Collaborative Edge-Cloud AI for IoT Driven Secure Healthcare System

Lav Gupta
Department of Computer Science
University of Missouri – St. Louis
St. Louis, MO
lgyn6@umsystem.edu

2023 IEEE International Systems Conference (SysCon) | 978-1-6654-3994-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/SysCon53073.2023.10131082

***Abstract*— In healthcare applications like monitoring patients in ICUs and performing precision robotic surgeries, IoT and sensor networks have become indispensable. These sensors generate a large amount of data that, when processed and visually presented to a medical professional, assists in the more accurate diagnosis and treatment of ailments. For some time now, hospital administrations have been taking advantage of public cloud(referred to as main clouds in this paper) resources to store and process patient data using the advanced AI analytical tools that these clouds provide. However, taking all the medical sensor data to the main cloud encounters network congestion and latencies that may negatively impact the outcomes. In this situation the power of edge-AI may appear appealing, but the state-of-the-art does not allow all the tasks of training complex AI models and drawing inference from them to take place at the edge. Techniques of complexity reduction like pruning and quantization have been applied to reduce storage and processing burden, but they compromise accuracy of the models. Researchers now agree on the necessity of collaborative edge-main cloud AI for demanding workloads.**

It is, however, necessary to realize that the multi-layer IoT-Edge-Main Cloud arrangement has an expanded attack surface. Any malicious attack on the dataflows among various layers may threaten patients' quality of life or even their lives. Although AI can be used to secure these dataflows, using large neural network models centrally on the main cloud results in long training and inference dispersion times. We propose a collaborative, hierarchically merged technique to help train large neural network models in real-time. This is achieved by synthesizing the main cloud model using the trained layers of the edge models, resulting in a dramatic reduction in the training times of the model in the main cloud while achieving high detection accuracy. As we shall see in the description, this method removes some of the problems faced with other collaborative methods, like federated learning, which works by disaggregating models for sharing training load.

Keywords—Edge-Cloud-AI, Collaborative AI, Merged Neural Network Models.

I. INTRODUCTION

Sensor devices are widely used in healthcare; on, in and around patients in hospitals, and those being monitored at offices and at homes. For instance, a robotic surgery arm may contain roughly 28 sensors, and remote patient monitoring for chronic patients may need up to 10 sensors[1]. By 2025, these devices will generate an astounding 75% of all medical data [2]. Cloud computing – in the main cloud or edge cloud or both - is frequently used to store and process large amounts of medical data. The main cloud computing offers benefits related to infrastructure costs, scalability, high utilization, and resilience from server failure. However, taking all this data through a service provider's network for processing in a distant public cloud (also referred to as the main cloud in this work) frequently causes bandwidth, latency and security issues that can negatively impact patient outcomes. When these issues arise in critical medical environments such as emergency rooms and ICUs, we have a recipe for disaster. Processing the data closer to the point of generation, i.e., at the edge, addresses some of these concerns. Edge computing is advancing with more powerful sensing devices and offers faster response times, lower bandwidth costs, and resilience from network failure. The icing on the cake is the advantage of improved security because, presumably, there is safety in numbers. This explains the immense popularity that edge computing has gained in the last few years.

AI gives edge computing the ability to process massive amounts of data and have predictive capabilities to produce actionable intelligence. Bringing AI workflows closer to the data sources provides many advantages for healthcare. For instance, obtaining more precise and quicker results during surgeries. At the same time, AI-powered medical devices and instruments at the edge provide on-demand insights to clinicians, helping them to make crucial decisions about patients more quickly. The power of edge-AI may appear appealing, but this power is limited by computing and storage constraints. These constraints force use of techniques to compress and tailor the DNNs for the edge by methods like pruning and quantization to fit complete AI inference and training on the edge and improve the response times of models. In pruning memory and processing requirements are reduced by removing unimportant weights, synapses, or neurons. Some of the same ends are achieved in quantization by reducing the precision of weights, biases, and activations. However, these methods are prone to loss of accuracy.

It is now believed by researchers and practitioners that transitioning completely to the edge or the main cloud will be difficult. A distributed, collaborative DNN inference among the cloud, edge, and end devices provides a promising way to boost edge intelligence [3], [4]. Federated learning is a distributed framework in which a joint model may reside in the cloud server and the data for training is collected and resides in edge devices. An edge server or cloud server periodically gathers the trained parameters to create and update a better and more

accurate model, which is sent back to the edge devices for local training. The heterogeneity of the models is a challenge in implementing federated learning. The results of several experiments demonstrate that FL with a centralized server is close in terms of performance and reliability to the results with the central approach of the methods used today as training models on a single machine [6].

The paradigm of computing and decision making at the edge, in collaboration with the main clouds, through powerful, results in dataflows crossing trust boundaries among domains in edge-main cloud multi-layered systems that need to be protected against malicious attacks. In this paper, we present our work on a system of hierarchically merged neural networks that protects the integrity of dataflows in collaborative IoT-edge cloud-main cloud AI in the healthcare sector. Specific contributions of this paper are as follows:

i) We consider the tacit convergence of IT and operational technology (OT) to analyze the horizontal and vertical collaborative AI interfaces. This helps us with characterization of attack surfaces, building a schema of metainformation that defines the dataflows, and establishing an architectural framework for the system.

ii) We present an exhaustive empirical investigation of the collaborative, hierarchical merged neural network system working across IoT, edge cloud and main cloud domains in healthcare systems. We show that our system achieves fast training convergence making it suitable for real-time applications. We compare the performance of the merged models with that of baseline unmerged models working across different layers.

iii) We show that the hierarchical merged models, working across edge and main clouds, not only achieve fast training convergence but also achieve high accuracy in detecting integrity violation, which result from malicious attacks on inter-domain dataflows.

iv) We show that as the environment changes, fast training of the collaborative, hierarchical, merged edge-cloud models allow the system to be adaptive.

