scientific reports

Check for updates

OPEN

A novel adaptive hybrid intrusion detection system with lightweight optimization for enhanced security in internet of medical things

Hassan Saeed[1,9], Mehwish Naseer[1,9], Afaf Rasool[2], Amjad Alsirhani[3], Faeiz Alserhani[4], Ghadah Naif Alwakid[5], Farhan Ullah[6], Hamad Naeem[7] & Yue Zhao[8]✉

The proliferation of Internet of Medical Things (IoMT) devices in e-Health systems has shown improved healthcare delivery but introduced severe cybersecurity vulnerabilities, including spoofing, denial-of-service, and data breaches. This study proposes leveraging artificial intelligence (AI) for an Intrusion Detection System (IDS) to secure IoMT environments and further assist in real-time threat detection and resilience of e-Health systems. This provided an improved model that implemented feature importance and ensemble learning, as well as contributed to developing a new hybrid system that uses the pre-trained Decision Tree (C4.5) model that incorporates a pre-trained Decision Tree (C4.5) model into the RL loop using Deep Q-Networks (DQN). This hybrid framework exploits the efficiency and low latency of pre-trained C4.5 for initial classification, and enables the ability of the system to learn dynamically from network interactions, adapt to changing patterns of attack, and improve detection performance over time. The general framework employs SMOTE to address class imbalance, while focal loss is utilized as an evaluation tool to analyze the classifiers' focus on hard-to-classify and minority class samples. It is important to note that the hybrid IDS has exhibited higher accuracy compared to Decision Tree - C4.5 with total rewards maximized, indicating the adaptive learning and stability in changing environments. The proposed model achieved an accuracy of 99.03% for binary classes, 98.55% for the five classes, and 99.56% for the 14-class experiment when using the initial classification with the Decision Tree (C4.5) model on the Canadian Institute for Cybersecurity, Internet of Medical Things-2024(CICIoMT2024) dataset. The initial classification and latency results are additionally compared to a few other lightweight classifiers such as Random Forest, XGBoost, and Simple Neural Networks. To bring adaptability and dynamic threat detection of Deep Reinforcement Learning (DRL) classifiers, the C4.5 model was integrated into a DQN framework to address evolving network threats over time. The hybrid model also persisted with improved performance, measuring 99.20% accuracy for the binary classes with CICIoMT2024 dataset. Proposed IDS was also evaluated for its generalization capability across heterogeneous datasets, i-e, WUSTL-EHMS, ECU-IoHT, DF_IOMT, and CICIOT23. The model consistently achieved high detection performance across the datasets and outperformed their respective previously achieved results with the C4.5 supervised classifier, which verified its robustness and flexibility across different IoMT contexts. The proposed hybrid IDS is therefore validated as a deployment-aware, lightweight, and adaptive framework capable of effective

intrusion detection in dynamic healthcare settings that are resource-limited and demand real-time responsiveness.

[1]Computer and Software Engineering Department, College of Electrical and Mechanical Engineering, National University of Sciences and Technology (NUST), 44080 Islamabad, Pakistan. [2]Rehman Medical Institute (RMI), 25000 Peshawar, Pakistan. [3]Department of Computer Science, College of Computer and Information Sciences, Jouf University, Sakaka 72388, Al Jouf, Saudi Arabia. [4]Department of Computer Engineering and Networks, College of Computer and Information Sciences, Jouf University, Sakaka 72388, Al Jouf, Saudi Arabia. [5]Department of Computer Science, College of Computer and Information Sciences, Jouf University, Sakaka, 72341 Al Jouf, Saudi Arabia. [6]Cybersecurity Center, Prince Mohammad Bin Fahd University, Khobar 31952, Saudi Arabia. [7]Department of Computer Science, College of Computer Sciences and Information Technology (CCSIT), King Faisal University, Al-Ahsa 31982, Saudi Arabia. [8]Department of Computer Science, College of Science, Mathematics and Technology, Wenzhou-Kean University, Wenzhou 325060, China. [9]Hassan Saeed and Mehwish Naseer contributed equally to this work. ✉email: yuezhao@kean.edu

The Internet of Medical Things (IoMT), which stems from IoT, is transforming healthcare through the interconnection of medical devices, sensors, and IT systems, facilitating remote monitoring, diagnostics, and personalized therapy[1]. The devices form adhoc networks used to transmit biomedical data and correspondingly form a multilayer architecture that may include sensors, network modules, storage units, and AI-based platforms[2]. Despite the benefits of IoMT networks, they also have security vulnerability issues due to the wide range of inter-connectivity and heterogeneous device setups.

