IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 25 December 2024, accepted 5 January 2025, date of publication 7 January 2025, date of current version 13 January 2025.

*Digital Object Identifier 10.1109/ACCESS.2025.3526883*

RESEARCH ARTICLE

# L2D2: A Novel LSTM Model for Multi-Class Intrusion Detection Systems in the Era of IoMT

**GÖKHAN AKAR[1], SHAABAN SAHMOUD[2], MUSTAFA ONAT[1], (Member, IEEE), ÜNAL CAVUSOGLU[3], (Member, IEEE), AND EMMANUEL MALONDO[4]**

[1]Department of Electrical and Electronics Faculty (Engineering), Marmara University, 34722 İstanbul, Türkiye
[2]Engineering Faculty, Fatih Sultan Mehmet Vakif University, 34664 İstanbul, Türkiye
[3]Computer Sciences Faculty, Sakarya University, 54187 Sakarya, Türkiye
[4]Engineering Faculty, CESI University, 76130 Rouen, France

Corresponding author: Gökhan Akar (gokhanakar@gmail.com)

This work was supported in part by Turk Telekom A. S., Türkiye.

**ABSTRACT** The rapid growth of IoT has significantly changed modern technology by allowing devices, systems, and services to connect easily across different areas. Due to the growing popularity of Internet of Things (IoT) devices, attackers focus more and more on finding new methods, ways, and vulnerabilities to penetrate IoT networks. Although IoT devices are utilized across a wide range of domains, the Internet of Medical Things (IoMT) holds particular significance due to the sensitive and critical nature of medical information. Consequently, the security of these devices must be treated as a paramount concern within the IoT landscape. In this paper, we propose a novel approach for detecting various intrusion attacks targeting Internet of Medical Things (IoMT) devices, utilizing an enhanced version of the LSTM deep learning algorithm. To evaluate and compare the proposed algorithm with other methods, we used the CICIoMT2024 dataset, which encompasses various types of equipment and corresponding attacks. The results demonstrate that the proposed novel approach achieved an accuracy of 98% for 19 classes, which is remarkably high for classifications and presents a significant and promising outcome for IoMT environments.

**INDEX TERMS** Internet of Medical Things (IoMT), intrusion detection system, Internet of Things Security, security of healthcare systems.

## I. INTRODUCTION

It was 1999 when Ashton used the word "Internet of Things" (IoT) in his presentation [1]. IoT is defined as a distributed networked system that communicates via wired or wireless technologies. This special network contains sensors, actuators, software, and network connectivity that allow these objects to gather, occasionally process and exchange data. These objects may also have limited computation, storage, energy consumption, and communication capabilities [2].

This kind of network is used for a variety of areas such as [3]:

- *Smart homes*: IoT technology enables the automation and remote control of devices such as lighting, thermostats, and security systems, enhancing convenience and energy efficiency [4].
- *Industrial IoT*: IIoT is used for smart manufacturing [5].
- *Smart vehicles*: IoT is used to connect different types of vehicles for control and management, traffic management and fleet tracking [6].
- *Smart cities*: IoT is utilized for infrastructure monitoring, urban planning and energy management [7].
- *Smart agriculture* : IoT is applied for precision farming, livestock monitoring and resource management [8].
- *Climate Change*: Researchers also deploy IoT for climate monitoring, disaster management, and renewable energy integration. [9].

The associate editor coordinating the review of this manuscript and approving it for publication was Renato Ferrero.

The Internet of Medical Things (IoMT) has recently emerged as one of the most critical domains within the broader IoT ecosystem. IoMT is primarily used for remote patient monitoring, enabling the continuous tracking of medication adherence, vital signs, and the integration of medical equipment and implants [10]. Other notable applications of IoMT can be counted as wearable devices such as smart watches, textile-based wearable systems, fitness trackers, health monitoring devices, as well as telehealth services [11]. These innovations have significantly advanced the healthcare sector by improving patient care and facilitating real-time health management.

Like all IoT devices, IoMT devices are connected to the internet and making them vulnerable to cyberattacks. However, the critical nature of healthcare services and the sensitivity of medical data amplify the significance of security issues in IoMT environments. According to CrowdStrike's Global Threat Report 2024 [12], 8% of intrusion attacks targeted healthcare systems. The report further reveals that, in some cases, private mental and physical healthcare data were either exposed or publicly shared during these attacks.

To counter cyberattacks on IT systems, Intrusion Detection and/or Prevention Systems (IDS/IPS) are commonly deployed. These systems monitor network traffic, detect anomalies, and identify potential security breaches in real-time. While some rely on signature-based detection, others have integrated artificial intelligence-enhanced sensors. From AI perspective, various technologies, techniques, and models have been proposed by researchers, as outlined in the Related Works section of this paper. Based on our extensive experiments and results, we focused on using the LSTM model to solve this problem. In this research work, we propose a novel model utilizing the LSTM model to accurately detect the various types of attacks that occur in IoMT devices and networks as a multi-classification tool. The CICIoMT24 dataset is used to assess the effectiveness of the proposed LSTM approach [13].

The CICIoMT2024 dataset was created by the Canadian Institute for Cybersecurity (CIC) as a comprehensive dataset designed for research in the Security of the Internet of Medical Things to simulate real-world IoMT environments. It includes various types of network traffic data including benign and malicious activities.

The layout of this paper includes the following: In Section II, a brief explanation of IoMT along with some security subjects is provided and discusses the LSTM model, while Section III reviews Related Works on IoMT security. The used CICIoMT2024 Dataset is introduced with preprocessing in Section IV. The proposed algorithm is explained in detail in Section V, the Section VI tells about the experiments and results discusses about limitations and constraints and the Section VII concludes the research and findings.

## II. BACKGROUND

### A. INTERNET OF MEDICAL THINGS

The Internet of Medical Things has emerged as a new concept in IoT terminology and is a subset of IoT in which medical devices are equipped with smart equipments to create, transfer and store medical data. Its technology facilitates the healthcare sector by enabling the connection and integration of medical devices, healthcare systems, and patient data.

Zois at their paper described a set of exemplary health applications that use such capabilities as telemedicine services, smart medication management systems, real-time health data tracking, and remote patient monitoring [14]. Furthermore, by giving medical staff real-time access to patient data and facilitating prompt interventions and individualized treatment plans, IoMT devices have the potential to improve patient outcomes and the general quality of care. Additionally, people can gain a great deal from IoMT devices by being able to regularly review their health records and take preventative measures without having to visit the doctor as often as without IoMT. However, lower hospital expenses may result from the adoption of IoMT devices in healthcare by eliminating avoidable problems and minimizing the visits to hospitals. The Internet of Medical Things has the potential to totally change the healthcare sector by lowering costs and enabling remote health monitoring.

For medical issues and easy usage, the IoMT devices are used either as implantable (such as cochlear implant, or deep brain stimulator) or wearable (such as Smart watches, ECG, and blood pressure monitors). Avinashiappan et al. classify IoMT devices according to their locations as: Community, In-Hospital, In-Clinic, In-home and On-body [15]. Since we are talking about devices that use the internet for connection, apparently it is considerable that these devices are susceptible to cyberattacks and illegal access to patient data by cybercriminals [16].

Despite the Internet of Medical Things presents a revolution in healthcare, according to Yaacoub et al., its potential is marred by significant security concerns [17]. At Device-Level Security and Network-Level Security, the followings can be considered for measures:

- There should be put in place measures like secure boot to prevent unauthorized code execution and ensure timely firmware updates to address vulnerabilities.
- To prevent unwanted access to devices and data, usage of role-based access control and enforcement of multi-factor authentication (MFA) should be considered.
- Robust encryption methods should be utilized to encrypt data while it is in use, in transit, and at rest, making it unreadable in the event of a breach.
- To reduce vulnerabilities during development, developers should adhere to secure coding techniques.
- To mitigate the impacts of the attacks and to isolate IoMT devices from vital healthcare systems, segmentation of the network should be considered.

