IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 18 November 2023, accepted 6 December 2023, date of publication 12 December 2023,
date of current version 21 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3341837

RESEARCH ARTICLE

Collabo: A Collaborative Machine Learning Model and Its Application to the Security of Heterogeneous Medical Data in an IoT Network

ZIE EYA EKOLLE, (Member, IEEE), HIDEKI OCHIAI, (Fellow, IEEE), AND RYUJI KOHNO, (Life Fellow, IEEE)
Department of Electrical and Computer Engineering, Yokohama National University, Yokohama 240-8501, Japan

ABSTRACT With the increase in globalization, the degree of electronic communication increases. Such an increase is also experienced by many sectors including the medical sector, which communicates and generates large amounts of data related to the spread of diseases as observed in the case of the COVID-19 pandemic. This has led to the deployment of Internet of Things (IoT) networks in many medical centers. However, one main challenge is how to maintain the security of the data and devices in the network. In this study, we discuss the cybersecurity risk associated with IoT networks used for medical services and provide a solution for protecting medical data and devices using an agent-based approach. Unlike most conventional cybersecurity models that use agents based on deterministic logic or independent learning agents to detect and prevent cybersecurity attacks, we propose a cybersecurity model using a collaborative network of learning agents, called Collabo, that share both mutual and causal values regarding their actions on a common security target. Our experimental results demonstrate the significance of our model over conventional models.

INDEX TERMS Cybersecurity, medical IoT, intelligent agents, multi-agent system, machine learning, deep learning, collaborative learning.

I. INTRODUCTION

The amount of medical data generated from medical devices has surged in recent years owing to the increase in medical attention and digital requirements imposed by the COVID-19 pandemic [1]. Coupled with the increase in electronic communication due to globalization, this increase in digital requirements also increases the risk of security threats to medical devices as well as their generated data.

Several techniques have been used to implement security solutions for medical devices and data [2]. Generally, the conventional approach uses a deterministic process based on a set of logical rules. This approach is used in conventional information and cybersecurity software, particularly for signature-based detection. Another approach is the use of a stochastic process, such as a machine learning process.

With the improvement of learning algorithms in recent years, the latter has gained much attention from research and industry, especially in behavioral detection. In some cases, a hybrid approach consisting of both deterministic and stochastic processes is employed.

One problem with medical data is that they are generated by devices with different properties, whose values may have little or no correlation with each other; that is, they are highly heterogeneous. Observation from a thermometer and from an Electrocardiogram (ECG), for example, may have little or no correlation with each other, and using one to interpret the other may lead to incorrect medical decisions. Managing such unrelated types of data is conventionally performed using non-relational databases such as big data [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Chien-Ming Chen.

In medical big data, data sources mostly originate from medical devices. Once collected from these sources, the data can be processed in batches or streamed. Analytics can be performed on the data to enable decision-making. For remote medical devices, there is a high requirement for their data to be collected and transferred to a centralized server that hosts the big data database, where most of the analytics is performed.

To support the high requirement for the transfer of medical data from one location to another, there is a need for an efficient communication network to interconnect the devices that generate the data. This requirement to interconnect medical devices has led to the use of Internet of Things (IoT) technology in most medical institutions.

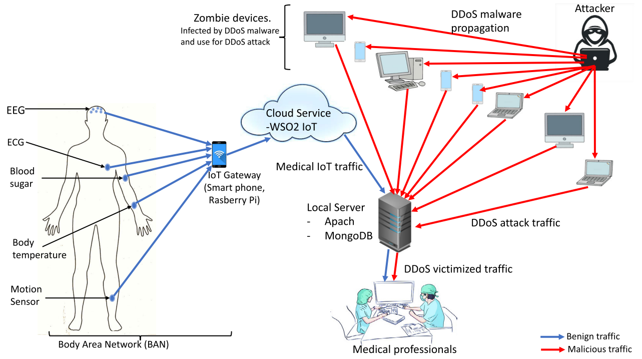

FIGURE 1. Medical IoT network with benign and malicious traffic.

The IoT is a communication technology that enables the interconnectivity of all types of devices. A simple architecture of an IoT network for medical devices is shown in Figure 1 where body sensors, a gateway (e.g mobile phone), cloud, local servers, and personal computers are interconnected. Similar to any other communication technology, IoT technology relies on different communication protocols organized in layers to maintain interconnectivity between devices. These protocols include Extensible Messaging and Presence Protocol (XMPP) and Message Queuing Telemetry Transport (MQTT). A comprehensive list of IoT protocols for medical device networks is provided in [4].

Using network technology such as IoT to connect and transfer data from different devices implies that the security requirement of the data depends not only on the devices but also on the network technology. A vulnerability to network protocols and devices puts the generated data in a vulnerable state, and a threat to the network protocols is a threat to the devices and data. In Table 1, a list of common threat profiles for Medical IoT networks is presented.

Developing a solution to detect and prevent medical IoT attacks is important in medical cybersecurity, and different approaches have been described in the literature [2]. These approaches can be classified as signature-based, behavior-based, or hybrid. Moreover, the solution logic can be deterministic, stochastic, or both.

From the list of security attacks presented in Table 1, we focus on the detection of a Distributed Denial of Service (DDoS) attack on medical IoT devices and networks.

DDoS is a security attack on data availability and integrity. It involves an attacker who uses a command and control system to recruit remote devices in a network by inflicting them with malware and later uses these recruited (or zombie) devices to launch an attack simultaneously on a target device, as illustrated in Figure 1.

A. RELATED WORKS

A DDoS attack on a medical IoT network is an attack on unauthorized access during DDoS malware propagation to recruit zombie devices, and also on availability, during DDoS traffic to the target device using the zombie devices. Thus, it poses a threat to both data and device security. Therefore, an unauthorized access prevention technique for an IoT device is required to achieve device security, whereas an availability-protection technique is required for data security.

In general, most techniques for medical IoT and big data security focus on approaches based on machine learning, statistics, software-defined networks (SDN), and fog computing. This study focuses on the machine learning approach.

Ilhan et al. [5] proposed a detection model of cybersecurity attacks in healthcare systems using recursive feature elimination (RFE) and multilayer perception (MLP). Their model uses logistic regression and extreme gradient boosting models for optimal feature selection. Their proposed model has high performance on different IoT and medical IoT datasets as compared to other models in their work.

Hussain et al. [6] developed a framework for malicious traffic detection in IoT healthcare environments. They first created a framework for IoT data generation called IoT-Flock and then used the framework to generate medical IoT data for machine learning applications in medical environments. The models include Naive Bayes (NB), K-Nearest Neighbors (KNN), Random Forest (RF), Adaboost (AB), Logistic Regression (LR), and Decision Tree (DT) classifiers. Their results validate the use of the models and generated data in IoT healthcare environments.

Khan et al. [7] proposed a model named XSRU-IoMT, for the effective and timely detection of sophisticated attack vectors such as DDoS in internet of medical things (IoMT) networks. The model uses bidirectional simple recurrent units (SRU) to achieve a fast training process in recurrent networks. The proposed model has a higher performance on the TON_IoT dataset as compared to other models.

Zachos et al. [8] proposed an anomaly-based intrusion detection system (IDS) for (IoMT) networks. Their model is focused on both host-based and network-based techniques to collect log files from the IoMT devices and the gateway. They used DT, NB, LR, RF, KNN, and support vector machines (SVM) models in the central detection (CD) component of their proposed anomaly-based IDS. The result

TABLE 1. Common cybersecurity threat profile for medical IoT networks.

Target Assets	Vulnerabilities	Threat Types	Description	Consequences
Medical data	Unencrypted stored data	Ransomware attack	It encrypts files in a device and renders them unusable unless a ransom is paid. After payment, the attacker provides instructions to decrypt the data.	It endangers the integrity of patient information encapsulated in the transmitted data and prevents the availability of such information.
Medical IoT devices and protocols	Weak or no authentication	Backdoor attack	It exploits the weakness of a system to bypass authorized authentication standards. Usually acts as a rootkit.	It bridges authentication protocol and endangers the integrity of patient information.
Medical IoT devices and network traffic	Low memory and computational devices	Denial of service (DoS) attack	It floods the target device with illegitimate traffic to exhaust its computational and memory resources	It prevents the availability of information and exhausts memory and computational resources
Medical IoT devices and network traffic	Low memory and computational devices	Distributed DoS (DDoS) attack	Flood target device with illegitimate traffic to exhaust its computational and memory resources	It excessively exhausts memory and computational resources and prevents the availability of information.
Medical IoT network traffic	Unsecure protocol	Man in the middle attack	Attacker secretly relays and alters, through eavesdropping, the private communication between two parties.	It bridges authentication protocols and endangers the integrity and privacy of patient information encapsulated in the transmitted data
Medical IoT systems	Unsecure protocol	Code injection attack	Involves the exploitation of vulnerabilities in devices, protocols, and software through their bugs, by injecting codes that will upset their manner of operation.	It endangers the integrity of the systems that support the collection, transmission, and visualization of patient data, hence endangering the security of patient data.
Medical IoT systems and password	weak system password	Password attack	Involves the random guess of device or network passwords using different techniques such as brute-force.	It bridges authentication protocols, thereby endangering the security of medical systems and data.
Medical IoT device	weak or no authentication	Scanning attack	It is a search operation carried out on a device or network using different properties such as network port, etc, to identify vulnerabilities.	If done by an attacker, it will lead to a bridge in communication protocols, hence endangering the security of medical systems and data.
Medical IoT software	Unsecure software	Malware attack	It is a malicious code or software that executes unauthorized actions on a system to cause harm or damage.	Depending on its purpose, it can endanger the integrity, privacy, and availability of medical IoT systems and data.
Medical IoT data and system	Unsecure data and system	Phishing attack	It is a technique used to steal sensitive information about a device or user by sending fraudulent messages that appear to come from a legitimate source.	It endangers the integrity and privacy of patient and device data.

shows that the DT, RF, and KNN models are more suitable for their proposed solution.