Recent research has suggested several intrusion detection system (IDS) frameworks specifically designed for the constantly evolving context of IoMT (Internet of Medical Things) networks. These include hybrid architectures based on convolutional networks (CNN) and recurrent neural networks (RNNs)[3], ensemble models with meta-learning to improve adaptability and detection rate[4], and transformer models that offer explainability and robustness[5]. Other promising models in this regard include deep reinforcement learning that enabled adaptive IDS systems to be developed; such as a hybrid CNN-LSTM with Deep Q-Network (DQN) and Proximal Policy Optimization (PPO) which provided real-time threat detection across numerous datasets in IoMT[6]. Another noteworthy model combines federated learning and reinforcement agents, to secure IoMT while keeping data private[7].

The incorporation of the Internet of Medical Things (IoMT) into healthcare systems has enabled real-time patient monitoring and proactive treatment interventions. The capability of conventional security approaches (particularly static signature-based IDS) in addressing these challenges is limited, highlights the need for an adaptable and dynamic threat management paradigm. Due to resource-constrained environments and vulnerable communication protocols, IoMT devices have often become prime targets for cyber threats and unauthorized data breaches. This research investigates these security concerns through a structured three phased methodology.

In phase I, we conduct a preliminary classification with a supervised learning model based on the C4.5 decision tree algorithm, which we targeted for a fast, low-latency response to known threats. In this phase of detection accuracy, we utilize feature importance, ensemble methods, and class balancing with SMOTE. In phase II, the model utilizes a DQN reinforcement learning loop integrated with a C4.5 classifier, forming a hybrid IDS. The C4.5 classifier can make rapid classification that is interpretable, while the DQN agent is learning dynamic threat patterns over time[8,9]. This hybrid approach managed the ability to learn and adapt to changing threats, incorporating the ability to learn new policies through feedback from the environment, hence providing security from a real time perspective[10]. In phase III, we tested the model's generalization capabilities. The hybrid IDS was validated across multiple heterogeneous IoMT datasets: WUSTL-EHMS, ECU-IoHT, DF_IOMT, and CICIOT23. We observed that the hybrid IDS demonstrated consistently high performance across these datasets and confirmed the flexibility of the framework for deployment and real-world use in unfamiliar scenarios. The proposed IDS thus is a holistic approach encompassing low latency supervised detection, adaptability of Deep Reinforcement Learning, and proven generalization across diverse threat landscapes.

Contributions of the current study

The contributions of this study can be summarized as follows:

1. A deployment-aware and lightweight IDS designed primarily for the Internet of Medical Things (IoMT) environments, motivated by the need for high detection accuracy at low inference latencies.
2. We introduce a hybrid IDS that merges a pre-trained Decision Tree (C4.5) with a DQN for building adaptability/ self-learning capability in the model, to provide enhanced detection of evolving threats.
3. Demonstrated how this hybrid IDS can efficiently capitalize on supervised learning for initial classification while gaining the continuous adaptation and reward optimization of reinforcement learning to produce improvement in long-term performance and resilience.

The structure of this paper is follows, Section Related Work reviews related work and discusses security aspects related to IDS for IoMT. Section Dataset describes the dataset and its composition,Proposed Methodology explains proposed methodology, including model training and design aspects. Section Results presents results and analyses. Section Discussion includes discussion on obtained results. Section Limitations explains the limitations observed in executing the proposed pipeline. Finally, Section Conclusion and Future Work concludes with key findings, contributions and future directions.

Related work

Security challenges in IoMT networks

IoMT devices are uniquely at risk due to these limitations on resources and the lack of a security baseline for organizations using IoMT devices[11].Threats commonly affecting IoMT include Denial of Service (DoS), Spoofing, Recon, Man-in-the-Middle (MiTM) attacks, Distributed Denial of Service (DDoS), and Brute Force. This contributes to problems such as weak authentication, inadequate encryption practices, and limited firmware updates, thus exposing IoMT systems to data breaches. These threats affect various components across the IoMT architecture from each perception layer to each application layer. The varying nature of devices and the sensitive nature of data increase the difficulty of developing a standard security approach. In the event of a security breach, the trust can deteriorate, which may disincentives technology adoption[12]. Privacy concerns may get escalated, assuming the nature of the data being collected, i.e., sleeping habits, diet, etc.