The rest of the paper has been structured as follows: Section II briefly describes the challenges and motivation for this work. In Section III, we discuss the methodology, including data collection and curation, implementation platform, and facilities used for the research. Section IV covers the state-of-the-art and how this work is different from the other existing works. It also mentions our prior work on which this research is built. In Section V we have the description of the proposed system, and in Section VI we share some important results of our evaluation. Section VII concludes this paper.

II. Challenges and Motivation

It is now understood that the best outcomes in healthcare applications may come from collaborative IoT-edge-cloud AI, which offers progressively more powerful AI analytics and data storage. However, this way of building the system also increases the threat to data integrity between the IoT, edge cloud, and main cloud domains. Deep neural network (DNN) models, running independently at different layers, can potentially be useful but have their own challenges. In large multi-domain environments, these models grow rapidly in size and complexity as we move from the Internet of Things (IoT) gateway to the main cloud. The resulting challenges include achieving fast training of the main cloud models while also achieving high accuracy in detecting known and unknown attacks on the data-in-motion. A thorough review of the literature reveals a dearth of studies on the protection of the integrity of dataflows in collaborative IoT-edge-cloud AI, particularly in healthcare. This provides the motivation for this work. We have worked on a cutting-edge security system based on distributed and hierarchically merged neural networks. The merged system facilitates real-time collaborative training of models across edge gateways, edge cloud servers, and main clouds. At the same time, the accuracy of detection of data integrity violations continues to be on par or superior to the baseline conventional unmerged system of models.

III. Methodology

Our system of distributed and hierarchically merged DNN models works across the IoT, edge, and main cloud layers and predicts attacks that could lead to loss of data integrity resulting in unwanted consequences for patients. The baseline set up involves unmerged models across all the domains. We will discuss below the datasets used and the experimental set-up used. This work builds upon our preliminary work on the merged models for multi-clouds ~~[25]~~[7]. We have evaluated the collaborative merged models for acceleration of convergence and accuracy and compared them with the baseline unmerged models.

A. Data collection

The importance of realistic and representative datasets cannot be overemphasized. When we started our work on merged models, we used datasets that were not specific to the IoT environment. In 2018, UNSW unveiled the BoT-IoT datasets [8], which incorporated many properties relevant to our target environment. Then again, in the year 2020 came the UNSW TON_IoT datasets, which have been generated in the IoT and Industrial IoT environments for evaluating the fidelity and efficiency of different cybersecurity applications based on Artificial Intelligence (AI) ~~[6]~~[9]. Use of sufficient and recent training data gives our system the power to weed out attacks fast and with high accuracy, so as to let the medical staff provide the best patient care without unduly worrying about false positives. A brief description follows below. For additional information, readers are requested to consult the indicated references to obtain more information.

i) BOT-IoT datasets: This dataset was released by UNSW in 2018. It incorporates legitimate and simulated IoT network traffic. The main novelty of the proposed dataset is the introduction of the IoT element in the environment. It consists of flow traffic extracted into about 72 million records with 43 features. The dataset includes normal records and those affected by DDoS, DoS, OS and Service Scan, Keylogging and Data Exfiltration attacks. To improve the performance of our model and make the training of models more efficient we analyzed the features and performed manual as well as automated evaluation through Weka® 3.8.6. Through our analysis, we selected 18 features. This checks out with an elaborate evaluation carried out by Nimbalkar and Kshirsagar in which they selected 16 features using JRip classifier in

Weka. They obtained a higher accuracy and detection rate of 99.9993%, and 99.5798% respectively [7] [10].

ii) TON-IoT datasets: These new generations datasets have been designed for Industry 4.0/Internet of Things (IoT) and Industrial IoT (IIoT) datasets were released in the year 2020. They have been developed in the IIoT, SDN and NFV environment which suits our study. We have used the dataset for Windows 10 that contains 10,000 normal records and 11104 attack records and has 125 features. These datasets have been called 'ToN_IoT' as they include heterogeneous data sources collected from Telemetry datasets of IoT and IIoT sensors.

B. Facilities and equipment

UMSL Networking and Cybersecurity Lab has an Nvidia GeForce RTX 2080B GPU based machine. We also have access to Raspberry PI 4 kits, Ethernet switches and sensors. The Cybersecurity and Information Technology Lab has a virtualized sandboxed ethical hacking and penetration testing environment with intentionally vulnerable machines created through Metasploitable 2 and Kali based attack machines.

C. Implementation platform

The code implementing the unmerged and merged models has been developed in Python on the Anaconda/Spider platform using Keras with TensorFlow backend. Some parts of the code have been ported to MATLAB. The hierarchically merged models with edge clouds and a main cloud have been tested on an 8-core Mac as well as a Windows machines with CPUs and Nvidia GTX 1080/2080 GPUs. Models have also been trained and tested on the Google Colab cloud platform using the latest TensorFlow and Keras releases.

