \tag{1}$$

Where, L stands for the total number of information to be processed by IoT devices.

· Step 3 : We use weights for deep learning-based IoT information to accurately process time series information in a short time using correlations between information-to-information ($Att_n$ , $Att_m$). Since weight information in IoT information is deep-learning IoT information classified in distributed layer environments, we determine the weights with high tension in deep learning as the strength of threshold limits based on results such as access paths or regular expression filtering checks.

### 3.4.3 IoT information delivery stage

The proposed technique performs the IoT information transfer process in two steps according to the reason and absence of IoT information change as follows.

· Step 1: IoT information determined by the strength of the threshold limit based on the results of the access path or regular expression filtering inspection is grouped into a block of blockchain with the index of time series information and delivered to the cloud server.

· Step 2: The cloud server (or data center) stores the information received in the database and checks for the change of IoT information processed by edge computing. If we check the IoT information change status, the cloud server (or data center) informs the IoT of the change and allows the IoT to handle the processing of the change information immediately.

## 4. Analysis

### 4.1. *Environment Setting*

The various parameters used for simulation performed a performance evaluation based on the mean of Monte Carlo simulations. Network coverage radius is set to 300 m and bandwidth is set to 15 MHz/10 MHz. IoT density and server density are assumed to be $10^{-3}/m^2$ and $10^{-5}/m^2$, respectively. Among the blockchain-related parameters, the block size limit is 0.5 to 1 Mbytes, the average transaction size is 100 to 250B, and the block reward rate that the transcoder can obtain is 0.03, and the time limit required to reach the variable reward coefficient is 0.01 s/KB.

### 4.2. Performance Analysis

#### *4.2.1 Process Time*

Table 2 shows analysis results that fast processing is possible because it dynamically multi-links massive amounts of data sent and received between IoT devices and cloud servers (or data centers) to blockchain and then processes index transactions of IoT information as a DNN model. Like Table 2, the proposed technique classifies IoT information into network regions of multi-subset size and then generates cumulative use of transactions via similarity between IoT information, resulting in an average 11.85% improvement in data processing time. These results are from the design of linking seed information in IoT information so that the cumulative use of transactions can be managed in a group at a certain size.

**Table 2: Process time between IoT vs. cloud server**

| Generation | | Number of AIoT | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | | | | 3 | | | | 5 | | | |
| | | Number of IoT Sensor | | | | Number of IoT Sensor | | | | Number of IoT Sensor | | | |
| | | 1 | 5 | 10 | 25 | 1 | 5 | 10 | 25 | 1 | 5 | 10 | 25 |
| Process time (ms) | AIoT Usage | 2.953 | 4.758 | 7.957 | 9.621 | 4.328 | 5.754 | 8.864 | 11.457 | 6.382 | 8.0632 | 9.712 | 12.214 |
| | Not AIoT Usage | 4.321 | 7.351 | 11.022 | 14.201 | 7.285 | 10.142 | 12.327 | 14.658 | 8.963 | 11.201 | 14.321 | 21.365 |

#### *4.2.2 Efficiency*

Table 3 shows how much difference in efficiency differs from cloud servers (or data centers) when intelligent IoT processes massive amounts of data sent and received between IoT devices and cloud servers (or data centers). Like Table 3, the proposed technique achieved an average 14.97 percent improvement in efficiency because it selected seeds to handle data information dynamically, linked them to blockchain, and then treated transactions as DNN models. This result is a result of the proposed technique generating the IoT information similarity between networks managed by multiple subsets as a time series by the cumulative use of transactions.

**Table 2: Process time between IoT vs. cloud server**

| Generation | Number of AIoT | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | | | | 3 | | | | 5 | | | |
| | Number of IoT Sensor | | | | Number of IoT Sensor | | | | Number of IoT Sensor | | | |
| | 1 | 5 | 10 | 25 | 1 | 5 | 10 | 25 | 1 | 5 | 10 | 25 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Efficiency (%) | AIoT Usage | 55.368 | 62.327 | 70.328 | 78.365 | 62.698 | 75.389 | 83.840 | 87.324 | 67.357 | 79.654 | 85.354 | 89.354 |
| | Not AIoT Usage | 63.751 | 70.365 | 78.379 | 84.789 | 68.528 | 79.365 | 87.415 | 89.635 | 70.452 | 83.325 | 88.326 | 91.365 |