Intrusion detection systems (IDS) for IoMT

Intrusion detection systems (IDS) protect networks from compromise and are especially important for IoMT networks. IDS relies on two general approaches to monitor networks for anomalies and known threats: signature-based detection and anomaly-based detection[13]. Signature-based detection systems are very accurate when detecting known threats, but perform poorly with unknown threats. Anomaly-based detection utilizes algorithms based on machine learning (ML) to identify unauthorized attacks, but it has a major issue with false positives. Hybrid intrusion detection systems have the potential to utilize both signature-based and anomaly-based detection to combine both types of systems. IDSs are essential components of network defenses for IoMT networks, but existing intrusion detection solutions are often less effective in real-time due to dynamic traffic and high false positive rates, which is particularly problematic in a medical context[14].

Advanced AI/ML-based IDS solutions for IoMT

AI/ML-based IDSs can recognize and respond to threats in real time and adapt to new threats. Several of these algorithms such as Support Vector Machines (SVM), Random Forests (RF), Convolutional Neural Networks (CNN), and Long Short Term Memory (LSTM) have shown a high degree of detection accuracy, whereas the ensemble based models (e.g. Xtreme Gradient Boosting; ERT) exhibited better performance compared to the individual classifiers[15].

Existing hybrid IDS models

Recently developed hybrid IDS models attempted to combine static classifiers with reinforcement learning in order to facilitate adaptability and improve detection performance. For instance, Shaikh et al. (2025) presented a hybrid CNN-LSTM design with Deep Q-Network (DQN) for real-time detection of threats specific to IoMT environments and recorded decent detection rates, albeit with comparatively high inference latency[6]. presented a hybrid approach referred to as HDRL-IDS, that integrated Deep Deterministic Policy Gradient (DDPG) and deep learning based strategies for intrusion detection in 5G-enabled medical networks; however, they identified interpretability and model complexity issues[16]. Other architectures such as MLP + PPO offer significant adaptability, but suffer from over-sensitivity to parameter tuning and lack of transparency in the decision processes.

These problems are further compounded by the hyper-parameter tuning of DQN and intrinsic lack of interpretability across neural networks. Due to these issues we seek to create C4.5-DQN hybrid, which integrates the explainability and low-latency of C4.5 with low-complexity dynamic policy learning that DQN affords, while aiming to remain computationally lightweight for IoMT settings. Meta-Learning has demonstrated the capacity to make and adjust the classifiers' weights dynamically, to explore ensemble learning capabilities. However, they are computationally intensive and make the classification time highly unsuitable for IoMT specifications[4], therefore requiring the use of lightweight models and edge/fog computing. Having realistic datasets, such as CICIoMT2024, is critical to further build upon the existing work and to develop models and solutions that can be generalized. Table 1 summarizes key strengths and limitations of common ML based models used in IDS, particularly in the IoMT context.

Existing and emerging hybrid IDS models

Recent hybrid IDS models are designed to combine static classifiers with reinforcement learning to facilitate adaptability and improve detection performance. For instance, researchers proposed a new hybrid CNN-LSTM

Technique	Accuracy	Latency	Adaptability	Findings
Random Forest	High	Medium	Medium	Robust and generalizes well, but highly memory extensive[17]
SVM	Linearly Separable (High)	Low	Low	Useful with simple attack patterns, but poor with evolving IoMT traffic[18]
CNN	High	Medium	Medium	Good for spatial features, lacks temporal insight[19]
LSTM	Very High	High	High	Learns temporal attack patterns, but slow inference[20]
XGBoost/ Ensemble	Very High	Low	Medium	Strong on generalization aspects, but lacks interpretability[21]
Transformer/ XAI	High	Medium	High	Improves modularity[21]
DQN (RL)	Very High	Medium	Very High	Highly adaptable to zero-day attacks[16]

Table 1. Comparison of AI/ML-based IDS techniques in IoMT Environment.

architecture with Deep Q-Network (DQN) for real-time identification of IoMT infrastructures specific threats, returning reasonable detection rates, but had high inference latency[6]. A hybrid Deep Reinforcement Learning model called HDRL-IDS was introduced by merging Deep Deterministic Policy Gradient (DDPG) and deep learning based approaches for intrusion detection on 5G-enabled medical networks. interpretability and model complexity were identified as key actors in achieving effective IDS[16]. Other architectures such as MLP + PPO offer significant adaptability but suffer from lack of transparency issues in the decision processes. These problems are further compounded by the hyper-parameter tuning of DQN and intrinsic lack of interpretability across neural networks.