IV. Related work

A lot of work has been done in the areas of IoT/sensor networks, edge, and cloud computing and the application of AI analytics to sensor generated data. To put this work in the right perspective and bring out the significance of the work that we propose to do, we have done an extensive examination of the related work in these areas and of the security issues in multi-layer implementations. We present here a sample of that work, and in the next section we differentiate our work from the state-of-the-art.

A. State-of-the-Art

i) *Edge-AI:* Among the options for meeting the challenge of training and deploying models on resource constrained IoT devices, is distribution of data among trusted peers or to third-party applications for analysis and decision-making. Some researchers believe that the main issues in healthcare systems are late diagnosis, wrong treatment, and misinterpretation [9][11]. Many of these issues can be tackled by improving the edge computing architecture such that processing takes place both at the IoT devices and the edge gateways.

ii) *Collaborative Edge-cloud AI*: Researchers have found that inference on edge devices costs up to two orders of magnitude greater energy and response time than central servers [10] [12]. Again, many researchers are of the opinion that edge devices can only run lightweight AI inference tasks, but training of models is beyond their capacities [11] [13]. They recommend studying how best to use the edge resources to collaboratively enhance the training process. The authors in [4] examine a distributed approach in which simpler cases are executed on the edge and difficult are ones offloaded to the cloud. According to the authors in [12][14], the easiest offloading strategy is to offload the inference task to the cloud when the network conditions are good otherwise, the model is compressed for executing the inference task in the edge layer.

iii) *Federated Learning*: Federated learning is a type of collaborative edge-cloud AI. The researchers in [13][15] present a shared global model in the cloud is learned from multiple nodes at the edges. The authors in [14] [16] apply federated transfer learning to wearable healthcare applications. In [15] [17] the authors suggest co-training of models across distributed clients as centralized training with sharing of training data between edge and main clouds has a security angle. The authors in [16] [18] try to resolve the edge capacity issue by constraining their exploration to the cases in which the edge device capacity meets the requirements of computations.

iv) Security in Collaborative Edge-Cloud AI: The work in [17] [19] outlines many of the challenges faced with federated learning. Aside from the major issues of managing heterogeneous systems on the same network and the statistical heterogeneity of data, there are many security and privacy concerns. It has been pointed out by researchers that federated learning, which was originally intended to protect privacy, is more vulnerable to attacks by malicious nodes than traditional deep learning frameworks [18][20] [27] [21]. If the anonymous clients include attackers that upload malicious data to the server, federated learning may be much less effective than traditional learning algorithms. Also, challenges related to security and complex computations associated with edge-AI in healthcare assume intensified importance [19] [22] [20] [23]. The authors in [14] also conclude that there are still prominent challenges in edge intelligence, including privacy and security issues.

In general, as an extension of cloud and edge computing, cloud-edge collaboration faces many of the same and some new security issues. The latter have been understudied in previous literature. One of the important security issues is the malicious use of APIs that edge devices and cloud servers need to work together. Other issues include design of security features on resource constrained and heterogenous edge devices [24].

B. How This Work is Different

In our work, we have attempted to give a profoundly new direction to the collaborative approach to IoT-edge-cloud AI by introducing an intuitive and yet powerful system of hierarchically merged edge-cloud neural network models for ensuring the security of healthcare related dataflows among the IoT gateways, the edge, and the main cloud gateways. The merged model technique is different from the way federated models are currently built. In many cases, the federated learning framework involves model disaggregation and distribution of the workload among edge nodes or between the cloud and the edge. In a heterogenous situation, exchanging parameters and using them to get a complete model is complicated and prone to loss of accuracy. Our work relies on

the synthesis of the main cloud model from the trained layers of edge cloud models. This has shown good results in terms of training time and accuracy of intrusion detection. Unlike methods that primarily function in a top-down manner [14][16] [28], our method works from the bottom-up. It can also be seen from much existing works that training involves many iterations between cloud and edge, with no guarantee that all the models are optimally trained at any stage and the communication costs may he high [13][15]. Furthermore, distributing large cloud models to edge nodes often necessitates using complex techniques like model compression and model pruning, which not only put a strain on already constrained resources but also negatively impact processing time and accuracy of inference. It has often been claimed that because of the exchange of encrypted data or parameters or stochastic gradient descent updates, the federated system is more secure. We need to realize that encrypted data and parameters are not immune to integrity violations. Any loss of integrity will lead to faulty training and wrong inferences [21] [25]. We have also not seen any concerted study of computational complexity and training convergence times in these models. Finally, we have not come across works that focus specially on the security of dataflows IoT-edge-cloud against unknown attacks.

C. *Our Prior related work*

Some of the prior works that have provided the background for the research described in this paper are briefly described below:

i) Multi-Cloud management and performance evaluation

Multi-cloud management platforms are multi-threaded, distributed and highly complex systems. They need application-specific optimization to perform well and be cost-effective. We have analyzed the performance of these systems mathematically and also empirically on the Washington University in St. Louis multi-cloud testbed and the CloudLab Meta-cloud multi-cloud platform, and the results have been published in [22] [26].

ii) Service Function Chain Placement

In [23][27] we studied the problem of deploying service function chains (SFCs) of network functions in multi-cloud and proposed the P-ART (Predictive-Adaptive Real Time) framework that relies on predictive-deductive features to minimize cost, optimize chosen performance parameters, and increase speed of placement

iii) Machine learning based fault management in multi-cloud virtual network services

We have worked on fault detection and localization models based on a combination of shallow and deep learning structures for virtualized network deployment in clouds. DNNs have been found to be useful for the complex localization function, where a large amount of information needs to be worked through to get to the root cause of the problem [24] [28].

iv) Security in Next Generation Healthcare Systems

Intrusive attacks on next generation healthcare can lead to serious threats to data confidentiality, integrity, and availability. We use deep learners to examine the data being transferred to and from the clouds, to identify maliciously altered communication, data generated by malware, and any other attacks that result in alteration of the value of the flow meta information or the payload [25] [7].