## 5. Conclusion

As various IoT devices are used in the cloud environment, many social changes are being made as various information generated by IoT devices is used in various fields. In particular, IoT devices are used a lot, regardless of individuals as well as institutions, resulting in many requirements for IoT data processing. In this paper, we propose an IoT information management technique that allows IoT information to be extended to n-layer multi-level structures to efficiently manage IoT information of different heterogeneous species. The proposed technique consists of hierarchical structures to ensure that information collected from heterogeneous IoT devices is reliable to each other, and lowers the cost of queries locally for different IoT devices through local clouds. As a result of the performance evaluation, IoT processing time results in an average 11.85% improvement in data processing time because IoT processing time classifies IoT information into multi-subset size network regions and then generates cumulative use of transactions through similarity between IoT information. The efficiency of the server resulted in an average 14.97 percent improvement in efficiency, as it was selected to dynamically process data information, linked to blockchain, and then treated transactions as DNN models. In future studies, based on the results of existing studies, we plan to conduct performance comparisons in various cloud environments as the number of IoT devices increases with the proposed techniques to the real environment.

## 6. Acknowledgment

This Research was supported by the Tongmyong University Research Grants 2018A045.

## References

1. A. Whitmore, A. Agarwal & L. Da Xu. (2015). The Internet of Things—A survey of topics and trends. Inf. Syst. Front., 17(2):261-274.
2. N. Golchha. (2015). Big data-the information revolution. Int. J. Adv. Res., 1(12):791-794.
3. M. D. Assuncao, R. N. Calheiros, S. Bianchi, M. A. S. Netto & R. Buyya. (2013). Big data computing and clouds: Challenges, solutions, and future directions.
4. I. A. T. Hashem et al., (2015). The rise of 'big data' on cloud computing: Review and open research issues. Inf. Syst., 47:98-115.
5. C. A. Steed. (2013). Big data visual analytics for exploratory earth system simulation analysis. Comput. Geosci., 61:71-82.
6. Q. Wu, K. He & X. Chen. (2020). Personbalized Federated Learning for Intelligent IoT Applications: A Cloud-edge Based Framework. IEEE Computer Graphics and Applications, PP(99), 1-1.
7. R. Hadidi, J. Cao, M. S. Ryoo & H. Kim. (2019). Robustly executing DNNs in IoT systems using coded distributed computing. Proceedings of ACM/IEEE Design Automation Conference (DAC), 234.
8. F. Forooghifar, A. Aminifar & D. Atienza. (2019). Resource-Aware Distributed Epilepsy Monitoring Using Self-Awareness From Edge to Cloud. IEEE Transactions on Biomedical Circuits and Systems, 13(6), 1338-1350.
9. Y. Kang, J. Hauswald, C. Gao, A. Rovinski, T. N. Mudge, J. Mars & L. Tang. (2017). Neurosurgeon: Collaborative intelligence between the cloud and mobile edge. Proceedings of International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), 615-629.
10. Z. Zhao, R. Zhao, J. Xia, X. Lei, D. Li, C. Yuen & L. Fan. (2020). A Novel Framework of Three-Hierarchical Offloading Optimization for MEC in Industrial IoT Networks. IEEE Transactions on Industrial Informatics (TII), 16(8), 5424-5434.
11. B. Yang, X. Cao, C. Yuen & L. Qian. (2020). Offloading Optimization in Edge Computing for Deep Learning Enabled Target Tracking by Internet-of UAVs. IEEE Internet of Things Journal(Early Access), 1-1.
12. S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He & K. Chan. (2019). Adaptive Federated Learning in Resource Constrained Edge Computing Systems. IEEE Journal of Selected Areas in Communications (JSAC), 37(6), 1205-1221.
13. J. Mills, J. Hu & G. Min. (2020). Communication-Efficient Federated Learning for Wireless Edge Intelligence in IoT. IEEE Internet of Things Journal (IoTJ), 7(7), 5986-5994.

14. M. Figurnov, M. D. Collins, Y. Zhu, L. Zhang, J. Huang, D. P. Vetrov & R. Salakhutdinov. (2017). Spatially adaptive computation time for residual networks. Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 1790-1799.
15. T. Bolukbasi, J. Wang, O. Dekel & V. Saligrama. (2017). Adaptive neural networks for efficient inference. Proceedings of International Conference on Machine Learning (ICML), 527-536.
16. M. Figurnov, M. D. Collins, Y. Zhu, L. Zhang, J. Huang, D. P. Vetrov & R. Salakhutdinov. (2017). Spatially adaptive computation time for residual networks. Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 1790-1799.
17. S. Teerapittayanon, B. McDanel & H. T. Kung. (2017). Distributed Deep Neural Networks Over the Cloud, the Edge and End Devices. Proceedings of International Conference on Distributed Computing Systems (ICDCS), 328-339.