Kaur and Gupta [9] proposed a DDoS detection model for IoT-based healthcare systems. Their solution, which involves the use of the density peak-based covariance matrix KNN (DPTCM-KNN), is based on the combination of a statistical approach and the KNN machine learning technique. This solution was tested on the Beut and NSL KDD datasets and proved to outperform the SVM in the same process.

Awan et al. [10] proposed a real-time DDoS detection using RF and multi-layer perceptron (MLP) machine learning models. Their solution was implemented with and without big data, but the big-data solution outperformed the non-big-data solution.

Mona et al. [11] proposed a solution based on mutual information and random forest feature importance. They used these methods for feature selection to reduce the DDoS misclassification of different machine learning models. Their results showed that such feature selection methods lead to a higher detection accuracy of DDoS using machine learning models. Maslan et al. [12] proposed a similar feature selection technique that uses a regression model called max-dependency.

Neto et al. [13] proposed a collaborative DDoS detection solution for a general IoT system, including e-health, by using federated learning. They used different deep-learning instances for each pool of data sources to generate the local parameters. To obtain a global result for the overall system, all local parameters were combined and optimized using federated learning techniques. A similar approach was proposed by Popoola et al. [14] for botnet attacks, such as DDoS, on data privacy. Federated learning approaches are on the rise in collaborative IoT security because of their distributed framework; however, the risk of such an approach is enormous, as explained in [15] and [16].

Furthermore, Ferrag et al. [17] present a comprehensive survey with an experimental analysis of federated deep learning approaches for cybersecurity in IoT networks. The main experimental finding in the paper is that federated learning models, specifically federated deep learning models, provide higher IoT device and data privacy than centralized models, and do so with higher detection accuracy on the Bot-IoT, MQTTset, and the TON_IoT dataset.

Finally, Bhayo et al. [18] presented a machine learning-based framework for DDoS attack detection in software-defined IoT (SD-IoT) networks. They built a machine learning framework based on NB, SVM, and DT into the SDN-WISE controller to detect DDoS attacks. Their framework proved to be superior in detection performance than other DDoS detection models used in SDN-WISE controllers.

For a general understanding of the related literature, a number of qualitative properties were used to categorize and compare the different related works in their significance on DDoS attack detection in medical IoT using machine learning. These properties include the use of DDoS attacks, machine learning models, medical datasets, and IoT datasets. This categorical comparison is presented in Table 2.

In this paper, we focus on both data and IoT device security in medical information technology (IT) infrastructure, using a machine learning approach based on collaborative agents that use heterogeneous data sources, defined using different input features. The importance of a collaborative approach to cybersecurity was presented in [19].

Literally, collaboration is the process where two or more entities work together, with the same or different purpose, to complete a common action (e.g., mission or task) or achieve a common target (e.g., vision, goal, objective, outcome, or output). Thus, agents can collaborate on a target or on an action, but since target and action are linked by a purpose, the collaboration on an action entails a collaboration on a target, and vice-versa. In this way, a collaborative action on a target may be divided into multiple partial actions defined by the nature of the target perceived by the agent. In this study, we use classification as the partial action on the target, to achieve a collaborative prediction on the target. However, partial regression and classification can also be used collectively.

The proposed model in this study distinguishes itself from the others in that it considers the environmental diversity and value exchange process of interactive logical agents. The model can be easily scaled for various applications.

B. CONTRIBUTIONS

The contributions of this paper are listed as follows:

- We propose a collaborative learning model based on causal and mutual value exchanges between agents with different input properties (i.e., distributed environments). These values are generated and learned by the agents. This is different from other collaborative learning approaches such as ensemble learning where there is no value exchange between the agents during learning, and federated learning where the exchange value between agents during learning is an accumulation of their parameters by a central agent. Our proposed approach uses a peer-to-peer value exchange mechanism between the agents in distributed environments.
- We propose a specific algorithm to implement the model. The algorithm is based on collaborative prediction and the learning of causal and mutual values. This is completely different from conventional methods because conventional methods are based on the optimization of causal values.

C. ORGANIZATION

The rest of the paper is organized as follows: Section II presents the conventional approach based on deep neural network (DNN). Section III focuses on modeling the proposed approach. These include feature segmentation, agent design, and detection processes. Section IV presents experimental results. This study is concluded in Section V.

II. CONVENTIONAL APPROACH

In this section, we present an approach to solving the medical IoT security problem using deep neural networks.

A deep neural network requires a vector of input properties (or features) to make predictions and training about an output (or target) property.

Consider a dataset denoted by $\mathcal{Q}$, with each element (i.e., instance) $\{Q_1, Q_2, \ldots\}$ in the set $\mathcal{Q}$ consisting of two tuples of input and output, expressed as $Q_i = (X_i, y_i)$, where X_i and y_i represent the ith input instance and the corresponding output instance.

Generally, the prediction operation of such a model is mathematically defined for each instance Q_i of $\mathcal{Q}$ as

$$\hat{y}_i \triangleq f(X_i; \theta) \tag{II.1}$$

where $\hat{y}_i$ is the predicted output value corresponding to the input X_i. Note that $f(X; \theta)$ is a fixed function that represents the prediction operation of the model, where X is an input vector and θ is a set of parameters that define the function.

Next, a learning mechanism is defined for the model to optimize a set of parameters θ such that the output value $\hat{y}_i$ approaches the true output value y_i. Let $\theta^{(0)}$ denote the initial set of the parameters. The training proceeds from $k = 0$ as follows:

$$\hat{y}_i^{(k)} = f(X_i; \theta^{(k)}) \tag{II.2}$$

$$Z_i^{(k)} = \Gamma(y_i, \hat{y}_i^{(k)}) \tag{II.3}$$

$$\theta^{(k+1)} = L(Z_i^{(k)}) \tag{II.4}$$

where y is the true output value, θ is the optimized parameter, $L(Z)$ is an abstraction of the learning operation of the model, Z is the learning value (i.e., the cost value), $\Gamma(y, \hat{y})$ is the function (i.e., the cost function) defining the learning value, and the superscript k in each variable represents the learning epoch.

As a specific example, based on the five input properties in the ECU-IoHT medical dataset [20], a deep neural network model with input property X defined by five input vectors

TABLE 2. Comparison of existing literature on machine learning approaches to DDoS attack in Medical IoT Networks.

References	Year	DDoS attack involved	Machine Learning involved	Medical dataset involved	IoT dataset involved	Main focus/contributions
Ilhan et al. [5]	2023	Yes	Yes	Yes	Yes	Detection of cybersecurity attacks in healthcare systems using RFE and MLP based on LR and XGB.
Hussain et al. [6]	2021	Yes	Yes	Yes	Yes	A framework for malicious traffic detection in IoT healthcare environments using data generated from the IoT-Flock framework and based on NB, KNN, RF, AB, LR, and DT machine learning models.
Khan et al. [7]	2022	Yes	Yes	No	Yes	Detection of sophisticated attack vectors such as DDoS in IoMT networks using bidirectional SRU in RNN.
Zachos et al. [8]	2021	Yes	Yes	No	Yes	Host-based and network-based anomaly IDS for IoMT networks using a combination of DT, NB, LR, RF, KNN, and SVM machine learning models.
Kaur et al. [9]	2022	Yes	Yes	No	Yes	DDoS detection model for IoT-based healthcare system using DPTCM-KNN model.
Awan et al. [10]	2021	Yes	Yes	No	No	Real-time DDoS detection using RF and MLP machine learning models.
Mona et al. [11]	2022	Yes	Yes	No	No	Reduction of DDoS misclassification using a feature selection technique based on mutual information and RF.
Maslan et al. [12]	2020	Yes	Yes	No	No	Reduction of DDoS misclassification using a feature selection technique based on a regression model called max-dependency.
Neto et al. [13]	2022	Yes	Yes	No	Yes	Collaborative DDoS detection for IoT systems using different deep-learning instances based on a federated learning approach.
Popoola et al. [14]	2022	Yes	Yes	No	Yes	Collaborative botnet attack detection for data privacy using different deep-learning instances based on a federated learning approach.
Ferrag et al. [17]	2021	Yes	Yes	No	Yes	A comprehensive survey with experimental analysis of federated deep learning approaches for cybersecurity in IoT networks.
Bhayo et al. [18]	2023	Yes	Yes	No	Yes	A machine learning-based framework for DDoS attack detection in SD-IoT networks.