- Usage of Intrusion Detection and Prevention Systems (IDS/IPS) to stop harmful network activity directed at IoMT devices and continuous monitoring to spot suspicious device activities and network traffic is essential.
- In IoMT networks, AI (ML/DL) techniques can be utilized for anomaly detection and real-time threat detection.

### B. LONG SHORT-TERM MEMORY (LSTM) NETWORKS IN DEEP LEARNING

Long Short-Term Memory (LSTM) is a special kind of recurrent neural network (RNN) architecture, which eliminates the vanishing gradient problem [18]. The key components of an LSTM Cell are Input Gate, Forget Gate, Cell State, and Output Gate.

At a LSTM Cell, Input Gate controls the flow of information into the cell state and decides which parts of the current input should be updated. At Forget Gate which information from the previous cell state should be forgotten is determined. Cell State stores information over time. By being updated based on the input and the previous cell state, decisions of the input and forget gates accomplished. Then the Output Gate decides which parts of the cell state should be the output.

Since LSTMs can retain information for extended periods of time and are built to solve the vanishing gradient problem, they are especially helpful in sequences where earlier inputs affect later outputs so they have the ability to handle long-term dependencies.

## III. RELATED WORKS

In healthcare and medical systems, security measures must be implemented with exceptional care and rigor due to the critical nature of medical data and the growing frequency of cyberattacks targeting healthcare infrastructures. Consequently, ensuring the security of the IoMT has become a significant research focus, motivated by both the sensitivity of medical information and the increasing threat landscape. Intrusion detection systems (IDS) play a crucial role in safeguarding sensitive medical data within IoMT environments by identifying malicious activities and unauthorized access. Various studies have explored the application of deep learning (DL) and machine learning (ML) methods to bolster IDS effectiveness in IoMT networks. For example, Anitha et al. explored some ML techniques to detect attacks, concluding that kNN yielded the best results with an accuracy of 89.79% [19]. Their work utilized IEEE Data Port datasets for binary classification purposes.

In another study, Ksibi et al. employed the novel ECU-IoHT dataset. Their binary classification efforts, leveraging the Random Forest ML algorithm, achieved an accuracy of 99.76% after using SMOTE for data balancing [20]. Tanzila Saba advocated for ensemble classifiers, such as bagged decision trees based on the bagging algorithm, to detect attacks against Smart City Hospitals. Her model achieved 93.2% accuracy using the KDDCup'99 dataset, which encompasses 5 classes [21]. Alsalman proposed FusionNet, a model combining Support Vector Machine, K-Nearest Neighbors, Random Forest, and Multi-Layer Perceptron for anomaly detection. According to his paper, this model reached 98.5% accuracy on the WUSTL EHMS 2020 Dataset and 99.5% on ICU-IoMT for binary classification [22]. Sun et al. attained a 98.5% accuracy by using Particle Swarm Optimization and AdaBoost on the NSL-KDD dataset, which includes 5 classes, including normal traffic [23]. Balhareth et al. worked with the CICIDS2017 dataset, applying Mutual Information and XGBoost for feature selection, resulting in a binary classification accuracy of 98.79% [24].

Deep learning, known for its ability to autonomously uncover complex patterns, has also been widely used in IDS for IoMT. Awotunde et al. used a Deep Auto Encoder on the NF-ToN-IoT dataset as an intrusion detection mechanism for secured IoMT systems, achieving 89% accuracy for a 10-class multi-classification [25]. Kulshrestha and Kumar, in their study using the ToN–IoT dataset, claimed around 99% accuracy for 4-class classification with the AdaBoost classifier [26]. Khan et al. proposed a hybrid CNN-LSTM model to address feature interdependencies and improve feature learning, which they applied to the IoT Malware dataset, achieving an approximate 99% accuracy for binary classification [27].

Combining more classifiers often uses additional system resources, yet hybrid AI techniques have demonstrated efficacy in enhancing IDS performance. Liaqat et al. proposed a hybrid DL architecture combining CNN and cuDNNLSTM for the IoMT environment employing the Bot–IoT dataset. They compared multiple configurations, concluding that CNN with cuDNNLSTM was the most effective, achieving an accuracy close to 99.99% for 3-class classification [28]. Faruqui et al. devised a model combining CNN and LSTM to improve cybersecurity in IoMT, reporting an average accuracy rate of 97.63% for 12 classes on the CIC-IDS2017 dataset, although it is not specific to IoMT [29].

Otoum et al. employed a Federated Transfer Learning-based IDS, and at most with achieving 95.1% accuracy rate for 3-classes multi-classification on the CICIDS2017-Tuesday dataset [30]. Ravi et al.'s CNN-LSTM model on the WUSTL EHMS 2020 dataset achieved 99% accuracy with 10-fold cross-validation in binary classification [31]. Khan et al. developed the XSRU-IoMT model based on a bidirectional simple recurrent unit (Bid-SRU), achieving a 99.38% accuracy for 8-class multi-classification on the ToN-IoT dataset [32]. Dadkhah et al. generated the CICIoMT2024 dataset to simulate an IoMT environment for IDS research. They tested Logistic Regression, AdaBoost, DNN, and Random Forest, achieving nearly 100% accuracy in binary classification but observing a decrease in accuracy to 73.3% in 19-class scenario [33]. Sánchez et al. studied the application of fine-tuning Transformer designs also using the CICIoMT2024 dataset and assessed it using the Aposemat IoT-23 dataset. Their experiments on their proposed model

**TABLE 1.** Related works for IoMT IDS with AI.

| Author | Dataset | ML/DL Technique | Classification Type (Number of Classes) | Accuracy |
|---|---|---|---|---|
| Anitha et al. [19] | IEEE Data Port | KNN | Binary | 89.79% |
| Ksibi et al. [20] | ECU-IoHT | Random Forest | Binary | 99.76% |
| Tanzila Saba [21] | KDDCup-99 | Bagged Decision Trees | Multi-class (5) | 93.20% |
| Alsalman [22] | WUSTL EHMS 2020, ICU-IoMT | FusionNet (SVM, KNN, RF, MLP) | Binary | 98.5%, 99.5% |
| Sun et al. [23] | NSL-KDD | PSO-AdaBoost | Multi-class (5) | 98.50% |
| Balhareth et al. [24] | CICIDS2017 | XGBoost | Binary | 98.79% |
| Awotunde et al. [25] | NF-ToN-IoT | Deep Autoencoder | Multi-class (10) | 89% |
| Kulshrestha et al. [26] | ToN-IoT | AdaBoost | Multi-class (4) | 99% |
| Khan and Akhunzada [27] | IoT Malware | CNN-LSTM | Binary | ∼99% |
| Liaqat et al. [28] | Bot-IoT | CNN-cuDNNLSTM | Multi-class (3) | ∼99.99% |
| Faruqui et al. [29] | CIC-IDS2017 | CNN-LSTM | Multi-class (12) | 97.63% |
| Otoum et al. [30] | CICIDS2017 | Federated Transfer Learning | Multi-class (3) | 95.10% |
| Ravi et al. [31] | WUSTL EHMS 2020 | CNN-LSTM | Binary | 99% |
| Khan et al. [32] | ToN-IoT | Bid-SRU | Multi-class (8) | 99.38% |
| Dadkhah et al. [33] | CICIoMT2024 | Logistic Regression, Adaboost, DNN, Random Forest | Multi-class (19) | 73.30% |
| Sánchez et al. [34] | CICIoMT2024, IoT-23 | XGBoost with Transformer | Multiclass (7) | 96% |

reached up to 96% for Accuracy, Precision, Recall and F1- Score metrics [34].

This overview highlights ongoing research into IoMT security and the intersection of advanced AI techniques and IDS frameworks continues to show promising advancements in the field. By reviewing the related works on IoMT, it is shown that the studies are continuing and this special part of IoT still attracts researchers to solve different research and security problems. At the same time, there is still a need to enhance the current proposed algorithms and propose new algorithms to handle and improve the achieved results especially for multi-classification as seen in the Table 1. The ongoing improvements in AI techniques also affect the approaches to IDS systems for IoMT, and the results reached are encouraging.