V. EDGE-CLOUD HEALTH SERVICE ARCHITECTURE ABSTRACTION AND SECURITY

The IoT-edge-cloud nexus has its genesis in the constraints that IoT devices have in terms of power, processing, and storage. This has traditionally resulted in pushing analytics and storage from the IoT-domain to the edge and the main clouds. However, more recently, this architecture has lent itself to taking the AI workloads down from the main clouds to the edge and IoT domains to avoid network latencies and bandwidth bottlenecks by processing data closer to the physical world where they are generated. Low latency processing could be used for predictive analysis of data for treating patients in intensive care or critical patients being transported in an ambulance. The cost of connecting edge clouds to the main clouds and the latency introduced in communication between the edge and the main cloud will be comparatively higher. Together, the edge cloud-main cloud combination can be optimized to provide low latency, large storage, optimized bandwidth, and high processing power [26] [29].

As far as the handling of AI workloads is concerned, the architecture shown in Fig. 1 presents many possibilities. Moving AI workloads around, splitting workloads between the main cloud and the edge, exchanging parameters, sending inferences down from the main and edge to the IoT domain are some of the things that can be done. One big downside of this architecture is that, if not taken care of, the healthcare dataflows are rendered vulnerable to a variety of attacks, often with disastrous consequences. Not only are the gateways convenient entry points for attackers, but the connectivity of clouds to the Internet makes it possible for the attackers to damage the integrity of the dataflows through a variety of attacks. Thankfully, this architecture leads us to a distributed security solution for achieving greater agility and a lower cost of healthcare.

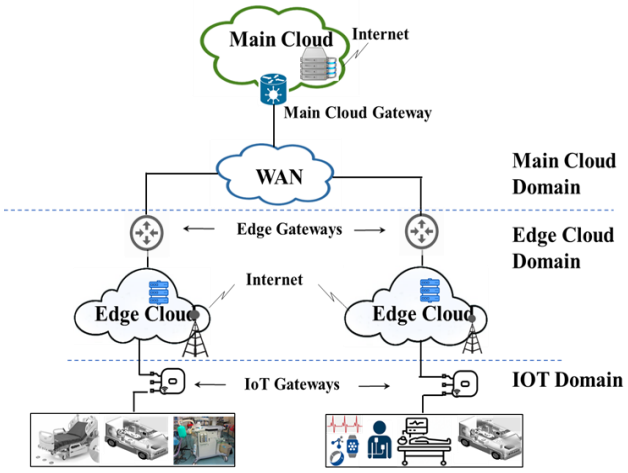

Fig. 1. Domains in IoT-Edge-Cloud based healthcare. Trust boundaries between domains have been marked.

A. *Threat model for edge-cloud AI*

Security by design is the overarching goal of this research. This requires the development of a comprehensive threat model that provides us with an understanding of attack surfaces, attacks and mitigation strategies, taking into cognizance the trust boundaries among layers of multi-cloud systems. The key points are as follows:

Attack surfaces and trust boundaries: Each of the gateways at the trust boundaries has two attack surfaces. For instance, the IoT gateways have one attack surface facing the IoT devices and the other facing the edge clouds. The clouds also face external entities, as they are connected to the Internet. Consequently, they have larger attack surfaces with more porous trust boundaries. A data breach may cause the edge or main cloud processors to act on compromised data, leading to incorrect decisions with potentially serious consequences.

Attackers and attacks: Attackers could be internal or external malicious agents who attempt to mutilate or alter the flow of data or device settings in any way. For instance, they can incapacitate patients' ventilators by launching a distributed denial of service attack or launch an advanced persistent threat attack to remain in the system for a long time to cause widespread damage.

Mitigation: Our primary concern is to protect the data in motion between the IoT, edge, and main cloud domains from attacks. Our strategy has been to create a hierarchical defense that works at all layers of the system. The challenge of real-time training and detection accuracy of models in the main cloud has been tackled.

B. *Security by design-collaborative edge-Cloud AI framework*

The security architecture of the proposed collaborative IoT-Edge Cloud-Main Cloud AI is shown in Fig. 2.