$(x_1, x_2, x_3, x_4, x_5)$ is described as follows:

$$X = (x_1, x_2, x_3, x_4, x_5) \tag{II.5}$$

$$\hat{y}_i^{(k)} = f(X_i; \theta^{(k)}) \tag{II.6}$$

$$Z_i^{(k)} = \Gamma(y_i, \hat{y}_i^{(k)}) \tag{II.7}$$

$$\theta^{(k+1)} = L(Z_i^{(k)}) \tag{II.8}$$

where y is the true security state of the medical device with respect to a DDoS attack and can take a binary value representing DDoS or Bagnin traffic, x_1 is the time stamp of the packet, x_2 is the source IP address, x_3 is the destination IP address, x_4 is the protocol type, x_5 is the length of the packet frame in bytes, and θ is a vector of parameters.

Figure 2 illustrates the structure of a deep neural network with four layers operating in the medical IoT environment defined by the ECU-IoHT medical dataset.

Different predictive functions can be used for each layer of a deep neural network. This includes the sigmoid function, softmax function, and ReLU function. In addition, the main learning mechanism of a deep neural network is backpropagation, and the learning values (i.e., the cost value) include cross entropy and mean square error.

Conventional deep neural networks are not capable of handling single or multiple output properties collaboratively, which is a limitation on their role as collaborative agents. In this regard, we propose a model that considers a collaborative mechanism, which can lead to more accurate prediction results than the conventional model.

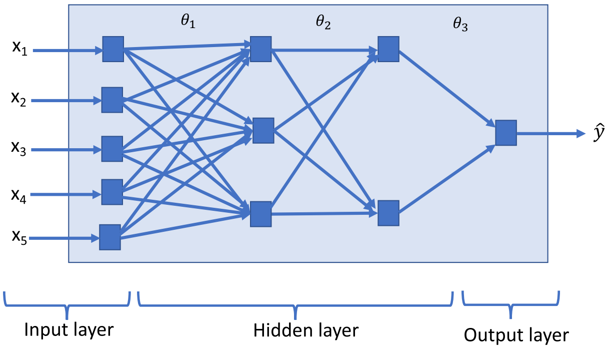

FIGURE 2. DDoS attack detection in an IoT network using deep neural network model.

III. PROPOSED METHOD

In this study, we focused on the use of collaborative learning agents to detect security attacks. Unlike the conventional learning model that uses independent learning agents, we introduce a mutual value exchange between agents to enhance their collaboration regarding the security target.

This approach is intuitive as it is analogous to how humans collaborate on a target; they share knowledge about each other's experiences on the target as they work independently on the target. The problem is defined below.

Consider a medical IoT traffic dataset denoted by $\mathcal{Q}$, with each element (i.e., instance) $\{Q_1, Q_2, \ldots\}$ in the set $\mathcal{Q}$ consisting of two tuples of input and output, expressed as $Q_i = (X_i, y_i)$, where X_i and y_i represent the ith input and the corresponding output instance. The output instance can be Bagnin or DDoS, but not both at the same input instance.

All input instances corresponding to Bagnin output are considered legitimate while those corresponding to DDoS are considered illegitimate. The research problem is to build an agent that can learn to detect if an instance is Bagnin or DDoS. Our proposed solution is depicted in Figure 3.

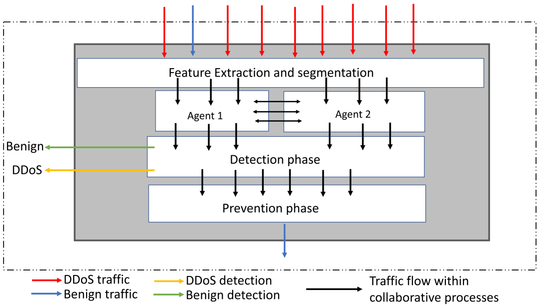

FIGURE 3. **Detection and prevention of DDoS attack in network traffic.**

The proposed solution involves the following four steps: 1) feature extraction and segmentation of incoming network traffic, 2) agent design, 3) intrusion detection, and 4) intrusion prevention. These steps are illustrated in Figure 3. In this study, we focus only on the first three steps. The proposed solution can be installed on a front-end server as a network-based security system for medical IoT edge (i.e., fog) networks.

A. FEATURE EXTRACTION AND SEGMENTATION

In the feature extraction and segmentation step, each instance of the input properties from traffic data X is extracted and segmented into two (or multiple) sub-vectors X_a and X_b, creating a data subset $Q_a = (X_a, y)$ and $Q_b = (X_b, y)$, respectively. Each sub-vector is considered as the input property of the security target y for a given agent.

If the extracted data are those provided in the ECU-IoHT medical dataset [20] described in Section II, we arbitrarily segment the five properties into two groups, as follows:

$$X = (X_a, X_b) \tag{III.1}$$

where $X_a = (x_1, x_2, x_3)$ and $X_b = (x_4, x_5)$.

The output property y defines the state of the target; in this case, it corresponds to the security state of a medical IoT device. This can include the state of DDoS or other related attacks on the same or different devices.

In the case of the ECU-IoHT medical dataset, we consider y as the true security state of the medical device with regard to a DDoS attack and can take a binary value representing DDoS or Bagnin traffic. Therefore, this output property is dedicated to each agent during the collaborative process.

B. AGENT DESIGN

We consider an agent as an entity that seeks a target. It takes actions (or beliefs) on the target. The actions taken by the agent can be causal or non-causal. In this study, causal action is considered as a conditional action [21], [22] while non-causal action is considered as a mutual action [23].

Consider a causal and mutual action defined using probabilistic functions. Given a vector of input property $X_{l,i}$ and an output y_i in a dataset $Q_{l,i}$, where $l \in \{a, b\}$ represents the index of a data subset or the agent, we can define the causal and mutual action instances of agent l in a collaborative network as follows:

$$\hat{y}_{c,l,i} \triangleq P(y_i|X_{l,i}; \theta_{c,l}) \quad \text{(causal action)} \tag{III.2}$$

$$\hat{y}_{m,l,i} \triangleq \frac{P(y_i|X_{l,i}; \theta_{m,l})}{P(y_i)} \quad \text{(mutual action)} \tag{III.3}$$

where y_i is an instance of the target (or output) property, $X_{l,i}$ is an instance of the vector of input properties of agent l, $\hat{y}_{c,l,i}$ is an instance of the causal action of agent l on the target, $\hat{y}_{m,l,i}$ is an instance of the mutual action of agent l on the target, $\theta_{c,l}$ is the parameter of agent l that enables the causal action, $\theta_{m,l}$ is the parameter of agent l that enables the mutual action, and $P(.)$ denotes a probability function.

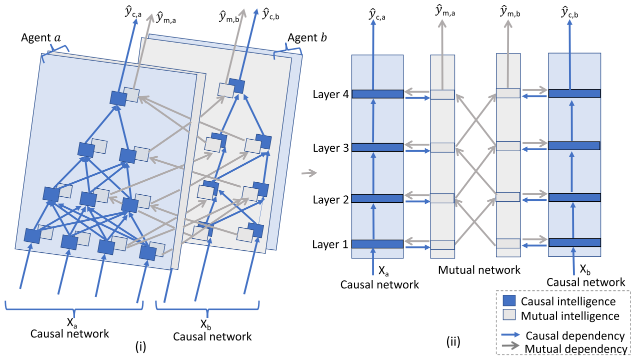

FIGURE 4. **A causal and mutual relationship flow in the proposed multi-agent collaborative learning model. (a) Transversal view (b) Lateral view.**

Consider Figure 4, which represents a collaborative network of two neural network agents, a and b. Each agent takes a sequence of causal and mutual action pairs arranged in layers. Each causal action of an agent depends on those in the agent's previous layer, and each mutual action depends on those in the agent's previous layer.

Layers 1, 2, 3, and 4 represent pairs of causal and mutual actions for each agent. Layer 1 is the input layer, layers 2 and 3 are the hidden layers, and layer 4 is the output layer. The outputs of each layer are the causal and mutual values that propagate in the collaborative network in a particular order.