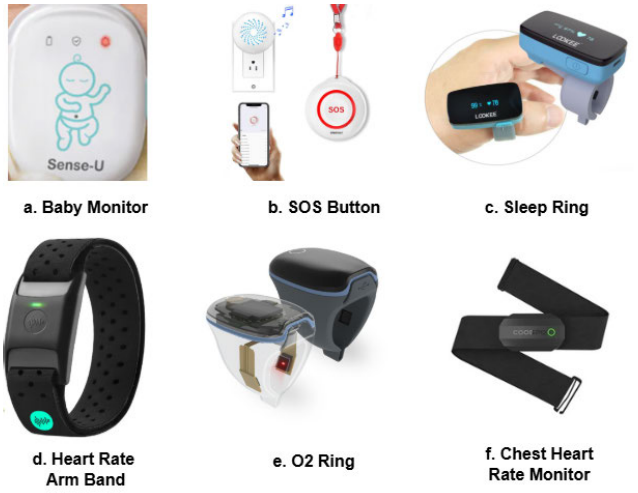
## IV. CICIOMT2024 DATASET

In our study, we utilized CICIoMT2024 dataset to test the proposed algorithms and models for better IDS mechanisms for IoMT environments. This dataset was gathered and issued by Dadkhah et al. from the Canadian Institute for Cybersecurity at the University of New Brunswick (UNB) [33].

### A. BRIEF INFORMATION ON THE DATASET

To produce this dataset for security solutions, the authors used a testbed consisting 25 real devices and 15 simulated devices [33]. The CICIoMT2024 dataset distinguishes itself from other IoT and IoMT datasets by offering comprehensive coverage of devices, attacks, and protocols tailored to healthcare security applications. While the datasets like BOT–IoT, ToN–IoT, UNSW–NB15 and Edge_IIoT are for IoT, but they do not contain significant medical devices because of the reason they were not produced especially for medical systems [35].

Some other IoMT datasets like WUSTL EHMS 2020 and ECU-IoHT focus on a narrow range of devices such as laptops, heart rate monitors, and temperature sensors, while CICIoMT2024 includes a much broader array of 40 IoMT devices, encompassing both real and simulated tools such as oxygen saturation sensors, glucometers, and heart rate monitors [36], [37]. This diversity provides a realistic representation of healthcare environments. Figure 1 represents some of the devices used during data gathering.



a. Baby Monitor

b. SOS Button

c. Sleep Ring

d. Heart Rate Arm Band

e. O2 Ring

f. Chest Heart Rate Monitor

**FIGURE 1.** Some IoMT devices used in CICIoMT2024 Dataset.

Figure 1.a is a baby monitoring device that integrates health tracking capabilities with video capabilities [38]. Figure 1.b is a SOS Button which is a wireless emergency warning device that is frequently used in homes, hospitals, and other establishments for general emergency signaling, patient aid, and elder care. When the SOS button is hit, it can transmit signals to a receiver or a linked mobile app over a Wi-Fi connection [39]. Figure 1.c is a sleep ring which is a small sleep monitoring gadget that measures heart rate and blood oxygen saturation levels [40]. Figure 1.d is an arm band to monitor heart rate [41]. Figure 1.e is a smart ring intended to

**TABLE 2.** List of devices used for the CICIoMT2024 test lab.

| The used equipments | Simulated devices |
|---|---|
| Sense-U Baby Monitor | Withings BPM Connect |
| SOS Multifunctional Pager | Withings Thermometer |
| SINGCALL SOS Button | Lookee Ring-Pro Sleep Monitor |
| Ecobee Camera | Qardio Base 2 |
| blink mini | Wellue EKG |
| M1T laxihub | iHealth Smart Wireless Gluco-Monitoring System |
| Owltron | Wellue Visual Oxy Wrist Pulse Oximeter |
| TP-Link_CIC (AP2) | Nasal/Mouth Air Flow Sensor |
| Raspberry pi 4 (4) | EMG (Electro-myography Sensor) |
| iPad | GSR (Galvanic Skin Response Sensor) |
| TP-Link_CICIoT_Doctor (AP1) | Industrial devices |
| Lookee Sleep ring | UASure II Meter |
| Powerlabs HR Monitor Arm band | Fall Detector |
| COOSPO 808s Chest HR Monitor | Baby Sleep Position - SenseU Baby |
| COOSPO HW807 Armband | Spirometer |
| Livlov Heart Rate Sensor | |
| Wellue O2 Ring - 3438 | |
| Lookee O2 Ring | |
| Checkme BP2A | |
| SleepU Sleep Oxygen Monitor | |
| Rhythm+ 2.0 | |
| Wellue Pulsebit EX | |
| Kinsa Thermometer | |
| Checkme O2 Wrist Pulse Oximeter (2) | |
| Dell CICM99 | |
| Samsung A11 | |

**TABLE 3.** Classes and sub-classes of the used dataset.

| Main Class | Sub Classes |
|---|---|
| *Benign* | Benign |
| *Spoofing* | ARP |
| *DDoS* | ICMP - SYN - TCP - UDP |
| *DoS* | ICMP - SYN - TCP - UDP |
| *MQTT* | DDoS Connect Flood - DDoS Publish Flood - DoS Connect Flood -DoS Publish Flood - Malformed Data |
| *Recon* | OS Scan - Ping Sweep - Port Scan - VulScan |

track heart rate and blood oxygen levels while you sleep [42]. Figure 1.f is another heart rate monitoring gets worn around the chest [43].

The list of used and simulated types of equipment is provided in Table 2. Table 3 gives the sorts of attacks from the CICIoMT2024 dataset, and in the Table 4 the counts of the logs are listed per attack type.

### B. TYPES OF ATTACKS IN CICIOMT2024

The 18 attacks in CICIoMT2024 dataset, cover a variety of protocols commonly used in healthcare, such as Wi-Fi, MQTT, and Bluetooth. They categorized the attacks into five classes: DDoS, DoS, Recon, MQTT and Spoofing.

We utilized this dataset to analyze three distinct class configurations: 2, 6, and 19 classes, respectively. The 2-class analysis solely focuses on distinguishing between benign and malicious attacks. The 6-class classification includes both benign instances and various types of attacks as DDoS, DoS, MQTT, Recon, and Spoofing. Additionally, the 19 subcategories within these classifications are detailed in the Table 3. The abbreviations in the Table 3 stand for:

- DoS (Denial of Service): DoS attacks cause services to be inaccessible by flooding a target with too many requests for computation or traffic [44].
- DDoS (Distributed Denial of Service): DDoS enhance DoS vulnerabilities by flooding a target with many systems, frequently botnets [45].
- MQTT (Message Queuing Telemetry Transport): MQTT is ideal for low-resource devices, especially in IoT applications. But its lightweight nature and lack of robust security mechanisms make it vulnerable to attacks [46].
- Recon (Reconnaissance): Systems that have unpatched software, detailed error messages, or unprotected ports are vulnerable to Reconnaissance attacks. Recon attacks, which frequently serve as a prelude to more focused operations, allow adversaries to map networks, locate open services, and find exploitable gaps. The OS Scan tries to identify operating system, Ping Sweep locates active hosts, Port Scan is utilized to identify open ports or services and VulScan detects the known vulnerabilities [47].
- ARP (Address Resolution Protocol): Because it lacks authentication by design, ARP is prone to spoofing.

**TABLE 4.** Quantity of instances in the CICIoMT2024 Dataset.

| Class | Category | Attack | Count |
|---|---|---|---|
| ATTACK | SPOOFING | ARP Spoofing | 17791 |
| | RECON | Port Scan | 106603 |
| | | OS Scan | 20666 |
| | | Recon VulScan | 3207 |
| | | Ping Sweep | 926 |
| | MQTT | DDoS Connect Flood | 214952 |
| | | DoS Publish Flood | 52881 |
| | | DDoS Publish Flood | 36039 |
| | | DoS Connect Flood | 15904 |
| | | Malformed Data | 6877 |
| | DoS | DoS UDP | 704503 |
| | | DoS SYN | 540498 |
| | | DoS ICMP | 514724 |
| | | DoS TCP | 462480 |
| | DDoS | DDoS UDP | 1998026 |
| | | DDoS ICMP | 1887175 |
| | | DDoS TCP | 987063 |
| | | DDoS SYN | 974359 |
| BENIGN | - | - | 230339 |

By sending fake ARP messages, attackers take advantage of this and can reroute traffic or carry out man-in-the-middle attacks on local area networks [48].