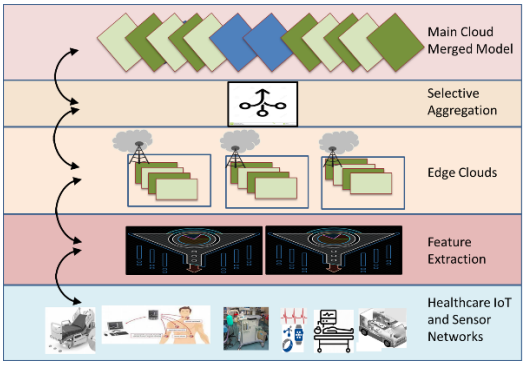

Fig. 2. Deep hierarchical merged models

Synthesis of core cloud models: In conventional federated and non-federated models, the model training times at the main clouds may be unacceptably high for real-time healthcare applications. In our hierarchically merged model design, the relatively small edge models are completely and independently trained with local data. The edge clouds collaborate to synthesize the main cloud models. The merge algorithm merges all or selected trained layers from edge cloud models and produces a composite model for the main cloud. The residual training of the resulting main cloud model takes much less time than a fresh untrained model will take. This is different from transfer learning in which a pre-trained model is used for a different problem. In the merged model The edge cloud models are independently trained on When trained layers from the edge clouds are used to form the larger main cloud model, the merged model so obtained can be trained much faster (almost in real time) while the detection accuracy for loss of integrity in inter-domain dataflows is preserved.

Collaborative security architecture: A diagrammatic representation of the distributed merged structure of the intrusion detection system is in Fig. 3. Extracted features like the packet or byte count, inter-packet times, source, and destination addresses pass through the models in the edge-cloud in the service area in which the IoT equipment falls. As the edge cloud areas are relatively smaller, e.g., a mobile base station area, the training of edge neural network models takes relatively less time compared to the training in the main cloud. The main cloud area is much larger, e.g., a city, a state, a region, or even a full country. The merged model allows aggregation of the selected trained layers of the edge models to be reused at the main model to reduce the training time. This is intuitively different from the conventional method of achieving

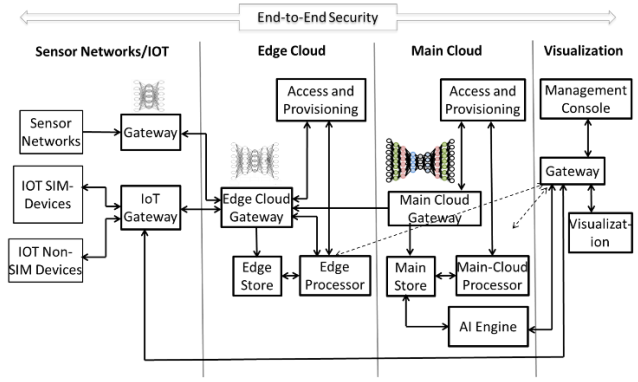

Fig. 3. Collaborative security architecture

model and data parallelisms that works by splitting the model over servers at different levels. With this arrangement, we are able to increase the number of training examples and the total number of parameters that models jointly handle, thereby achieving faster convergence with good accuracy.

VI. Evaluation and Results

Our work on edge-cloud collaborative AI security has produced promising results. As discussed in Section III, we have tested our hierarchical merged model system with the BoT-IoT and the TON-IoT datasets. Following the results of our manual and automated curation with Weka® we have used 18 features from the BoT-IoT datasets and 65 features from the TON-IoT datasets. The number of records taken varies from 4000 to about 10000. These datasets were mutually exclusively subdivided for all the edge clouds and the main cloud, keeping the ratio of train to test as 80:20. We use stochastic gradient based optimization with a loss function based on L2 norm. The large number of weights and biases associated with each SSAE are learnable parameters. The hyperparameters, like the number of layers and neurons per layer are suitably set to obtain good results. Each epoch is a combination of one forward and one back-propagation iteration of the complete dataset.

A. Train and Test results for edge and main cloud models:

Benchmarking of the proposed synthesized model is carried out against traditional, unmerged model.

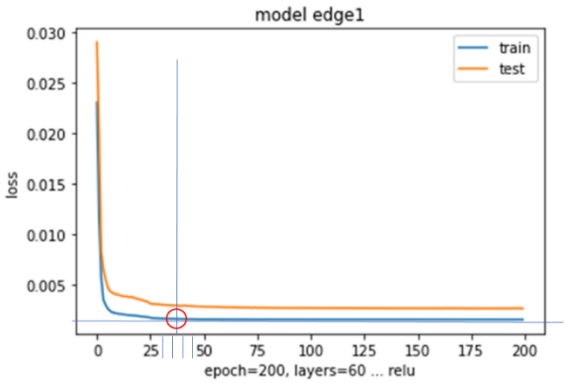

Fig. 4. A(1) Training and testing of edge cloud models (using BoT-IoT datasets)

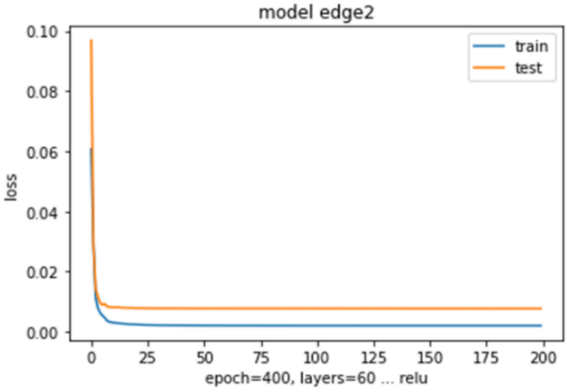

Fig. 4 A(2) Training and testing of edge cloud models (using BoT-IoT datasets)

Unmerged model: Training and test results for two of the edge clouds are shown in Fig. 4 A for the BoT-IoT dataset and in Fig. 4B for the TON-IOT dataset. Extra details have been added to Fig. 4A(1) to show the point in the training process where the losses stabilize. It is seen that the cloud model takes between 35 and 40 epochs to train. We see similar results in Figs. 4B(1) and 4B(2) for the training of the edge models with the TON-IoT datasets.