The relationship that defines the collaborative action of the two agents on the target y, and their causal and mutual actions as shown in Figure 4 are represented as the joint causal actions $\hat{y}_{c,ab}$ of the two agents on the target y based on their partial causal action $\hat{y}_{c,a}$ and $\hat{y}_{c,b}$, while interchanging their mutual actions $\hat{y}_{m,b}$ and $\hat{y}_{m,b}$. The following axiom and proposition (proven in Appendix A) define the collaborative action:

Axiom 3.1: (Conditional Independence of Input Properties)

$$P(X_a|y, X_b) = P(X_a|y) \text{ and } P(X_b|y, X_a) = P(X_b|y)$$

Proposition 3.1:

$$\begin{aligned}\hat{y}_{c,ab} &= P(y|X_a, X_b; \theta_{c,a}, \theta_{c,b}, \theta_{m,a}, \theta_{m,b}) \\ &= P(y|X_a; \theta_a)P(y|X_b; \theta_{c,b})\left[\frac{P(X_a|X_b; \theta'_b)}{P(X_a)}\right]^{-1}\frac{1}{P(y)} \quad \text{(III.4)}\end{aligned}$$

where $P(y|X_a; \theta_a) = \hat{y}_{c,a}$ and $P(y|X_b; \theta_b) = \hat{y}_{c,b}$ are partial causal actions of agents a and b, $\frac{P(X_b|X_a;\theta'_a)}{P(X_b)} = \hat{y}_{m,a}$ and $\frac{P(X_a|X_b;\theta'_b)}{P(X_a)} = \hat{y}_{m,b}$ are mutual actions of agents a and b, and $P(y) = \hat{y}$ is a prior collaborative action of the agents.

For mutual collaboration to exist between the agents, we defined a mutual equivalence property given as

$$\frac{P(X_a|X_b; \theta_{m,b})}{P(X_a)} = \frac{P(X_b|X_a; \theta_{m,a})}{P(X_b)} \quad \text{(III.5)}$$

Using this collaborative framework, it is easy to observe that if the causal action of one agent (i.e., partial causal action) is independent of the causal action of the other agent, this will lead to an ensemble learning model [24] when they share segmented input instances rather than input properties. In addition, the same situation leads to a federated learning model [25] when they share their parameters on a vertically or horizontally decentralized dataset.

1) PREDICTION ACTION

According to (III.4), the joint causal action $\hat{y}_{c,ab}$ of the collaborative agents defines the collaborative prediction of the agents and is a combination of their respective causal and mutual actions. This can be expressed as

$$\hat{y}_{c,ab} = \frac{\hat{y}_{c,a}\hat{y}_{c,b}}{\hat{y}}\frac{1}{\hat{y}_{m,b}} \quad \text{(III.6)}$$

The collaborative action $\hat{y}_{c,ab}$ is optimized by simultaneously optimizing the causal and mutual actions of each agent as described in Section III-B2, and implemented using Algorithm 1. In addition, $\hat{y}$ and $\hat{y}_{m,b}$ are respectively used as the regularizer and normalizer of the partial causal predictions.

Furthermore, $\hat{y}$ can also be used to define the global cost of the collaboration during learning. This may not be necessary because our approach is based on partial learning to achieve a joint value rather than directly learning the joint value.

In fact, $\hat{y}$ can be omitted and $\hat{y}_{c,ab}$ can be predicted and learned using the following approximation of (III.6):

$$\hat{y}_{c,ab} = \hat{y}_{c,a}\hat{y}_{c,b}\frac{n}{\hat{y}_{m,b}} \quad \text{(III.7)}$$

where n corresponds to the number of agents in collaboration and $n = 2$ in our example model.

The value of n influences the complexity, flexibility, and uncertainty in the collaborative network. The analysis of these properties is reserved as future work.

2) LEARNING ACTION DEFINITION

The learning process of agents in the collaborative network presented in Figure 4 involves the simultaneous optimization of their causal and mutual values about the target y by learning the parameters $\theta_{c,l}$ and $\theta_{m,l}$ of each agent $l \in \{a, b\}$. If the learning process is closed in each agent, it follows that

$$\hat{y}^{(k_c)}_{c,l,i} = P(y^{(k_c)}_i|X_{l,i}; \theta^{(k_c)}_{c,l}) \quad \text{(causal action)} \quad \text{(III.8)}$$

$$\hat{y}^{(k_m)}_{m,l,i} = \frac{P(y^{(k_m)}_i|X_{l,i}; \theta^{(k_m)}_{m,l})}{P(y^{(k_m)}_i)} \quad \text{(mutual action)} \quad \text{(III.9)}$$

$$Z^{(k_c)}_{c,l} = \Gamma_c(y_i, \hat{y}^{(k_c)}_{c,l,i}) \quad \text{(causal learning value)} \quad \text{(III.10)}$$

$$Z^{(k_m)}_{m,l} = \Gamma_m(y_i, \hat{y}^{(k_m)}_{m,l,i}) \quad \text{(mutual learning value)} \quad \text{(III.11)}$$

$$\theta^{(k_c+1)}_{c,l} = L_c(Z^{(k_c)}_{c,l}), \quad \theta^{(k_m+1)}_{c,l} = L_m(Z^{(k_m)}_{m,l}) \quad \text{(III.12)}$$

where $L_c(Z_{c,l})$ and $L_m(Z_{m,l})$ are the abstractions of the causal and mutual learning models, $Z_{c,l}$ and $Z_{m,l}$ are the causal and mutual learning values, $\Gamma_c(y, \hat{y}_{c,l})$ and $\Gamma_m(y, \hat{y}_{m,l})$ are the functions of the causal and mutual learning values, and k_c and k_m are the causal and mutual learning epochs, respectively.

Considering a backpropagation learning mechanism based on gradient descent optimization of the learning value, the learning process for each agent is defined as

$$\theta^{k_c+1}_{c,l} = \theta^{k_c}_{c,l} - \eta_{c,l}\frac{\partial Z_{c,l}}{\partial \theta^{k_c}_{c,l}} \quad \text{(causal learning)} \quad \text{(III.13)}$$

$$\theta^{k_m+1}_{m,l} = \theta^{k_m}_{m,l} - \eta_{m,l}\frac{\partial Z_{m,l}}{\partial \theta^{k_m}_{m,l}} \quad \text{(mutual learning)} \quad \text{(III.14)}$$

where $\eta_{c,l}$ and $\eta_{m,l}$ are the causal and mutual learning rates.

C. DETECTION PROCESS

The cyber attack detection process entails both causal and mutual actions in the learning and prediction processes. The algorithm for the detection process is as follows:

Algorithm 1 Collaborative Batch Learning and prediction

```
Require: $\hat{y}_c, \hat{y}_m, \theta_c, \theta_m, X_l, k_c, k_m, Z_c, Z_m$.
Ensure: $min(Z_c), min(Z_m)$
  $k_c \leftarrow k_m$                          ▷ Initializing the learning epoch
  $n \leftarrow num(l)$                  ▷ number of collaborative causal agents
  for $\tau = 1$ to $k_c$ do                                  ▷ learning cycle
      $\hat{y}^{(\tau)}_c \leftarrow P(y^{(\tau)}|X_l; \theta^{(\tau)}_c)$
      $\hat{y}^{(\tau)}_m \leftarrow \frac{P(y^{(\tau)}|X_l;\theta^{(\tau)}_m)}{P(y^{(\tau)})}$
      $Z^{(\tau)}_c, Z^{(\tau)}_m \leftarrow \Gamma_c(y, \hat{y}^{(\tau)}_c), \Gamma_m(y, \hat{y}^{(\tau)}_m))$
      $\theta_c, \theta_m \leftarrow L_c(Z^{(\tau)}_c), L_m(Z^{(\tau)}_m)$
  end for
  $\hat{y}_{c,ab} \leftarrow \hat{y}_c \odot n(\hat{y}_m)^{-1}$              ▷ collaborative prediction
```

where $\odot$ represents element-wise multiplication of two vectors, k_c is the causal learning epoch, k_m is the mutual learning epoch, and τ is the collaborative learning epoch.

Different learning techniques can be used for causal and mutual learning. In addition, apart from the *minmin* synchronized optimization used in Algorithm 1, the optimization can also be a *maxmax*, *maxmin* or *minmax* optimization depending on the target and learning value of the agents.

It should be noted that this learning technique can also be used to explain other multi-agent machine learning models such as the Generative Adversarial Networks (GAN) model [26], which consists of a set of sequentially synchronized learning agents.

D. PERFORMANCE EVALUATION

We present different types of performance metrics that can be used to evaluate the proposed model. We group these into three categories: predictive performance, learning performance, and system performance.

1) PREDICTIVE PERFORMANCE

We consider the predictive performance to be the performance of the agents in taking a prediction action on the target compared to that of the correct action on the target, for single or multiple observation instances. Prediction performance measures can be distinguished for both discrete and continuous target output. These are mostly statistical measures.

For a discrete target output, the measures include accuracy, precision, recall, F1-score, and Area under the receiver operating characteristic curve (AUC ROC), which can all be evaluated using the confusion matrix. For a continuous target output, the performance measures include the mean square error, mean absolute error, root mean square error, and R^2 coefficient of determination. The prediction performance measures are discussed in more detail in [27].