- ICMP (Internet Control Message Protocol): Ping floods and other flooding attacks, which overload a target by taking use of its response to ICMP requests, frequently take advantage of ICMP's availability to network diagnostics [49].
- SYN (Synchronize): The SYN flag is used to start a connection between two devices in the TCP handshake process. By sending a large number of connection requests without fulfilling them, attackers take advantage of this process in attacks like SYN floods, which exhaust the target server's resources [50].
- TCP (Transmission Control Protocol): Attackers can take over sessions or interfere with communication by taking use of TCP connection-oriented flaws [51].
- UDP (User Datagram Protocol): By sending small requests to servers that are incorrectly configured, UDP amplification attacks take advantage of connectionless communication. Systems lacking source validation or rate-limiting are therefore especially susceptible [52].

In terms of attack coverage, other datasets typically address limited scenarios, such as spoofing or DoS attacks. CICIoMT2024 goes further by simulating 18 attack types, including DDoS, MQTT-specific attacks, reconnaissance, and spoofing. These scenarios are designed to evaluate vulnerabilities across multiple dimensions, making the dataset more robust for research on IoMT cybersecurity.

CICIoMT2024 also stands out in its specific focus on healthcare, incorporating medical-grade devices and realistic attack simulations. Other datasets, though focused on IoT, lack the targeted application for critical healthcare settings. By addressing this gap, CICIoMT2024 provides a valuable resource for improving cybersecurity in IoMT, especially in environments where confidentiality, integrity, and availability are paramount..

CICIoMT2024 addresses the limitations of other datasets by providing a diverse testbed of devices and an extensive range of attacks. These features make it a robust resource for researchers aiming to develop advanced AI-based solutions to secure IoMT.

### C. PREPROCESSING

To prepare the CICIoMT2024 dataset for classification, a comprehensive preprocessing pipeline was applied to ensure data quality and model compatibility. The dataset contains 45 features and includes some incomplete and missing data [33]. Firstly, missing values were handled, through imputation methods using mean imputation for numerical data, or mode imputation for categorical features, in some cases by removing rows with excessive missing information. Data normalization or standardization was also applied to bring all feature values into comparable scales, enhancing the performance of distance-based and gradient-based classification algorithms. Additionally, to facilitate supervised learning, target class labels were appended to each row across all files in the dataset, ensuring that the model could correctly associate each data instance with its class during the training and testing phases. This organization enables the classifier to map features to their respective classes, streamlining the model's learning and evaluation process.

The features in the CICIoMT2024 dataset were selected based on their significance in identifying and addressing security threats specific to IoMT environments. These features include network traffic characteristics, device behaviors, and protocol-specific parameters, which are crucial for detecting anomalies, potential vulnerabilities, and malicious activities. For example, features such as packet size, flow duration, protocol type, and specific header fields were chosen because they provide insights into the nature of communication among IoMT devices. These parameters are essential for identifying unauthorized access attempts, data breaches, or abnormal communication patterns that could signal potential threats. Moreover, advanced techniques such as correlation analysis and feature importance ranking were employed to ensure that the selected features contribute effectively to enhancing the detection capabilities of machine learning models used in IoMT security applications. In our training and testing tasks, we left the feature selection to the LSTM deep learning algorithms like many other deep learning models.

## V. PROPOSED LSTM MODEL

LSTM models are widely recognized for their ability to capture and retain information over extended sequences, making them particularly effective in analyzing time-series data and patterns in sequential events. Due to their unique memory cell structure, LSTMs can selectively remember or forget information across longer time intervals, addressing the limitations of traditional RNNs that struggle with long-term dependencies [53]. This characteristic is precious in intrusion detection, where the model must identify anomalous behavior over a series of network events to distinguish between benign and malicious activities accurately. In this research, LSTM models have been employed to tackle the attack detection problem for IoMT environments, leveraging their capacity to learn complex temporal patterns in network traffic data, thus enhancing the detection of subtle and sophisticated attacks within an IoMT environment.

The proposed model in this research uses Two-LSTM and Two-Dense Layers and optimizes with AdamW optimizer. According to the results of Section VI, this model achieves the best results compared to many other LSTM and machine learning models. Figure 2 describes the structure of the proposed LSTM model in detail to handle the binary and especially multi-class classification problem in IoMT systems. Specifically, the proposed model applied to the instances of the CICIoMT2024 dataset in the case of 2, 6, and 19 classes respectively.

In the proposed model, the architecture with stacked LSTM layers, followed by dense (fully connected) layers, can effectively be used for tasks that require capturing sequential dependencies in time series or sequential data. The model comprising two LSTM layers followed by two dense layers, each with unique configurations and purposes, is described in detail as follows:

- Input Layer: The input layer is designed to handle sequential data of a fixed length and feature dimension. Typically, for LSTM models, the input is structured as a three-dimensional tensor with the shape (batch size, timesteps, features). Each element in the sequence represents a one-time step of data, and each time step includes a set of features.
- LSTM Layers: The core of the model consists of two stacked LSTM layers with 64 units each. LSTM layers are composed of memory cells that allow the network to retain information over long sequences, making them ideal for capturing temporal dependencies.
  - **First LSTM Layer**: It includes 64 hidden units that provide the memory cell with the capacity to capture complex sequential patterns in the input data. A ReLU (Rectified Linear Unit) activation function is applied that introduces non-linearity to the network, which allows the model to learn more complex patterns. Although the default activation in LSTMs is the sigmoid function, ReLU is used for more flexibility. The return sequences layer for LSTM is configured to return the full sequence output for each step, thus outputting a sequence of the same length as the input. This configuration is beneficial when stacking multiple LSTM layers, as it allows the subsequent LSTM layer to process the complete sequence of outputs.
  - **Second LSTM Layer**: Similar to the first LSTM layer, this layer has 64 units, maintaining the model's capacity to learn from temporal dependencies. The ReLU activation function is again applied to introduce non-linearity. For the layer of return sequences, we typically set the `return_sequence` parameter to False, which means it outputs only the final state of the LSTM after processing the entire input sequence. This is particularly useful when the subsequent layers in the network do not require sequential input.
- Dense (Fully Connected) Layers: Following the LSTM layers, two fully connected (dense) layers are included. Dense layers are commonly used after LSTMs to consolidate the learned features into a fixed output space and are useful for classification or regression tasks.
  - **First Dense Layer**: This layer has 64 neurons, providing substantial capacity to process and interpret the abstract representations learned by the LSTM layers. The ReLU activation function is employed again, enabling the layer to capture non-linear patterns and relationships in the data.
  - **Second Dense Layer (Output Layer)**: The final dense layer is typically configured to match the required output format. For instance, in binary classification, it would have a single unit with a sigmoid activation, while in multi-class classification, it would have as many units as the number of classes with a softmax activation.

## VI. EXPERIMENTS AND RESULTS

This section describes the two main experiments conducted to evaluate the performance of the proposed model, provides a detailed evaluation of the results, and discusses the limitations and constraints.

### A. PERFORMANCE COMPARISON OF THE LSTM MODEL WITH OTHER MACHINE LEARNING MODELS

To test the effectiveness of the LSTM model for solving the classification problem of the CICIoMT2024 dataset, we compared it with a set of well-known machine learning algorithms such as Logistic Regression and ANN for 2, 6, and 19 classes as shown in Tables 5, 6 and 7 respectively.