Some hybrid IDS models are designed to combine static classifiers with reinforcement learning to facilitate adaptability and improve detection performance. For instance, researchers proposed a new hybrid CNN-LSTM architecture with Deep Q-Network (DQN) for real-time identification of IoMT infrastructures specific threats, returning reasonable detection rates, but had high inference latency. A hybrid Deep Reinforcement Learning model called HDRL-IDS was introduced by merging Deep Deterministic Policy Gradient (DDPG) and deep learning based approaches for intrusion detection on 5G-enabled medical networks. Other architectures such as MLP + PPO offer significant adaptability but suffer from lack of transparency issues in the decision processes. These problems are further compounded by the hyper-parameter tuning of DQN and intrinsic lack of interpretability across neural networks.

Recently, much of the work has evolved in the area of intrusion detection systems (IDS) for Internet of Medical Things (IoMT) environments. For example, a recent study[22] proposed a multi-attention Deep Convolutional Recurrent Neural Network (DeepCRNN)-based cyberattack detector that was efficient for devices in smart IoMT environments. Another work[23] presented a hybrid deep learning method incorporating an Autoencoder (AE) and Long Short-Term Memory (LSTM) network for intrusion detection under imbalanced Industrial Internet of Things (IIoT) traffic, where the attack samples are few and far between compared to benign traffic. Similarly, a survey[24] examined Internet of Things (IoT) security models inclusive of a Firefly Algorithm (FA)-optimized feature selection with LSTM for intrusion detection, while other researchers[25,26] explored genetic algorithm (GA)-driven and metaheuristic-optimized LSTM models for intrusion detection based on IoT-edge interfaces. In the Cyber-Physical Systems—Industrial Internet of Things (CPS–IIoT) domain, attention-based explainable privacy-preserving architectural designs have been implemented in recent work[27,28], recognizing both resilience and interpretability. Due to these issues, we intend to create a hybrid model that integrates the low-latency of a suitable static classifier with the dynamic policy learning capabilities of DQN, while remaining computationally lightweight for IoMT settings.

Emerging paradigms for IoMT security

Edge or fog computing reduces latency while improving privacy by leveraging local processing of data. Blockchain delivers transparency, immutability, and distributed authentication; however, it is not suitable for high data throughput. Federated Learning (FL) allows collaborators to train algorithms without sharing raw data and reduces the communication overhead and latency of sharing updates[29].Each concept addresses its own set of challenges, and when combined, they create powerful hybrid security architectures. Explainable AI (XAI) has become both a necessity and a focus area in the healthcare space, as clinician acceptance and adoption of AI models rely on their transparency. Ensemble learning architectures can produce higher accuracies through model diversity, but can make interpretability process ambiguous due to decision outcomes being spread across multiple learners[30,31]. Post hoc methods such as SHAP and LIME aim to counteract this ambiguity, though they may produce inconsistent explainability across sub-models of the ensemble architectures[32]. Deep reinforcement learning (DRL) adds even more opacity given the deep policy networks and must be interrogated sequentially, which will not always be available unless otherwise integrated into formal verification health decision frameworks[33]. Given these limitations, some studies have examined hybrid approaches for ensemble learning, such as hierarchical RL with built-in interpretability layer, to fulfill an acceptable and accurate interpretation of the AI model that can also be utilized for wisdom of IoMT deployments. Features like flow duration and biometric information serve as valuable sources for anomaly intrusion detection systems (IDS) models. Models that offer greater transparency are more likely to be accepted and demonstrate accountability, both foundational principles for clinical use[34].

Research gaps and opportunities

The Internet of Medical Things provides healthcare systems with a pathway for e-health services. However, the high risk nature of IoMT presents significant security problems in the healthcare space, in particular, the absence of lightweight options capable of working in a resource-constrained IoMT environment, poor adaptability to changing threats, and insufficient evaluation across diverse real-world datasets[35]. Most existing approaches to IDS either incur high computational costs or are unable to adjust dynamically to evolving attack surfaces. In addition, generalizability across different operational contexts is often ignored, which further limits real-world deployment. These problems emphasize the need to develop IDS frameworks that are both computationally efficient and enable the system to generalize robustly across unseen contexts[36,37]. This research sets out to address these significant limitations by developing a hybrid, lightweight, and generalizable intrusion detection system suitable for the situated perspective of IoMT environments.