2) LEARNING PERFORMANCE

We consider the learning performance as the predictive value with respect to the true value that the collaborative agents have optimized over the learning epoch. This value can be measured using any learning value (i.e., the cost value), such as root mean square error, cross-entropy, and Kullback-Liebler (KL) divergence over a learning epoch. This enables us to capture the generalization error of the model, which is a combination of the bias and variance errors.

In this study, we used cross entropy as the learning value for both causal and mutual actions.

$$\text{Cross Entropy} = -\sum_{i=1}^{n} P(y_i) \log P(\hat{y}_i) \tag{III.15}$$

where n is the number of classes, y is the true label, $\hat{y}$ is the predicted label, and $P(.)$ is the probability of a label.

3) SYSTEM PERFORMANCE

We consider the system performance as the implementation performance of the model as a system. This is related to the computational complexity performance of the model. In this category are measures such as the Big O notation, floating-point operations (FLOPs), floating-point operations per second (FLOPS), multiply-accumulate operations (MACs), latency, throughput, response time, robustness, stability, memory operations, and energy usage of the model as a system.

Apart from performance evaluation measures, other evaluation measures such as statistical significance, confidence interval, sensitivity analysis, and model comparison measures can also be used to evaluate the model. Some of these evaluation measures used in this study are the Akaike Information Criterion (AIC) for model comparison and the 2σ uncertainty measure for sensitivity analysis.

IV. EXPERIMENT AND RESULT

The aim of the experiment is to demonstrate the performance of our proposed collaborative learning model in detecting DDoS attacks in medical IoT networks and compare the performance results with conventional models. To validate the generalization of our proposed model in different environments, we perform a simulation on different datasets and compare the results with those of other models.

A. EXPERIMENTAL SETUP

The dataset for the experiment is based on the ECU-IoHT dataset [20], which has been used for analyzing cyberattacks on the Internet of Health Things [28]. The Intensive Care Unit (ICU) dataset [6] created using the IoT-Flock tool [29] and the ToN_IoT dataset [30] were used to validate the generalization of our model in different environments.

1) DATASET AND FEATURE PRESENTATION

The ECU-IoHT dataset [20] has eight fields: time stamp of packet, source address of packet, destination address of packet, network protocol of packet, length of packet frame in bytes, information of packet, network status, and attack type.

The type (or network status) field and type of attack field represent the attack labels of the dataset. The network status field was used as the label. Regarding the input features, we excluded the packet information field because it contains packet information reports that may not be helpful in attack detection. Therefore, we used five input features to train the model. Table 3 presents statistics of the dataset.

TABLE 3. List of attacks and input fields used in the ECU-IoHT dataset.

Attack types	Instances	Input fields used
ARP Spoofing	2359	5
DoS	639	5
Nmap Port scan	6836	5
Smurf Attack	77920	5
No Attack	23454	5
Total instance	111207	

Denial of Service (DoS), and Address Resolution Protocol (ARP).

A DDoS attack constitutes both a DoS attack and a port scanning attack. In addition, a Smurf attack is a type of DDoS attack that uses DDoS.smurf malware to render network service inoperable by flooding Internet Control Message Protocol (ICMP) packets to targeted network devices. In this regard, we used DoS, Nmap port scanning, and Smurf attack instances in this experiment.

Therefore, 85395 attack instances and 23454 no-attack instances were used during this experiment, resulting in a total of 108849 attack instances.

Related to the ICU [6] and ToN_IoT [30] datasets, their contents are described in Tables 4 and 5, respectively.

TABLE 4. List of classes and input fields used in the ICU dataset.

Dataset classes	Instances	Input fields used
Attack	80126	50
Environment monitoring	31758	50
Patient monitoring	76810	50
Total instance	188694	

Each class represents a separate csv file which we merge into one file.

TABLE 5. List of attacks and input fields used in ToN_IoT (Network) dataset.

Attack types	Instances	Input fields used
Backdoor	20000	43
DDoS	20000	43
DoS	20000	43
Injection	20000	43
Man in the middle(mitm)	1043	43
Password	20000	43
Ransomware	20000	43
Scanning	20000	6
Cross site scripting(xss)	20000	43
Normal	300000	43
Total instance	461043	

The ToN_IoT file used is the Train_Test Network dataset.

2) DATASET PRE-PROCESSING

The first step in pre-processing is to encode categorical information in the dataset. The label encoding technique was used to encode the target output y, whereas One-Hot encoding was used to encode the source IP, destination IP, and network protocol fields.

Next, normalization operations were performed on the dataset using *minmax scaling*.

$$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{IV.1}$$

where X is a field in the dataset.

The 108849 attack instances of the dataset were split into training and testing instances, as shown in Table 6.

TABLE 6. Partition of attack instances into training and test instances.

Attacks	Attack instances	Training instances (70%)	Testing instances (30%)
DoS	639	447	192
Nmap Port scan	6836	4785	2051
Smurf Attack	77920	54544	23376
No Attack	23454	16418	7036
Total instance	108849	76194	32655

The five input fields are divided into two groups. One group consists of the time stamp and the length of the packet fields, whereas the other group consists of the remaining fields. Each collaborative agent was assigned one group of input fields.

We also split the ICU and ToN_IoT datasets into 70% training instances and 30% test instances. One-Hot encoding was used to encode categorical data, while label encoding was used to encode the labels.

3) AGENT MODELS AND PARAMETERS

Table 7 presents the simulation parameters of the models used in this experiment.

TABLE 7. Simulation parameters of our model and conventional models.

	DNN	Ensemble	Federated	Our model
# of causal net.	1	2	2	2
# of mutual net.	0	0	0	2
# of layers per net.	4	4	4	4, 4
Nodes per layer per causal net.	5,3,2,1	net1:5,3,2,1; net2:5,3,2,1	net1:3,3,2,1; net2:2,3,2,1	net1:3,3,2,1; net2:2,3,2,1
Nodes per layer per mutual net.				net1:3,3,2,1; net2:2,3,2,1
Node type	Sigmoid	Sigmoid	Sigmoid	Sigmoid
Learning algorithm	GD	GD,Bagging	GD,FedAvg	GD,GS
Learning rate	0.61	0.61	0.61	0.61
# of learning cycle	300	300	300	300
Learning value	CE	CE	CE	CE

Deep Neural Network (DNN), Gradient Descent (GD), Gibbs Sampling (GS), Cross Entropy (CE), and # indicates "number".

These parameter settings are for the ECU-IoHT dataset. For the other datasets, only the number of input nodes was changed with respect to their number of features whereas the remaining parameters and hyperparameters were unchanged.

For the ICU dataset, the following number of input nodes per network was used: (50) for DNN, (50, 50) for Ensemble, (25, 25) for Federated, and (25, 25, 25, 25) for our model. For the ToN_IoT datasets, the following number of input nodes per network was used: (43) for DNN, (43, 43) for Ensemble, (20, 23) for Federated, and (20, 23, 20, 23) for our model.

Also, the hyperparameters such as the learning rate, learning cycle (i.e., epoch), the number of layers per network, and the number of nodes per layer, were selected without rigorous hyperparameter turning (optimization) technique. We will investigate hyperparameter training in future work.

B. RESULTS AND DISCUSSION

1) ECU-IOHT DATASET RESULTS

The learning and the receiver operating characteristic (ROC) curves for the ECU-IoHT datasets for each model in Table 7 are presented in Figures 5, 6 and 7.

Figure 5 illustrates the learning curves of our collaborative agent model compared with the conventional models. From the learning curve, the proposed model reduces the normalized cost value to a lower point of stability than the others. The corresponding accuracies during learning are shown in Figure 6. The proposed model turns out to learn more accurately than the other models on the dataset.

However, to confirm the degree of generalization on the dataset, we also performed a prediction performance

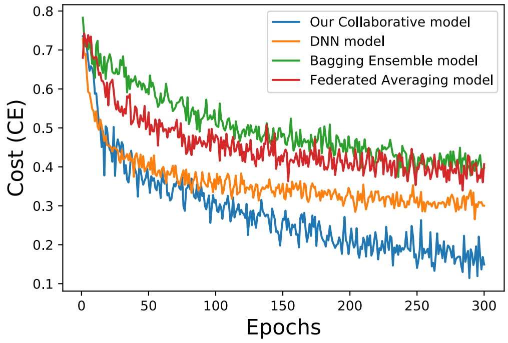

FIGURE 5. Cost-based learning curve of our model on DDoS attack detection compared to other models using ECU-IoHT training dataset.

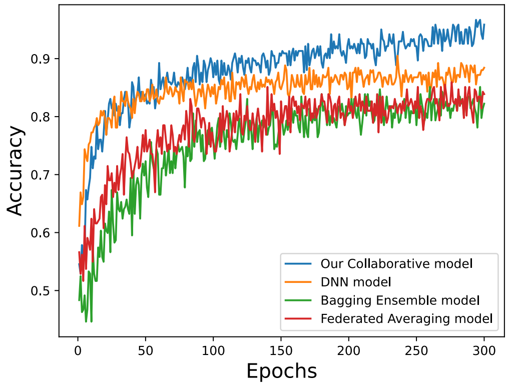

FIGURE 6. Accuracy-based learning curve of our model on DDoS attack detection compared to other models using ECU-IoHT training dataset.