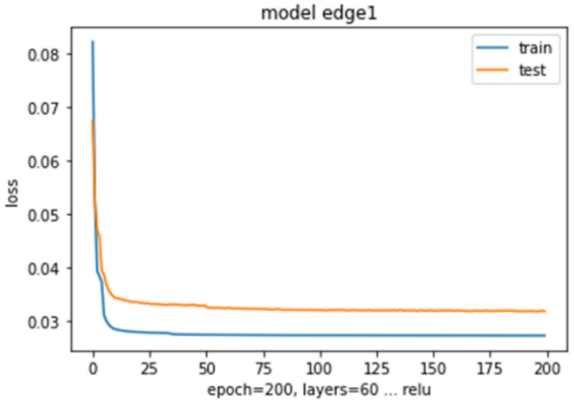

Fig. 4 B(1) Training and testing of edge cloud models (using TON-IoT)

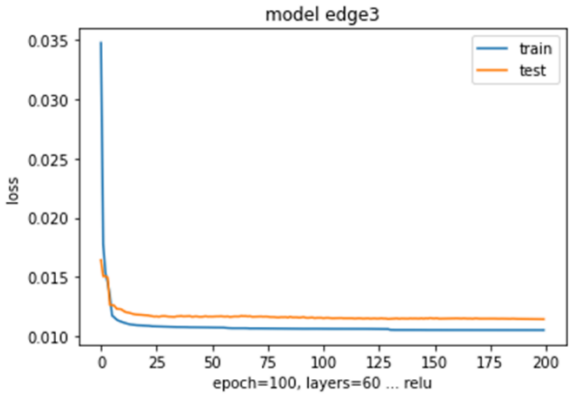

Fig. 4 B(2) Training and testing of edge cloud models (using TON-IoT)

Merged Model: Fig. 5A gives the training and test results for the merged main model. It is seen that for the BoT-IOT, the merged model takes around 10 epochs to train, as compared to between 35 and 40 epochs taken by the edge cloud models. This is a large improvement in the speed of training of the model in the main cloud because of the reuse of layers trained in the edge clouds. Of course, each epoch during training of the main cloud takes longer than each epoch for training of edge clouds. Edge clouds are much smaller than the main cloud, resulting in a smaller number of trainable parameters and the training examples required to train edge-clouds are much less. Fig. 5B shows a similar improvement with the TON-IoT dataset. The overall effect of training the main cloud model with the merged technique is a reduction in the training time.

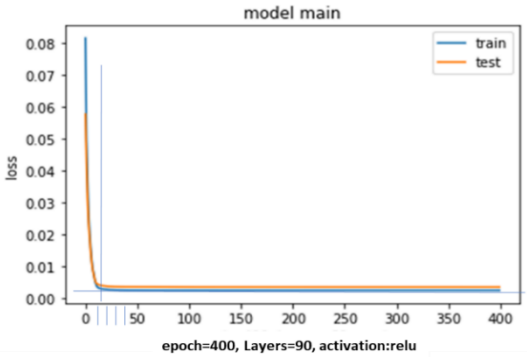

Fig. 5A. Training and testing of main cloud merged model (BoT-IoT)

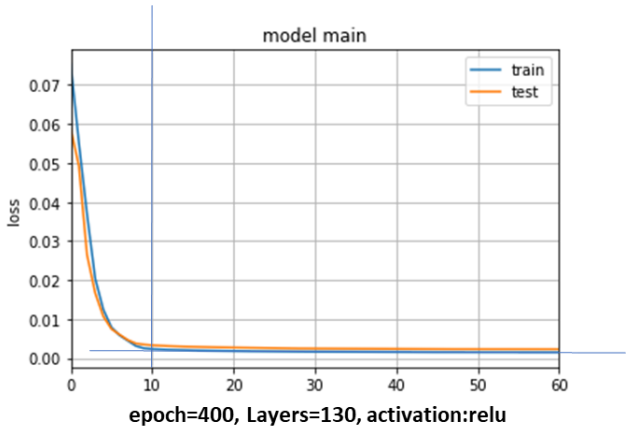

Fig. 5B. Training and testing of main cloud merged model (TON-IOT)

B. *Execution time for merged and unmerged models*

The edge cloud models were constructed with two encoder and two decoder layers. The width of the first layer of the neural network models was varied for testing and was eventually fixed at 60 for the BoT-IoT datasets and 70 for the TON-IOT datasets. For each of the datasets, we performed a large number of training cycles for the edge cloud neural network models and observed an average training time of 21.46 seconds. With the TON-IoT datasets, the edge cloud models take, on average, about 26.06 seconds for training. The main cloud models were larger with 6 to 8 layers with a width of 60-90 for the BoT-IoT datasets and 130 to 260 for the TON-IoT datasets. The average training time in the two cases is 65.74 seconds and 76.31 seconds, respectively. To get an idea of the work involved, in a set of cases, the trainable parameters are 8735 at the edge and 64,025 at the main cloud. Let us now analyze the improvement of the convergence rate of the large neural networks in the main cloud with the hierarchical, merged model technique. We recall that the hierarchical merged models are synthesized using trained layers from the edge clouds. We have seen in Section VI A that while the edge model takes 35-40 epochs to train, the main cloud takes about 10 epochs. Of course, each epoch consisting of forward and backward propagation involves matrix manipulations, and an epoch of the main model takes longer than an epoch of the edge cloud model. In other words, when translated to time, 10 epochs of the main cloud model will take much longer than 35-40 epoch of an edge cloud model. If we compare this with the traditional unmerged technique in which the trained layers of the edge cloud models are not reused at the main cloud, the main cloud models will also take 35-40 epochs, increasing the total training time manyfold. Table I gives a comparison of the time taken in both the cases. We see that training of models with layer reuse takes more than 20% less time than models without layer reuse. In the merged model, many of the trainable parameters have already been trained at the edge. The difference in the training time becomes rapidly large as the model size and depth and the training data dimensionality and size increase.