By analyzing the results of the 2-class classification problem, we found that the Logistic Regression achieved a strong performance with the metrics of accuracy, precision, recall, and F1-Score all equal to 98%. Deep Learning (DL) Artificial Neural Network (ANN) obtained the best classification performance with all metrics scoring 100%. Similarly, Deep Learning LSTM achieved perfect classification with 100% across all metrics. For the 2-class problem, both the DL ANN and LSTM significantly outperformed logistic regression, achieving perfect classification.

At the next step our aim was to assess the capabilities of the algorithms with 6-class classification. It has been obvious that Logistic Regression struggles with multi-classification with the values of 57% for accuracy and recall, precision with 37% and 43% for F1-Score. While ANN with Adam Optimization gives better results than LR(Logistic Regression)

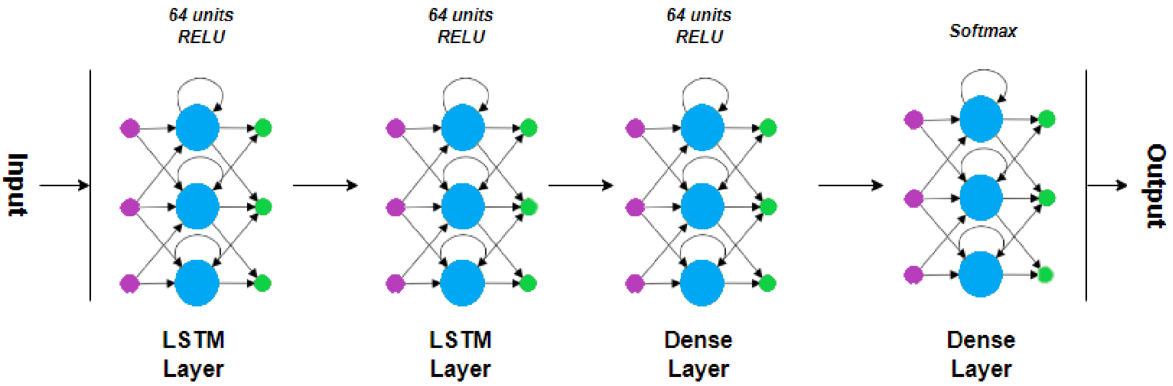


**FIGURE 2.** The proposed model with 2 LSTM and 2 dense layers.

**TABLE 5.** Classification results for 2-classes problem.

| Algorithm | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Logistic Regression | 0.98 | 0.98 | 0.98 | 0.98 |
| DL ANN + Adam (50 epochs) | **1** | **1** | **1** | **1** |
| DL LSTM (50 epochs) | **1** | **1** | **1** | **1** |

**TABLE 6.** Classification results for 6-classes problem.

| Algorithm | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Logistic Regression | 0.57 | 0.37 | 0.57 | 0.43 |
| DL ANN + Adam (50 epochs) | 0.73 | 0.73 | 0.73 | 0.70 |
| DL LSTM (50 epochs) | **0.98** | **0.98** | **0.98** | **0.98** |

**TABLE 7.** Classification results for 19-classes problem.

| Algorithms | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Logistic Regression | 0.30 | 0.12 | 0.30 | 0.17 |
| DL ANN + Adam (50 epochs) | 0.71 | 0.65 | 0.71 | 0.65 |
| DL LSTM (50 epochs) | **0.95** | **0.96** | **0.95** | **0.95** |
| Deep and Wide Learning (50 epochs) | 0.72 | 0.70 | 0.72 | 0.67 |
| Gradient Boosting Classifier | 0.88 | 0.93 | 0.88 | 0.85 |

around 70%, is still less than LSTM for IoMT. With LSTM algoritm the results are 98% for all performance evaluation metrics.

While the CICIoMT2024 Dataset contains 19-classes, the last experiment was to see the results with 19-classes multi-classification. As expected the Logistic Regression showed a very poor performance from 12% up to 30% for the metrics, and the ANN with Adam was around 70%. Then we added more players to the experiments as Deep and Wide Learning (DWL) and Gradient Boosting Classifier (GBC) to experience some other DL algortihms capabilities. At the conclusion we experienced that DWL gives the results like the ANN around 70% but GBC has good performance as 88% accuracy, 93% for precision, 88% for recall, and 85% for F1-Score. Finally we tried LSTM and its metrics were around 95% which shows us that LSTM outperforms than the others.

The conclusion of these experiments shows that the LSTM algorithm consistently achieves high performance across all classification tasks, especially distinguishing in multi-class problems. Despite Dadkhah et al. who are the producers of the CICIoMT2024 dataset got accuracy as 73.3% for 19 classes; our experiment with a model consisting of 1 LSTM and 1 Dense layer attained an 95% accuracy.

### B. OPTIMIZING THE ARCHITECTURE OF THE PROPOSED LSTM MODEL

The results of the first experiment show that many machine learning algorithms can achieve excellent results for the 2-classes. But when it comes to multi-classification, their success substantially decreases. When the number of classes becomes 6 or 19, the performance of many algorithms and models degrades or fails to achieve acceptable results (see Table 6 and Table 7). Although the success rate of LSTM for 19 classes is around 95% and it is the highest obtained result, in this subsection, we tried to make this model more reliable and better for IoMT environments.

**TABLE 8.** Quantity of LSTM and dense layers and the results per model.

| Algorithms | Number of Units | Number of LSTM Layers | Number of Dense Layers | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|---|---|---|
| LSTM 1 (50 epochs) | 64 | 1 | 4 | 0.95 | 0.96 | 0.95 | 0.95 |
| LSTM 2 (50 epochs) | 64 | 2 | 3 | 0.71 | 0.69 | 0.71 | 0.67 |
| LSTM 3 (50 epochs) | 64 | 3 | 2 | 0.96 | 0.96 | 0.96 | 0.95 |
| LSTM 4 (50 epochs) | 64 | 4 | 1 | 0.71 | 0.69 | 0.71 | 0.66 |
| LSTM 5 (50 epochs) | 64 | 1 | 3 | 0.70 | 0.70 | 0.70 | 0.69 |
| LSTM 6 (50 epochs) | 64 | 3 | 1 | **0.98** | **0.98** | **0.98** | 0.97 |
| LSTM 7 (50 epochs) | 64 | 2 | 2 | **0.98** | **0.98** | **0.98** | **0.98** |
| LSTM 8 (50 epochs) | 64 | 1 | 2 | 0.70 | 0.73 | 0.70 | 0.67 |
| LSTM 9 (50 epochs) | 64 | 2 | 1 | 0.97 | **0.98** | 0.97 | 0.97 |
| LSTM 10 (50 epochs) | 128 | 1 | 4 | 0.75 | 0.74 | 0.81 | 0.69 |

In the first part of this experiment, we investigated the structure of the LSTM model by implementing and testing 10 different LSTM models with different numbers and types of layers. Table 8 represents the performance of the considered different LSTM architectures, each with varying numbers of LSTM units, layers, and dense layers. The evaluation metrics include accuracy, precision, recall, and F1-Score. LSTM models were set up with sequential, 50 epochs and 64 Units.

As seen in the Table 8, a total of 3 to 5 layers are employed with varying quantities of LSTM and Dense layers. Best results gained from LSTM 7 which comprises 2 LSTM and 2 Dense layers. LSTM 6, which consists of 3 LSTM layers and 1 Dense layer, is also a successful model. However, it performs slightly worse than LSTM 7, making LSTM 7 the preferred choice for further work. But before trying optimizers, we wanted to be sure about the number of units to be employed. So we utilized 128 units for LSTM 1 model, which had 95% rate of accuracy with 64 units. After the experiment, it was demonstrated that; increasing the unit number does not affect positively, on the contrary, the accuracy rate decreased down to 75%. Consequently the experiment continued with 64 units scenario.

The results of Table 8 show that the LSTM 7 model obtained the best performance in the experiment with 98% of accuracy, precision, recall, and F1-score. The results show that the proposed LSTM model with 2 LSTM layers followed by 2 Dense layers not only obtains the best results but also maintains a balanced performance for all metrics.