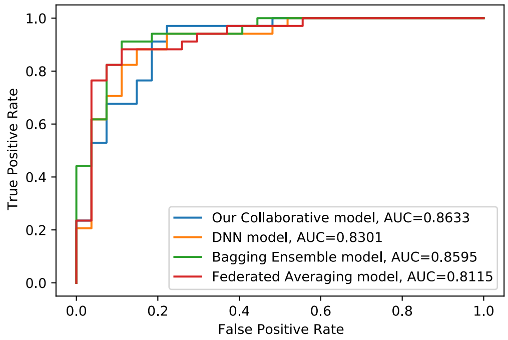

FIGURE 7. ROC curve of our model on DDoS attack detection compared to other models using ECU-IoHT test dataset.

measurement using the ROC function. The results are presented in Figure 7, where we observe that our model has a higher AUC ROC than the conventional models and hence can distinguish between target classes more accurately than the other models.

2) ICU AND TON_IOT DATASETS RESULTS

The learning and ROC curves obtained by the ICU datasets are shown in Figures 8, 9 and 10, whereas those by the ToN_IoT datasets are shown in Figures 11, 12 and 13, all evaluated for each model in Table 7. For both datasets, we observe that our model has a higher AUC, accuracy, and a lower normalized cost value than the conventional models and generalizes on the dataset better than the other models.

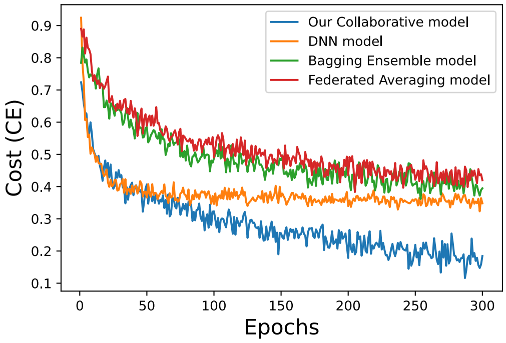

FIGURE 8. Cost-based learning curve of our model on DDoS attack detection compared to other models using ICU training dataset.

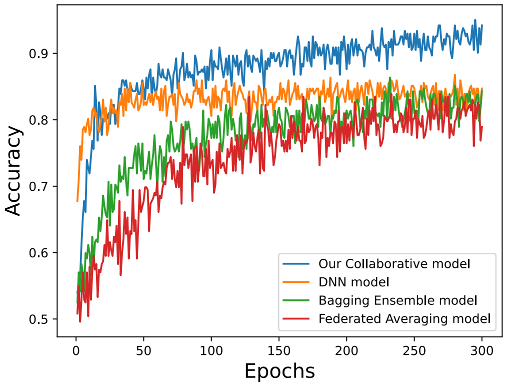

FIGURE 9. Accuracy-based learning curve of our model on DDoS attack detection compared to other models using ICU training dataset.

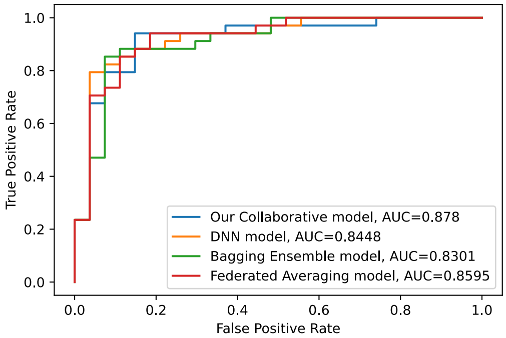

FIGURE 10. ROC curve of our model on DDoS attack detection compared to other models using ICU test dataset.

3) NUMERICAL COMPARISONS

Table 8 compares the models in terms of their accuracy, precision, recall, F-score, and AIC delta score (△AIC) on the prediction of the test datasets. It demonstrates that our model performs well under most of the prediction metrics on all the datasets. Due to its large number of parameters, it turns out to have a higher △AIC value than the other models. However, such a large number of parameters is due to the collaborative framework, which is more of a merit than a demerit as explained in Section III.

Apart from these predictive performance results, we also evaluated the computational performance on the prediction actions of the models based on the FLOPs, FLOPS, and MACs computational measures. The experiments were

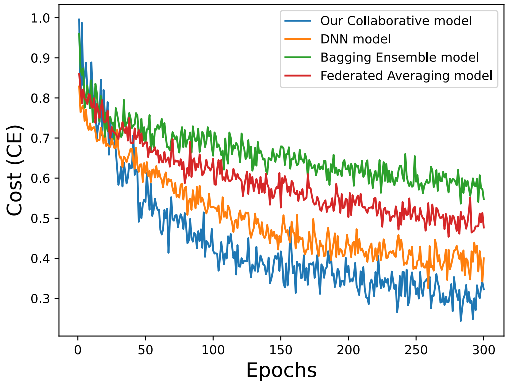

FIGURE 11. Cost-based learning curve of our model on DDoS attack detection compared to other models using ToN_IoT training dataset.

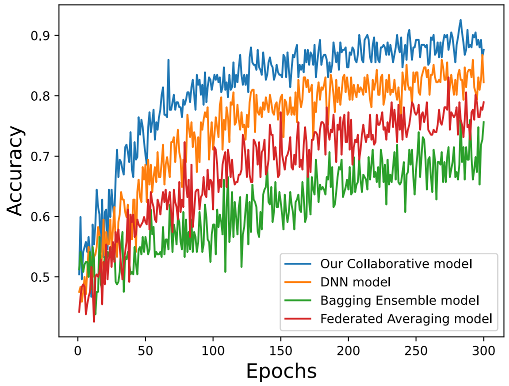

FIGURE 12. Accuracy-based earning curve of our model on DDoS attack detection compared to other models using ToN_IoT training dataset.

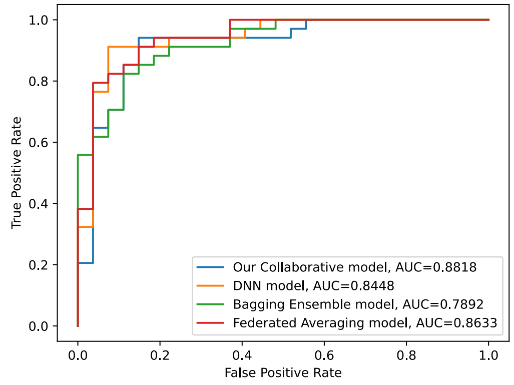

FIGURE 13. ROC curve of our model on DDoS attack detection compared to other models using ToN_IoT test dataset.

TABLE 8. Predictive performance and comparison of the models.

Datasets	Models	Acc(%)	Prec(%)	Rec(%)	F-s (%)	△AIC
ECU-IoHT	DNN	98.17	95.10	89.04	91.21	36.83
ECU-IoHT	Ensemble	98.78	93.03	96.08	94.26	74.54
ECU-IoHT	Federated	97.59	94.10	90.74	92.36	65.16
ECU-IoHT	Our model	**99.99**	**99.99**	**99.86**	**99.57**	**110.78**
ICU	DNN	99.19	93.73	86.91	96.55	37.14
ICU	Ensemble	99.48	96.89	91.99	97.93	75.32
ICU	Federated	99.01	96.32	90.28	95.59	66.07
ICU	Our model	**99.96**	**99.97**	**99.87**	**99.64**	**109.02**
ToN_IoT	DNN	98.80	94.91	90.89	93.90	37.10
ToN_IoT	Ensemble	98.99	98.19	94.57	99.99	72.11
ToN_IoT	Federated	98.92	95.81	92.91	97.80	63.52
ToN_IoT	Our model	**99.98**	**99.95**	**99.65**	**99.97**	**112.78**

Accuracy (Acc), Precision (Prec), Recall (Rec), F-score (F-s), and Akaike Information Criterion delta score (△AIC) with $\text{AIC}_{min} = 0$.

TABLE 9. Computational performance and uncertainty of the models.