TABLE I TRAINING TIMES OF MERGED AND UNMERGED MODELS

	No layer reuse	***With layer reuse***
Parameter	***8 layers***	***8 layers***
Trainable	72670	64025
Training Time	85.02 s	67.78 s

C. *Accuracy of training and attack detection performance*

For discrete runs, the model test accuracy follows the train accuracy indicating good training performance (Fig. 6). Other than a few points of deviation, the test accuracy is lower than train accuracy indicating that there is no overfitting which should lead to good generalization over unseen examples.

Fig. 7 charts the rise in accuracy in training of the main models as epochs of training pass. We see that the test accuracy settles close to the training accuracy with a very few deviations. We have seen in our experiments and as discussed below, that this leads to good detection accuracy.

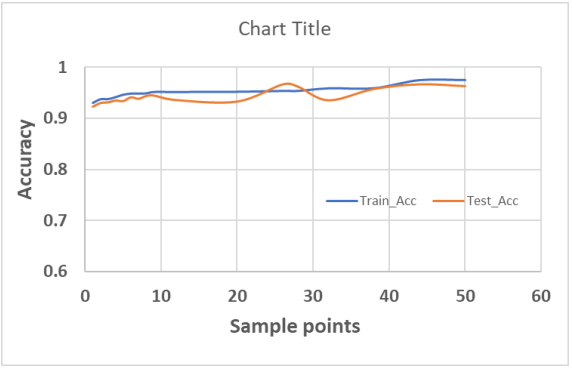

Fig. 6. Sample training and corresponding test accuracy

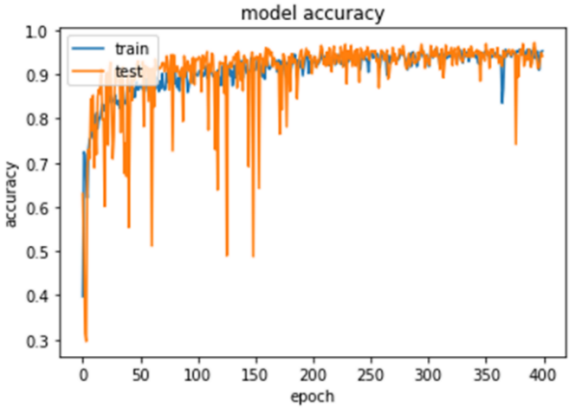

Fig. 7 Main model test accuracy for a complete training cycle (BoT-IoT)

Performance of the system in filtering out attack cases: This step is performed using a mix of unlabeled attack and normal samples from the BOT-IOT datasets. From a large number of random runs with 20 sets of 100 attack vectors each, we get an average false positives rate of 0.4% and accuracy ranging from 98.8% to 99.6%. This is a marked improvement over the baseline accuracy of 92% to 93% of the unmerged model. An important observation with this model is that there are no false negative cases in any of the 20 vector-sets. In other words, all attack cases have been detected as attacks.

VII. CONCLUSIONS

To the best of our knowledge, no investigation is available to ensure the integrity of dataflows in the collaborative IoT-edge cloud-main cloud AI, especially with hierarchically merged DNNs applied to healthcare systems. The system discussed in this paper fills this gap. The proposed hierarchical and merged models work very well with the hierarchical structure of the healthcare network. One common problem with sophisticated deep learning models is their time complexity. In the conventional federated learning systems, repeated exchange of information among the layers increases the threat to the integrity of dataflows and does not help the training convergence time. This has the risk of rendering the system inapplicable to real-time healthcare environments. We use synthesis instead of disaggregation of neural network models and reduce the training time by reusing the intelligence from the IoT gateways in the edge cloud models and from trained edge-cloud models in the main cloud models. This method has exceeded our initial expectations by not only improving

training times drastically but also improving the detection accuracy in comparison with the baseline merged models.

Another important achievement of this work is the testing of adaptive AI features of the merged models. With the changing environment at the physical sensor layer, the models adapt fast to the changing nature of the training data. They maintain a training state that reflects the current situation and are able to filter out cases where the integrity of dataflows among the layers has been damaged. As healthcare systems involve human lives and the quality of life of patients, we plan to continue our work on the adaptability of AI and include features that would help explain the reasons for decisions given by the system.

REFERENCES

[1] A. S. Syed, D. Sierra-Sosa, A. Kumar, and A. Elmaghraby, "IoT in Smart Cities: A Survey of Technologies, Practices and Challenges," Smart Cities, vol. 4, no. 2, 2021, pp. 429–475.

[2] "How Edge Computing is Transforming Healthcare," Vanessa Braunstein, 2021, Web: https://developer.nvidia.com/blog/healthcare-at-the-edge/

[3] "Edge and Cloud Computing Can They Coexist Peacefully?" 2022 Web: https://www.scribd.com/article/577878115/Edge-And-Cloud-Computing-Can-They-Coexist-Peacefully.