By achieving the best results from LSTM 7, the number of LSTM and Dense layers was set in the model. However, to confirm the optimal optimizer choice, alternatives were tested. For example, AdamW which improves [54] upon the Adam optimizer by decoupling weight decay [55] from the gradient updates, leads to better generalization performance. Nadam Optimizer is a variant of the Adam optimizer that incorporates Nesterov momentum, accelerating convergence during optimization [56]. To keep the learning rate from getting too low and to support the maintenance of an efficient rate of convergence, RMSprop Optimizer modifies the learning rate for each parameter using a moving average of the squared gradients. [57]. By Stochastic Gradient Descent (SGD) Optimizer, the model parameters are updated by using randomly selected batches, helping with faster convergence, but it does require careful learning rate tuning [58]. Adagrad Optimizer adapts the learning rate individually for each parameter, making it suitable for sparse data [59]. Adadelta Optimizer is an extension of Adagrad that dynamically adjusts learning rates without needing a fixed global learning rate that is overcoming Adagrad's diminishing learning rates [60].

To address class imbalance, weight balancing was tested, assigning greater importance to minority class samples during training to reduce imbalance effects. Another method used in this experiment to balance classes was SMOTE (Synthetic Minority Over-sampling Technique), which generates synthetic samples for the minority class by interpolating between existing samples [61]. Focal Loss, a modified cross-entropy loss that focuses learning on hard-to-classify examples by down-weighting well-classified especially useful for imbalanced datasets was also tested [62]. Additionally, Bidirectional LSTM, effective for sequential data, was tried [63]. Finally, Dropout, a regularization technique that randomly "drops" neurons during training to prevent overfitting and promote generalization, applied to the experiment [64].

Table 9 describes the performance of different variations of the LSTM algorithm tested under various hyperparameters, optimizers, and conditions such as learning rate and number of epochs. The evaluation metrics provided are again accuracy, precision, recall, and F1-score. As shown in Figure 3, the importance of the number of epochs is evident from the results of the experiments. Around 100 epochs, model gets stable and increasing number of epochs does not effect the output of the experiment.

All these Weight-balancing, Focal Loss, SMOTE, bi-directional LSTM, Stochastic Gradient Descent (SGD), Adagrad, AdamW, Nadam, RMSprop were applied to the LSTM 7 model. The very best result gained from the LSTM 7 with AdamW Optimizer model. Since the proposed model has 2 LSTM and 2 Dense layers, we named it as "L2D2".

### C. LIMITATIONS AND CONSTRAINTS

Our method demonstrates superiority over traditional machine learning (ML) approaches but still faces limitations for the reason of resource constraints of IoT devices. In our initial experiment, ML methods were shown to

**TABLE 9.** **The LSTM 7 model with different optimizers.**

| Algorithms | Learning Rate | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|---|
| Weight-balanced LSTM 7 (50 epochs) | 0.001 | 0.82 | 0.89 | 0.82 | 0.81 |
| Weight-balanced LSTM 7 (100 epochs) | 0.001 | 0.96 | **0.98** | 0.96 | 0.96 |
| LSTM 7 (100 epochs) | 0.001 | 0.89 | 0.86 | 0.89 | 0.86 |
| LSTM 7 with Focal Loss (50 epochs) | 0.001 | 0.91 | 0.87 | 0.91 | 0.88 |
| LSTM 7 with SMOTE (50 epochs) | 0.001 | 0.70 | 0.79 | 0.70 | 0.68 |
| LSTM 7 with bidirectional LSTM (50 epochs) | 0.001 | 0.78 | 0.79 | 0.78 | 0.74 |
| LSTM 7 with SGD optimizer (50 epochs) | 0.01 | 0.16 | 0.03 | 0.16 | 0.05 |
| LSTM 7 with Adagrad optimizer (50 epochs) | 0.01 | 0.71 | 0.69 | 0.71 | 0.66 |
| LSTM 7(50 epochs) | 0.0001 | 0.87 | 0.89 | 0.87 | 0.85 |
| LSTM 7 with AdamW optimizer (50 epochs) | 0.001 | 0.96 | 0.96 | 0.96 | 0.95 |
| LSTM 7 with Nadam optimizer (50 epochs) | 0.001 | 0.97 | 0.97 | 0.97 | 0.97 |
| LSTM 7 with RMSprop optimizer (50 epochs) | 0.001 | 0.55 | 0.52 | 0.55 | 0.49 |
| LSTM 7+ weight balancing +SMOTE + bidirectional LSTM+ drop out layers | 0.0001 | 0.65 | 0.72 | 0.65 | 0.61 |
| LSTM 7 + weight balancing (50 epochs) | 0.0001 | 0.68 | 0.73 | 0.68 | 0.66 |
| LSTM 7 with AdamW optimizer (50 epochs) | 0.0001 | 0.93 | 0.94 | 0.93 | 0.93 |
| LSTM 7 with Nadam optimizer (50 epochs) | 0.0001 | 0.95 | 0.95 | 0.95 | 0.94 |
| LSTM 7 with AdamW optimizer (100 epochs) | 0.0001 | **0.98** | **0.98** | **0.98** | **0.98** |

**TABLE 10.** **Comparison of studies that utilized CICIoMT2024 dataset for multi-classification of attacks against IoMT.**

| Related Works with CICIoMT2024 | Dataset | ML/DL Technique | Number of Classes | Accuracy |
|---|---|---|---|---|
| Towards Enhanced IoT Security: Advanced Anomaly Detection using Transformer Models (Sanchez et al.) [34] | CICIoMT2024 and IoT-23 | XGBoost with Transformer | 19-7 | 0.95 |
| CICIoMT2024: A benchmark dataset for multi-protocol security assessment in IoMT (Dadkhah et al.) [33] | CICIoMT2024 | Random Forest | 19 | 0.73 |
| L2D2 (Proposed Model) | CICIoMT2024 | LSTM | 19 | **0.98** |

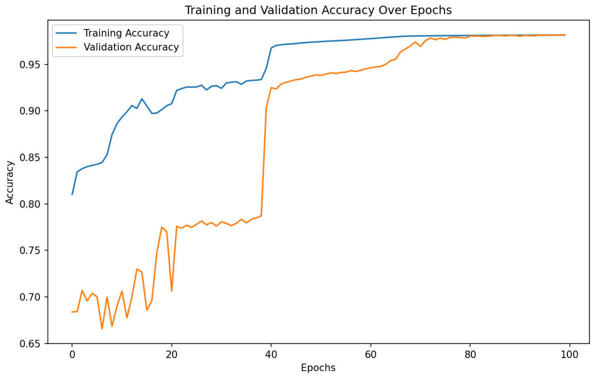


**FIGURE 3.** **The performance of the best model during training and validation.**

effectively detect the presence of intrusions. For simpler tasks like detecting only whether an attack is occurring in an IoMT environment, lightweight techniques such as Logistic Regression suffice due to their minimal resource requirements and efficiency.

However, when the objective shifts to accurately classifying the specific type of attack, our L2D2 model (which was derived from LSTM and additionally optimized by AdamW), exhibits superior performance compared to other techniques that are already mentioned in the Related Works Section. This performance comes at the cost of increased resource consumption. For resource-constrained IoMT devices, the L2D2 model is not recommended for simple binary detection of attacks, as it is unnecessarily computationally intensive.

Hybrid network architectures have promising successes for future works. Moreover, recent neural networks such as Graph Neural Networks (GNNs), also show significant accuracy rates at IoT systems, particularly those with complex dependencies and hierarchical structures, if the high resource usage is not an issue [65]. However, high memory usage and computational intensity remain significant challenges for deploying these models in resource-constrained IoMT environments. The new methods like Graph Neural Networks could get more accurate results than LSTMs, but we chose LSTM because of the trade-off between accuracy and resource consumption. Also deploying new neural networks on edge devices requires specialized hardware accelerators or optimized frameworks. While LSTMs are more readily deployable on edge devices using lightweight frameworks like TensorFlow Lite or PyTorch Mobile. Because of the constraints of IoMT, LSTM with the proposed model seems to be at the optimized level for multi-classification of cyberattacks.