Datasets	Models	FLOPs	Latency	FLOPS	MACs	ME	2σ-U
ECU-IoHT	DNN	1.1M	21ms	51.90M	0.55M	94.09	0.053
ECU-IoHT	Ensemble	2.2M	23ms	94.74M	1.1M	101.12	0.081
ECU-IoHT	Federated	1M	30ms	33.31M	0.5M	105.69	0.094
ECU-IoHT	Our model	**2M**	**27ms**	**74.12M**	**1M**	**69.27**	**0.032**
ICU	DNN	18.86M	38ms	0.53G	9.43M	112.55	0.074
ICU	Ensemble	37.72M	41ms	0.90G	18.86M	120.93	0.052
ICU	Federated	18.86M	45ms	0.41G	9.43M	118.01	0.071
ICU	Our model	**37.75M**	**42ms**	**0.94G**	**18.88M**	**70.04**	**0.023**
ToN_IoT	DNN	29.24M	56ms	0.54G	14.62M	93.90	0.094
ToN_IoT	Ensemble	58.48M	57ms	1.02G	29.24G	134.90	0.063
ToN_IoT	Federated	29.24M	61ms	0.52G	14.62M	125.14	0.081
ToN_IoT	Our model	**58.51M**	**59ms**	**1.18G**	**29.26M**	**89.04**	**0.051**

Floating-point operations (FLOPs), Floating-point operation per second (FLOPS), Multiply accumulate operations (MACs), median error (ME), 2σ uncertainty (2σ-U), Mega (M), millisecond (ms), Giga (G).

TABLE 10. Comparison of our model with models from other research works.

References	Models	Datasets	Acc(%)	Prec(%)	Rec(%)	F-s(%)
Ilhan et al. [5]	MLP	ECU-IoHT	99.99	99.99	99.9	99.99
Current work	Collabo	ECU-IoHT	**99.99**	**99.99**	**99.86**	**99.57**
Ilhan et al. [5]	MLP	ICU	99.94	99.94	99.92	99.93
Hussain et al. [6]	NB	ICU	52.18	79.67	99.70	68.50
	KNN	ICU	99.48	99.65	99.68	99.58
	RF	ICU	99.51	99.70	99.79	99.65
	AB	ICU	99.50	99.55	99.44	99.47
	LR	ICU	99.50	95.28	90.35	94.70
	DT	ICU	99.47	99.69	99.79	99.63
Current work	Collabo	ICU	**99.96**	**99.97**	**99.87**	**99.64**
Ilhan et al. [5]	MLP	ToN_IoT	98.12	98.15	98.12	98.12
Khan et al. [7]	XSRU	ToN_IoT	99.38	99.39	98.99	99.37
Zachos et al. [8]	DT	ToN_IoT	99.97	99.97	99.91	99.94
	NB	ToN_IoT	34.44	27.91	99.97	43.64
	LR	ToN_IoT	98.70	95.52	99.55	97.50
	RF	ToN_IoT	99.96	99.89	99.95	99.92
	KNN	ToN_IoT	99.98	99.98	99.97	99.96
	SVM	ToN_IoT	98.73	95.30	99.93	97.56
Current work	Collabo	ToN_IoT	**99.98**	**99.95**	**99.65**	**99.97**

Multi-Layer Perceptron (MLP), Naive Bayes (NB), K-Nearest Neighbor (KNN), Random Forest (RF), AdaBoost (AB), Logistic Regression (LR), Decision Tree (DT), Explainable simple recurrent units Internet of Medical Things (XSRU-IoMT), and Support Vector Machine (SVM)

performed on a quadcore 4GHz processor with 16 double-precision (DP) FLOPs per cycle (i.e., 256 GFLOPS DP), but the required FLOPS of the models were estimated using the latency (execution time) and FLOPs of their predictions.

Furthermore, the uncertainties (epistemic) of the models for each dataset were also evaluated using the ensemble method with 25 ensembles for each model.

As shown in Table 9, the FLOPs, latency, FLOPS, and MACs values of our model are higher than the other models. This is due to its collaborative framework consisting of many computing agents. However, the generalization (i.e., error) and generalization uncertainty (i.e., error variability) of our model are better than those of the other models on all the datasets because of its lower median and 2σ uncertainty values, respectively. This can also be confirmed by the graphs of the normalized cost values in Section IV-B1–IV-B2.

Also, the fact that the required FLOPS of the models are less than the 256 GFLOPS can be due to many factors such as computational overload, memory operations, and latency degradations of the computing platform.

Lastly, in Table 10 we compare the statistical performance results of our model with those of other studies using the same datasets in medical IoT security.

From the prediction results in Table 10, our model outperforms most of the other models on the ICU and ToN_IoT datasets under most of the metrics, but the model proposed by Ilhan et al. [5] has a better recall and F-score on the ECU-IoHT dataset and a better recall on the ICU dataset. In addition, the DT and KNN models proposed by Zachos et al. [8] have better precision and recall. These are some of the few instances in which the other models outperformed the proposed model.

V. CONCLUSION

This study aimed to detect DDoS attacks on medical IoT systems using a collaborative machine learning approach. We defined the agents for this purpose and established a mutual value exchange mechanism among the agents on their separate causal actions on the target.

The drawback of the proposed model is that it requires higher design complexity than conventional feedforward neural networks, making it more computationally expensive.

Nevertheless, apart from having a better generalization on a given dataset, joining two networks to collaborate synchronously or asynchronously on a target is an important feature of our model. This is because such a collaborative technique is a type of multi-task learning that can be used to solve spatial and temporal identical multi-label problems in machine learning by attributing each agent in the collaborative network an instance of the same target, in a temporal and/or spatial domain, and letting them share mutual values with other agents during learning to enhance their performance on the target. Insofar as they exchange mutual value between themselves, their collaborative action on the target, given in Proposition 3.1, will be their partial actions on their respective target instances irrespective of space and time.

Furthermore, related to the security and privacy of medical IoT data, devices, and networks, one of the advantages of this model is the use of information from distributed (segmented) environments. Similar to federated learning, this provides high data and device privacy that enhances data and device security properties such as integrity and confidentiality. This is because one of the major cyber security vulnerabilities for medical IoT networks is the heterogeneity of their data sources. If any data source is compromised, the whole system is at risk of an attack. The challenge with conventional models is that they involve the building of different independent solutions for a common attack based on the data sources. The Collabo approach provides a single model that takes into account all the different segments of the data sources used by any given medical IoT network.

In both the collabo and federated model, the prediction of a security attack such as DDoS further provides security related to the availability and integrity of the data, device, and network. But unlike federated learning which approaches the security problem by exchanging learning parameters via a central server, the collabo approach uses a mutual value, which is exchanged between the agents without the use of any central agent. This explains its low latency as compared to federated learning in Table 9.

Finally, the proposed collaborative machine learning model, where multiple learning agents are used to learn and predict a target in a heterogeneous environment, may be deployed in different applications where heterogeneity is present such as in signal processing, natural language processing, and autonomous driving.

APPENDIX A PROOF OF PROPOSITION 3.1

Consider the joint probability distribution $P(X_a, X_b, y)$, which can be expressed as

$$P(X_a, X_b, y) = P(X_b)P(X_a|X_b)P(y|X_a, X_b) \tag{A.1}$$

or

$$P(X_a, X_b, y) = P(y)P(X_a|y)P(X_b|X_a, y) \tag{A.2}$$

Equating (A.1) and (A.2), we obtain

$$P(y|X_a, X_b) = P(y)P(X_a|y)P(X_b|X_a, y)\frac{1}{P(X_b)P(X_a|X_b)} \tag{A.3}$$

Applying Axiom 3.1, we obtain

$$P(y|X_a, X_b) = P(y)P(X_a|y)P(X_b|y)\frac{1}{P(X_b)P(X_a|X_b)} \tag{A.4}$$

Applying Bayes rule to $P(X_a|y)$ and $P(X_b|y)$, we have

$$P(y|X_a, X_b) = P(y|X_a)P(y|X_b)\left[\frac{P(X_a|X_b)}{P(X_a)}\right]^{-1}\frac{1}{P(y)} \tag{A.5}$$

REFERENCES

[1] V. Chamola, V. Hassija, V. Gupta, and M. Guizani, "A comprehensive review of the COVID-19 pandemic and the role of IoT, drones, AI, blockchain, and 5G in managing its impact," *IEEE Access*, vol. 8, pp. 90225–90265, 2020.

[2] M. Elhoseny, N. N. Thilakarathne, M. I. Alghamdi, R. K. Mahendran, A. A. Gardezi, H. Weerasinghe, and A. Welhenge, "Security and privacy issues in medical Internet of Things: Overview, countermeasures, challenges and future directions," *Sustainability*, vol. 13, no. 21, p. 11645, Oct. 2021.

[3] J. Wang, Y. Yang, T. Wang, R. S. Sherratt, and J. Zhang, "Big data service architecture: A survey," *J. Internet Technol.*, vol. 21, no. 2, pp. 393–405, 2020.

[4] A. Rghioui, S. Sendra, J. Lloret, and A. Oumnad, "Internet of Things for measuring human activities in ambient assisted living and e-health," *Netw. Protocols Algorithms*, vol. 8, no. 3, p. 15, Dec. 2016.

[5] I. F. Kilincer, F. Ertam, A. Sengur, R.-S. Tan, and U. R. Acharya, "Automated detection of cybersecurity attacks in healthcare systems with recursive feature elimination and multilayer perceptron optimization," *Biocybern. Biomed. Eng.*, vol. 43, no. 1, pp. 30–41, Jan. 2023.