[4] A. Banitalebi-Dehkordi, N. Vedula, J. Pei, F. Xia, L. Wang, Y. Zhang, "Auto-Split: A General Framework of Collaborative Edge-Cloud AI," arXiv:2108.13041 [cs.LG], 2021]

[5] W. Ren, Y. Qu, C. Dong, Y. Jing, H. Sun, Q. Wu, S. Guo, "A Survey on Collaborative DNN Inference for Edge Intelligence," arXiv:2207.07812 [cs.DC], 2022

[6] Bao, G., Guo, P. Federated learning in cloud-edge collaborative architecture: key technologies, applications and challenges. *J Cloud Comp* **11**, 94 (2022).

[7] L. Gupta, T. Salman, A. Ghubaish, D. Unal, A.K. Al-Ali, R. Jain, "Cybersecurity of Multi-Cloud Healthcare Systems: A Hierarchical Deep Learning Approach," Elsevier Applied Soft Computing, vol. 118, 2022.

[8] N. Koroniotis, N. Moustafa, E. Sitnikova, B. Turnbull, Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset, Future Gener. Comput. Syst. (2019).

[9] A. Alsaedi, N. Mustafa, Z. Tar, A. Mahmood, A. Anwar, "Ton_IOT Telemetry Dataset: A New Generation dataset of IoT And IIoT for Data-Driven Intrusion Detection Systems," IEEE Access, 2020.

[10] P. Nimbalkar, D. Kshirsagar, "Feature selection for intrusion detection system in Internet-of-Things (IoT)," ICT Express, 2021, pp 177-181.

[11] M.M. Kamruzzaman, I. Alrashdi, A. Alqazzaz, "New Opportunities, Challenges, and Applications of Edge-AI for Connected Healthcare in Internet of Medical Things for Smart Cities," J Healthc Eng. 2022

[12] T. Guo, "Cloud-Based or On-Device: An Empirical Study of Mobile Deep Inference," IEEE International Conference on Cloud Engineering (IC2E), 2018, pp. 184–190

[13] Y. Jin et al., "Edge-Based Collaborative Training System for Artificial Intelligence-of-Things," IEEE Transactions on Industrial Informatics, vol. 18, no. 10, 2022, pp. 7162-7173

[14] D. Xu et al., Edge Intelligence: Architectures, Challenges, and Applications," arXiv:2003.12172v2 [cs.NI], 2020

[15] Q. Wu, X. Chen, Z. Zhou, J. Zhang "FedHome: Cloud-Edge Based Personalized Federated Learning for In-Home Health Monitoring," IEEE Transactions on Mobile Computing, 2022, pp2818-2832.

[16] J. Yang et al., "Security of Federated Learning for Cloud-Edge Intelligence Collaborative Computing," International Journal of Intelligent Systems, 2022

[17] H.G. Abreha, M. Hayajneh, M.A. Serhani, "Federated Learning in Edge Computing: A Systematic Survey," Sensors, 2022.

[18] S. Wang et al., "Adaptive Federated Learning in Resource Constrained Edge Computing Systems," IEEE Journal on Selected Areas in Communications, 2019.

[19] "An Introduction to Federated Learning: Challenges and Applications," Gaudenz Boesch, 2022, Web: https://viso.ai/deep-learning/federated-learning/

[20] K. Zhang et a., "Challenges and Future Directions of Secure Federated Learning: A Survey," Frontiers of Computer Science, 2022

[21] R. Krishnamurthi, D. Gopinathan , A. Nayyar, B. Qureshi, "An Overview of IoT Sensor Data Processing, Fusion, and Analysis Techniques," Sensors, 2020, 23 pages.

[22] X. Li et al., "Edge Care Leveraging Edge Computing for Collaborative Data Management in Mobile Healthcare Systems," IEEE Access, 2019.

[23] S.U. Amin, M.S. Hossain, "Edge Intelligence and Internet of Things in Healthcare: A Survey," IEEE Access, 2021, pp. 45-59.

[24] J. Yang, T.-Y Lee, W.-T., Lee, L.A. Xu, "Design and Application of Municipal Service Platform Based on Cloud-Edge Collaboration for Smart Cities," Sensors 2022.

[25] C. Ma, J. Li, M. Ding, H. H. Yang, F. Shu, T. Q. S. Quek, and H. V. Poor, "On Safeguarding Privacy and Security in the Framework of Federated Learning," IEEE Network, 2020, pp. 1-7.

[26] L. Gupta, R. Jain, M. Samaka, "Analysis of Application Delivery Platform for Software Defined Infrastructures," International Journal of Communication Networks and Distributed Systems, 2016.

[27] L. Gupta, R. Jain, A. Erbad, D. Bhamare, "The P-ART Framework for Placement of Virtual Network Services in a Multi-cloud Environment," Elsevier Computer Communications, 2019, pp. 103-122.

[28] L. Gupta, T. Salman, M. Zolanvari, A. Erbad, R. Jain, "Fault and Performance Management in Multi-Cloud Virtual Network Services Using AI: A Tutorial and A Case Study," Computer Networks, 2019.

[29] R. Krishnamurthi, D. Gopinathan , A. Nayyar, B. Qureshi, "An Overview of IoT Sensor Data Processing, Fusion, and Analysis Techniques," Sensors, 2020, 23 pages.