As Table 10 demonstrates, the proposed model achieves the highest performance compared to other methods which were applied on CICIoMT2024 dataset. In 19-class multi-classification, our proposed model achieved the highest accuracy, recall, precision, and F1 scores, surpassing the performances reported by most research up to the preparation of this paper, based on our thorough search.

Consequently, in scenarios requiring multi-class classification for IoMT, the experiments clearly highlight the suitability of the L2D2 method, providing a balanced trade-off between accuracy and computational demand.

## VII. CONCLUSION

For IoMT environments, improving the security of the IDS sensors/systems is essential. The capabilities of these systems are increased by introducing new facilitation techniques and by employing artificial intelligence techniques. In this paper, we proposed a novel LSTM model optimized by AdamW and called it L2D2, to be used to enhance the security of intrusion detection systems when working in IoMT environments. To investigate the performance of the proposed model and reach the most efficient and accurate results, we tested many well-known machine learning and deep learning algorithms as well as optimizers. Due to the encouraging results obtained by the LSTM algorithm, a model that includes two LSTM and two Dense Layers with AdamW Optimizer was proposed and investigated to solve the intrusion detection problem using CICIoMT2024 dataset. Three different numbers of classifications were tested as 2-classes, 6-classes, and 19-classes. The proposed model has only four layers in total, yet is not a complex model, but achieved a result of 98% for 19-classes for accuracy, recall, F-1 Score, and precision which is the best result obtained for multi-classification in our experimental study and the literature. So this model can be used as a successful –but still not complex– model, for multi-classification of intrusion detections aiming cyberattacks, in IoMT environments.

## REFERENCES

[1] *That 'Internet of Things' Thing*. Accessed: Sep. 23, 2024. [Online]. Available: https://www.rfidjournal.com/expert-views/that-internet-of-things-thing/73881/

[2] F. Hussain, R. Hussain, S. A. Hassan, and E. Hossain, "Machine learning in IoT security: Current solutions and future challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1686–1721, 3rd Quart., 2020.

[3] J. Wang, M. K. Lim, C. Wang, and M.-L. Tseng, "The evolution of the Internet of Things (IoT) over the past 20 years," *Comput. Ind. Eng.*, vol. 155, May 2021, Art. no. 107174.

[4] B. L. Risteska Stojkoska and K. V. Trivodaliev, "A review of Internet of Things for smart home: Challenges and solutions," *J. Cleaner Prod.*, vol. 140, pp. 1454–1464, Jan. 2017.

[5] H. Xu, W. Yu, D. Griffith, and N. Golmie, "A survey on industrial Internet of Things: A cyber-physical systems perspective," *IEEE Access*, vol. 6, pp. 78238–78259, 2018.

[6] L. D. Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.

[7] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[8] A. Khanna and S. Kaur, "Evolution of Internet of Things (IoT) and its significant impact in the field of precision agriculture," *Comput. Electron. Agricult.*, vol. 157, pp. 218–231, Feb. 2019.

[9] A. Salam, "Internet of Things for environmental sustainability and climate change," in *Internet of Things for Sustainable Community Development: Wireless Communications, Sensing, and Systems*. Cham, Switzerland: Springer, 2024, pp. 33–69.

[10] Z. Ashfaq, A. Rafay, R. Mumtaz, S. M. H. Zaidi, H. Saleem, S. A. R. Zaidi, S. Mumtaz, and A. Haque, "A review of enabling technologies for Internet of Medical Things (IoMT) ecosystem," *Ain Shams Eng. J.*, vol. 13, no. 4, Jun. 2022, Art. no. 101660.

[11] A. Sharma, T. Choudhury, and P. Kumar, "Health monitoring & management using IoT devices in a cloud based framework," in *Proc. Int. Conf. Adv. Comput. Commun. Eng. (ICACCE)*, Jun. 2018, pp. 219–224.

[12] *Global Threat Report 2024*. Accessed: Sep. 23, 2024. [Online]. Available: https://go.crowdstrike.com/rs/281-OBQ-266/images/GlobalThreatReport2024.pdf/

[13] *CIC-IoMT 2024 Dataset*. Accessed: Sep. 23, 2024. [Online]. Available: https://www.unb.ca/cic/datasets/iomt-dataset-2024.html/

[14] D.-S. Zois, "Sequential decision-making in healthcare IoT: Real-time health monitoring, treatments and interventions," in *Proc. IEEE 3rd World Forum Internet Things (WF-IoT)*, Dec. 2016, pp. 24–29.

[15] A. Avinashiappan and B. Mayilsamy, "Internet of Medical Things: Security threats, security challenges, and potential solutions," in *Internet of Medical Things: Remote Healthcare Systems and Applications*. Springer, 2021, pp. 1–16.

[16] R. A. Jegatheswaran, I. J. Sakira, and N. A. A. Rahman, "A review on IoMT device vulnerabilities and countermeasures," *J. Phys., Conf. Ser.*, vol. 1712, no. 1, Dec. 2020, Art. no. 012020.

[17] J.-P.-A. Yaacoub, M. Noura, H. N. Noura, O. Salman, E. Yaacoub, R. Couturier, and A. Chehab, "Securing Internet of Medical Things systems: Limitations, issues and recommendations," *Future Gener. Comput. Syst.*, vol. 105, pp. 581–606, Apr. 2020.

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735—1780, 1997. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/9377276

[19] C. Anitha, C. V. Vivekanand, S. D. Lalitha, S. Boopathi, and R. Revathi, "Artificial intelligence driven security model for Internet of Medical Things (IoMT)," in *Proc. 3rd Int. Conf. Innov. Practices Technol. Manage. (ICIPTM)*, Feb. 2023, pp. 1–7.

[20] S. Ksibi, F. Jaidi, and A. Bouhoula, "IoMT security model based on machine learning and risk assessment techniques," in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, Jun. 2023, pp. 614–619.

[21] T. Saba, "Intrusion detection in smart city hospitals using ensemble classifiers," in *Proc. 13th Int. Conf. Develop. eSystems Eng. (DeSE)*, Dec. 2020, pp. 418–422.

[22] D. Alsalman, "A comparative study of anomaly detection techniques for IoT security using adaptive machine learning for IoT threats," *IEEE Access*, vol. 12, pp. 14719–14730, 2024.

[23] Z. Sun, G. An, Y. Yang, and Y. Liu, "Optimized machine learning enabled intrusion detection 2 system for Internet of Medical Things," *Franklin Open*, vol. 6, Mar. 2024, Art. no. 100056.

[24] G. Balhareth and M. Ilyas, "Optimized intrusion detection for IoMT networks with tree-based machine learning and filter-based feature selection," *Sensors*, vol. 24, no. 17, p. 5712, Sep. 2024.

[25] J. B. Awotunde, K. M. Abiodun, E. A. Adeniyi, S. O. Folorunso, and R. G. Jimoh, "A deep learning-based intrusion detection technique for a secured iomt system," in *Informatics and Intelligent Applications*, S. Misra, J. Oluranti, R. Damaševičius, and R. Maskeliunas, Eds., Cham, Switzerland: Springer, 2022, pp. 50–62.

[26] P. Kulshrestha and T. V. Vijay Kumar, "Machine learning based intrusion detection system for IoMT," *Int. J. Syst. Assurance Eng. Manage.*, vol. 15, no. 5, pp. 1802–1814, May 2024.

[27] S. Khan and A. Akhunzada, "A hybrid DL-driven intelligent SDN-enabled malware detection framework for Internet of Medical Things (IoMT)," *Comput. Commun.*, vol. 170, pp. 209–216, Mar. 2021.

[28] S. Liaqat, A. Akhunzada, F. S. Shaikh, A. Giannetsos, and M. A. Jan, "SDN orchestration to combat evolving cyber threats in Internet of Medical Things (IoMT)," *Comput. Commun.*, vol. 160, pp. 697–705, Jul. 2020.