[6] F. Hussain, S. G. Abbas, G. A. Shah, I. M. Pires, U. U. Fayyaz, F. Shahzad, N. M. Garcia, and E. Zdravevski, "A framework for malicious traffic detection in IoT healthcare environment," *Sensors*, vol. 21, no. 9, p. 3025, Apr. 2021.

[7] I. A. Khan, N. Moustafa, I. Razzak, M. Tanveer, D. Pi, Y. Pan, and B. S. Ali, "XSRU-IoMT: Explainable simple recurrent units for threat detection in Internet of Medical Things networks," *Future Gener. Comput. Syst.*, vol. 127, pp. 181–193, Feb. 2022.

[8] G. Zachos, I. Essop, G. Mantas, K. Porfyrakis, J. C. Ribeiro, and J. Rodriguez, "An anomaly-based intrusion detection system for Internet of Medical Things networks," *Electronics*, vol. 10, no. 21, p. 2562, Oct. 2021.

[9] G. Kaur and P. Gupta, "Detection of distributed denial of service attacks for IoT-based healthcare systems," *Comput. Assist. Methods Eng. Sci.*, vol. 30, no. 2, pp. 167–186, Jun. 2022.

[10] M. J. Awan, U. Farooq, H. M. A. Babar, A. Yasin, H. Nobanee, M. Hussain, O. Hakeem, and A. M. Zain, "Real-time DDoS attack detection system using big data approach," *Sustainability*, vol. 13, no. 19, p. 10743, Sep. 2021.

[11] M. Alduailij, Q. W. Khan, M. Tahir, M. Sardaraz, M. Alduailij, and F. Malik, "Machine-learning-based DDoS attack detection using mutual information and random forest feature importance method," *Symmetry*, vol. 14, no. 6, p. 1095, May 2022.

[12] A. Maslan, K. M. B. Mohamad, and F. B. M. Foozy, "Feature selection for DDoS detection using classification machine learning techniques," *IAES Int. J. Artif. Intell.*, vol. 9, no. 1, p. 137, Mar. 2020.

[13] E. C. P. Neto, S. Dadkhah, and A. A. Ghorbani, "Collaborative DDoS detection in distributed multi-tenant IoT using federated learning," in *Proc. 19th Annu. Int. Conf. Privacy, Secur. Trust (PST)*, Fredericton, NB, Canada, Aug. 2022, pp. 1–10.

[14] S. I. Popoola, R. Ande, B. Adebisi, G. Gui, M. Hammoudeh, and O. Jogunola, "Federated deep learning for zero-day botnet attack detection in IoT-edge devices," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3930–3944, Mar. 2022.

[15] K. M. J. Rahman, F. Ahmed, N. Akhter, M. Hasan, R. Amin, K. E. Aziz, A. K. M. M. Islam, M. S. H. Mukta, and A. K. M. N. Islam, "Challenges, applications and design aspects of federated learning: A survey," *IEEE Access*, vol. 9, pp. 124682–124700, 2021.

[16] A. Bogdanova, N. Attoh-Okine, and T. Sakurai, "Risk and advantages of federated learning for health care data collaboration," *ASCE-ASME J. Risk Uncertainty Eng. Syst. A, Civil Eng.*, vol. 6, no. 3, Sep. 2020, Art. no. 04020031.

[17] M. A. Ferrag, O. Friha, L. Maglaras, H. Janicke, and L. Shu, "Federated deep learning for cyber security in the Internet of Things: Concepts, applications, and experimental analysis," *IEEE Access*, vol. 9, pp. 138509–138542, 2021.

[18] J. Bhayo, S. A. Shah, S. Hameed, A. Ahmed, J. Nasir, and D. Draheim, "Towards a machine learning-based framework for DDOS attack detection in software-defined IoT (SD-IoT) networks," *Eng. Appl. Artif. Intell.*, vol. 123, Aug. 2023, Art. no. 106432.

[19] T. Martin, "On the need for collaborative intelligence in cybersecurity," *Electronics*, vol. 11, no. 13, p. 2067, Jun. 2022.

[20] M. Ahmed, S. Byreddy, A. Nutakki, L. F. Sikos, and P. Haskell-Dowland, "ECU-IoHT: A dataset for analyzing cyberattacks in Internet of Health Things," *Ad Hoc Netw.*, vol. 122, Nov. 2021, Art. no. 102621.

[21] J. Pearl, *Causality: Model, Reasoning, and Inference*, vol. 19, no. 4. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[22] C. Hitchcock, "Probabilistic causation," in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed. Stanford, CA, USA: Stanford Univ., Spring 2021.

[23] R. C. Koons, "A representational account of mutual belief," *Synthese*, vol. 81, no. 1, pp. 21–45, Oct. 1989.

[24] I. D. Mienye and Y. Sun, "A survey of ensemble learning: Concepts, algorithms, applications, and prospects," *IEEE Access*, vol. 10, pp. 99129–99149, 2022.

[25] C. Zhang, Y. Xie, B. Hang, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowl.-Based Syst.*, vol. 216, Mar. 2021, Art. no. 106775.

[26] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, Oct. 2020.

[27] M. Vihinen, "How to evaluate performance of prediction methods? Measures and their interpretation in variation effect analysis," *BMC Genomics*, vol. 13, no. 4, p. S2, 2012.

[28] C. A. da Costa, C. F. Pasluosta, B. Eskofier, D. B. da Silva, and R. da Rosa Righi, "Internet of Health Things: Toward intelligent vital signs monitoring in hospital wards," *Artif. Intell. Med.*, vol. 89, pp. 61–69, Jul. 2018.

[29] S. Ghazanfar, F. Hussain, A. U. Rehman, U. U. Fayyaz, F. Shahzad, and G. A. Shah, "IoT-flock: An open-source framework for IoT traffic generation," in *Proc. Int. Conf. Emerg. Trends Smart Technol. (ICETST)*, Mar. 2020, pp. 1–6.

[30] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems," *IEEE Access*, vol. 8, pp. 165130–165150, 2020.

ZIE EYA EKOLLE (Member, IEEE) received the master's and Ph.D. degrees in engineering from the Department of Electrical and Computer Engineering, Yokohama National University, Japan, in 2019 and 2023, respectively. His current research interests include computational theory, information systems, information geometry, dependable systems, cognitive and rational intelligent agents, explainable and ethical AI, and collaborative machine learning.

HIDEKI OCHIAI (Fellow, IEEE) received the B.E. degree in communication engineering from Osaka University, Osaka, Japan, in 1996, and the M.E. and Ph.D. degrees in information and communication engineering from The University of Tokyo, Tokyo, Japan, in 1998 and 2001, respectively. From 1994 to 1995, he was with the Department of Electrical Engineering, University of California at Los Angeles (UCLA), Los Angeles, CA, USA, under the scholarship of the Ministry of Education, Science and Culture. From 2001 to 2003, he was a Research Associate with The University of Electro-Communications, Tokyo. Since April 2003, he has been with Yokohama National University, Yokohama, Japan, where he is currently a Professor. From 2003 to 2004, he was a Visiting Scientist with Harvard University, Cambridge, MA, USA. From 2019 to 2020, he was a Visiting Professor with the University of Waterloo, ON, Canada, and a Visiting Fellow with Princeton University, Princeton, NJ, USA. His research interest includes wireless communications and networks. He served as an Editor for IEEE Transactions on Wireless Communications, from 2007 to 2011, and IEEE Wireless Communications Letters, from 2011 to 2016.

RYUJI KOHNO (Life Fellow, IEEE) received the Ph.D. degree from the Department of Electrical Engineering, The University of Tokyo, in 1984. From 1984 to 1985, he was a Visiting Scientist with the Department of Electrical Engineering, University of Toronto. He was the Director of Sony CSL/ATL, from 1998 to 2002. He has been a Professor, since 1998, retired, and a Professor Emeritus with Yokohama National University (YNU), since 2021. He was the Principal Leader of the MEXT 21st Century Program and the Global COE Program, YNU, from 2002 to 2007 and from 2008 to 2013, respectively. From 2002 to 2011, he was the Director of the UWB Technical Institute. He was a Program Coordinator with the Medical ICT Institute, National Institute of Information and Communications Technology (NICT), Japan, from 2002 to 2011. Since 2003, he has been the Director of the Medical ICT Center, YNU. From 2007 to 2020, he was a Finnish Distinguished Professor with the University of Oulu, Finland. From 2012 to 2021, he was the CEO of the University of Oulu Research Institute Japan-CWC-Nippon Company. Since 2021, he has been the Vice President of YRP International Alliance Institute Ltd. Since 2006, he has been an Associate Member of the Science Council of Japan. He is currently an IEICE Fellow. He was elected as a BoG Member of the IEEE Information Theory Society, in 2000, 2002, and 2006. He received the IEICE Greatest Contribution Award and the NTT DoCoMo Mobile Science Award, in 1999 and 2002, respectively.