Dataset and preprocessing

A comprehensive overview of diverse multiple attacks encompassing different kinds of attacks is used for validation of the proposed model. Each dataset selected shows various aspects of IoT/ IoMT environments, such as varying network characteristics, numerous features related to various network layers, thereby ensuring a complete assessment of the model's performance across a broad spectrum of different kinds of attacks.

Dataset name	Domain	Characteristics	Attack types
Binary Classification	General	2-class problem (Normal vs Attack); Simplified classification	Benign, Attack
Categorical Classification	General	5 attack categories for coarse-grained labeling; better than binary for IDS training	Benign, Spoofing, Recon, DoS, DDoS
Multiclass Classification	General	Fine-grained detection across 14 types; supports detailed forensic analysis	Benign, ARP Spoofing, Ping Sweep, Recon VulScan, OS Scan, Port Scan, DoS TCP, DoS ICMP, DoS SYN, DoS UDP, DDoS SYN, DDoS TCP, DDoS ICMP, DDoS UDP

Table 2. Attack types for binary, categorical, and multiclass classification problems.

Dataset name	Domain	Features	# Samples	Characteristics	Attack types
CICIoMT2024	IoMT	45	∼9,000,000	40 devices (25 real + 15 simulated); Wi-Fi, MQTT, Bluetooth; realistic device profiling	DDoS, DoS, Recon, MQTT-based, Spoofing (18 attacks—14 evaluated for this study)
WUSTL-EHMS	IoMT	44	16,318	Real-time EHMS testbed	Man-in-the-Middle (Spoofing, Data Injection, etc.)
CICIoT2023	IoT	47	7,332,065	105 real IoT devices; 33 attack types in 7 attack categories	DDoS, DoS, Recon, Web-based, Brute Force, Spoofing, Botnets (Mirai)
ECU-IoHT	IoHT	6	111,207	Clinical testbed aiming at IoHT components; publicly available dataset for IDS generalization	Recon, DoS, Spoofing
DF_IoMT	IoMT	50	188,694	Best suited for cross-domain generalization	DoS, DDoS, Exfiltration, Stealth Scan, Port Scan, TCP reset floods, Fast/short malicious sessions

Table 3. Datasets used for validating generalization of the proposed IDS.

CICIoMT2024 Dataset: The rationale for selecting the CICIoMT2024 dataset for this work, instead of some other dataset, is that the dataset is comprehensive, as it was specifically designed to assess security solutions in the Internet of Medical Things (IoMT) space. It includes 14 different cyberattack scenarios targeted against 40 IoMT devices, incorporating both actual and emulated hardware that simulated a realistic threat event. These attacks exist in various classes, namely binary (2-class), Categorical (5-class), and Multiclass (14-class) as given in Table 2.

Additionally, the CICIoMT2024 dataset is ideal for machine learning research, being constructed to help support and measure the performance of machine learning-based intrusion detection systems[11].

WUSTL-EHMS: This dataset is produced from an enriched healthcare monitoring system comprising a total of 44 features, including 35 features of network flow and 8 features of biometric patient data. This dataset is primarily concerned with Man-in-the-Middle (MiTM) At-tacks like spoofing or data injection intended to corrupt medical telemetry. This dataset is especially useful for evaluating the performance of IDS under stealthy attacks[38].

ECU-IoHT. Designed to address the lack of publicly available IoHT attack datasets, this da-taset simulates cyberattacks against healthcare Internet of Things (IoT) infrastructure, including body wearables, infusion pumps, etc. This dataset is well suited for generalization testing of IDS frameworks in medical conditions. However, it lacks any comprehensive description of features in the literature, so special effort was made to understand and assess the features and information it conveys[39].

DF_IoMT. Feature descriptions are specific to each dataset, but DF_IoMT is a unique dataset as it provides new testing scenarios based on distinct devices, distinct forms of communication, and different forms of attack. An inclusion of this dataset provides diversity to the datasets used in this study for benchmarking cross-dataset generalization in the IoMT context[40].