[29] N. Faruqui, M. A. Yousuf, M. Whaiduzzaman, A. Azad, S. A. Alyami, P. Lió, M. A. Kabir, and M. A. Moni, "SafetyMed: A novel IoMT intrusion detection system using CNN-LSTM hybridization," *Electronics*, vol. 12, no. 17, p. 3541, Aug. 2023.

[30] Y. Otoum, Y. Wan, and A. Nayak, "Federated transfer learning-based IDS for the Internet of Medical Things (IoMT)," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2021, pp. 1–6.

[31] V. Ravi, T. D. Pham, and M. Alazab, "Deep learning-based network intrusion detection system for Internet of Medical Things," *IEEE Internet Things Mag.*, vol. 6, no. 2, pp. 50–54, Jun. 2023.

[32] 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.

[33] S. Dadkhah, E. C. P. Neto, R. Ferreira, R. C. Molokwu, S. Sadeghi, and A. A. Ghorbani, "CICIoMT2024: A benchmark dataset for multi-protocol security assessment in IoMT," *Internet Things*, vol. 28, Dec. 2024, Art. no. 101351.

[34] N. Sánchez, A. Calvo, S. Escuder, J. Escrig, J. Domenech, N. Ortiz, and S. Mhiri, "Towards enhanced IoT security: Advanced anomaly detection using transformer models," in *Proc. KDD 4th Workshop Artif. Intell.-Enabled Cybersecurity Anal.*, Barcelona, Spain, Aug. 2024. [Online]. Available: https://ai4cyber-kdd.com/KDD-AISec_files/Submission_7_final.pdf

[35] O. Priscilla Olawale and S. Ebadinezhad, "Cybersecurity anomaly detection: AI and Ethereum blockchain for a secure and tamperproof IoHT data management," *IEEE Access*, vol. 12, pp. 131605–131620, 2024.

[36] 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.

[37] A. A. Hady, A. Ghubaish, T. Salman, D. Unal, and R. Jain, "Intrusion detection system for healthcare systems using medical and network data: A comparison study," *IEEE Access*, vol. 8, pp. 106576–106584, 2020, doi: 10.1109/ACCESS.2020.3000421.

[38] *Sense-U Baby Monitor Description*. Accessed: Sep. 30, 2024. [Online]. Available: https://sense-u.com/en-as

[39] *SINGCALL SOS Button*. Accessed: Sep. 30, 2024. [Online]. Available: https://sense-u.com/en-as

[40] *Lookee Sleep Ring*. Accessed: Sep. 30, 2024. [Online]. Available: https://www.lookeetech.com/products/lookee-ring-sleep-monitor-w-vibrating-notification-for-sleep-apnea

[41] *Powerlabs HR Monitor Arm Band*. Accessed: Sep. 30, 2024. [Online]. Available: https://www.powr-labs.com/products/powr-labs

[42] (2024). *Wellue O2 Ring—3438*. Accessed: Sep. 29, 2024. [Online]. Available: https://getwellue.com/pages/o2ring-oxygen-monitor

[43] *COOSPO 808s Chest HR Monitor*. Accessed: Sep. 30, 2024. [Online]. Available: https://coospo.com/products/h808s-chest-strap-heart-rate-monitor

[44] J. Burgess, "Modern DDoS attacks and defences—Survey," 2022, *arXiv:2211.15404*.

[45] J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 2, pp. 39–53, Apr. 2004.

[46] D. Soni and A. Makwana, "A survey on MQTT: A protocol of Internet of Things (IoT)," in *Proc. Int. Conf. Telecommun. Power Anal. Comput. Technol. (ICTPACT)*, vol. 20, 2017, pp. 173–177.

[47] M. F. Hyder and M. A. Ismail, "Securing control and data planes from reconnaissance attacks using distributed shadow controllers, reactive and proactive approaches," *IEEE Access*, vol. 9, pp. 21881–21894, 2021.

[48] C. L. Abad and R. I. Bonilla, "An analysis on the schemes for detecting and preventing ARP cache poisoning attacks," in *Proc. 27th Int. Conf. Distrib. Comput. Syst. Workshops (ICDCSW)*, Jun. 2007, p. 60.

[49] F. Gont, "ICMP attacks against TCP," Internet Eng. Task Force (IETF), Tech. Rep., Jul. 2010. [Online]. Available: https://www.si6networks.com/files/publications/rfcs/rfc5927.txt

[50] W. M. Eddy, "Defenses against TCP SYN flooding attacks," *Internet Protocol J.*, vol. 9, no. 4, pp. 2–16, 2006.

[51] B. Harris and R. Hunt, "TCP/IP security threats and attack methods," *Comput. Commun.*, vol. 22, no. 10, pp. 885–897, Jun. 1999.

[52] D. R. Thomas, R. Clayton, and A. R. Beresford, "1000 days of UDP amplification DDoS attacks," in *Proc. APWG Symp. Electron. Crime Res. (eCrime)*, Apr. 2017, pp. 79–84.

[53] X. Zhu, H. Li, G. Xiong, and H. Song, "Automated qualitative rule extraction based on bidirectional long shortterm memory model," in *Proc. 29th Int. Workshop Intell. Comput. Eng. (EG-ICE)*, Aarhus, Denmark, Jul. 2022. [Online]. Available: https://orca.cardiff.ac.uk/id/eprint/149027/1/2022-EG-ICE-Paper%20second%20submission.pdf

[54] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.

[55] Z. Zhang, "Improved Adam optimizer for deep neural networks," in *Proc. IEEE/ACM 26th Int. Symp. Quality Service (IWQoS)*, Jun. 2018, pp. 1–2.

[56] T. Dozat, "Incorporating Nesterov momentum into Adam," in *Proc. ICLR*, 2016. [Online]. Available: https://openreview.net/pdf?id=OM0jvwB8jIp57ZJjtNEZ

[57] G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning lecture 6A overview of mini-batch gradient descent," *Cited*, vol. 14, no. 8, p. 2, 2012.

[58] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. 19th Int. Conf. Comput. Statist.*, Paris, France. Cham, Switzerland: Springer, Jan. 2010, pp. 177–186.

[59] J. C. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, no. 61, pp. 2121–2159, Feb. 2011.

[60] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, *arXiv:1212.5701*.

[61] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002.

[62] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," 2017, *arXiv:1708.02002*.

[63] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[64] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014.

[65] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Dec. 2008.

**GÖKHAN AKAR** received the bachelor's degree from the Electrical and Electronics Faculty, Istanbul Technical University, in 1996, and the M.Sc. degree from Istanbul Technical University's Energy Institute, focusing on energy sciences, with a thesis titled IT security for nuclear facilities, in 2016. He is currently pursuing the Ph.D. degree with the Engineering Faculty, Electrical, and Electronics Department, Marmara University.

He is an Electronics and Communication Engineer, with a research thesis on "Transmitting Smell Over Electromagnetic Waves. He is also experienced in project management and holds relevant certifications.

**SHAABAN SAHMOUD** received the bachelor's and master's degrees in computer engineering from the Islamic University of Gaza, Palestine, in 2006 and 2012, respectively, and the Ph.D. degree in computer engineering from Marmara University, Türkiye, in 2019. He is currently an Assistant Professor with the Computer Engineering Department, Fatih Sultan Mehmet Vakif University, İstanbul. His research interests include dynamic optimization, evolutionary algorithms, multi-objective optimization, Deep learning, computer vision, biometrics, and pattern recognition.

**MUSTAFA ONAT** (Member, IEEE) received the Ph.D. degree from Marmara University, İstanbul, Türkiye, in 2001. He is currently a Professor with the Electrical and Electronics Engineering Department, Faculty of Engineering, Marmara University.

**ÜNAL CAVUSOGLU** (Member, IEEE) received the Ph.D. degree from the Department of Software Engineering, Sakarya University. His research interests include information security, software engineering, information systems, communications, distributed systems management, and wireless ad hoc networks.

**EMMANUEL MALONDO** is currently pursuing the degree in computer engineering with the School of Engineering Faculty, CESI University. He has developed skills in machine learning, artificial intelligence, data structures, and embedded systems through coursework and hands-on projects and is actively engaged in research and collaborative projects in AI.