CICIoT2023: This dataset covers 33 types of attacks and is divided into 7 categories, i-e, DoS, DDoS, Brute Force, Reconnaissance, Web-based, Spoofing, and botnet attacks. CICIoT2023 is one of the most extensive, feature-rich, publicly available datasets on the topic of IoT security. Collected data from 105 real IoT devices, which can expand the scope of research by making IDS from IoMT to generic IoT systems[41]. A summary of datasets used and corresponding at-tacks covered is given in Table 3.

Proposed methodology

The proposed methodology employs a systematic multi-stage framework for threat detection and adaptive primary response to threats, consisting of data processing and supervised classification (Phase I), subsequently layered with Deep Q— Network Architecture (Deep Reinforcement Learning) to learn a dynamic policy (Phase II) and then finally thoroughly evaluated to measure generalization on "real-world" datasets (Phase III). While there are three interdependent parts of the process, the objective is to combine and maximize accuracy, adaptation, and readiness based on characterization of the threats and the system environment. The overall methodology is depicted in Fig. 1 which shows the sequential application of all parts of the methodology.

The classification stage provides accurate anomaly detection, which feeds into the reinforcement agent for accurate, real-time decisions. The generalization stage is intentionally positioned as a demonstration of system validation. Earlier results from the 'Classification' and 'Reinforcement' parts would not be credible without

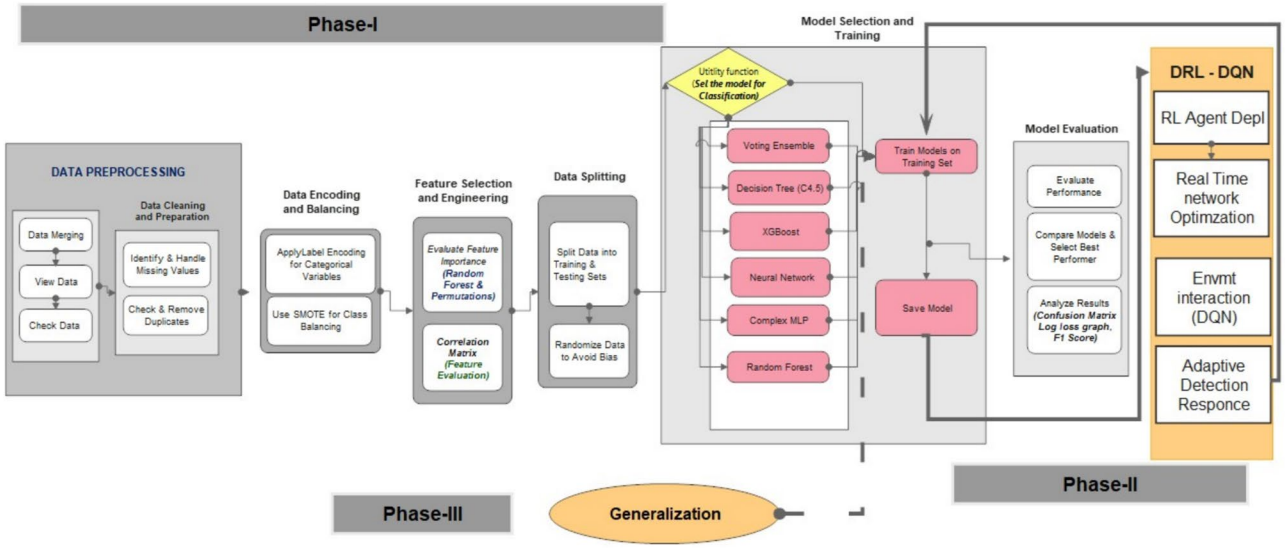


Fig. 1. Execution work flow.

showing consistent performance on various datasets. Consequently, Phase III demonstrates the transferability of the system, as well as the practical use of the system.

Phase I: Initial supervised classification

The first phase of the methodology pertains to building a high-performing classification model to accomplish maximum accuracy with minimum latency. It involves several critical sub-stages.

Data preprocessing

Data preprocessing is a necessary step to maximize machine learning outcomes and optimize processes to account for noise, missing values, and duplication of records. Data preprocessing minimizes noise, thus maximizing the model's ability to generalize on meaningful patterns in datasets, especially regarding high-dimensional or sensor-based datasets, where raw inconsistencies lead to output even poorer than expected[8]. Without prior preprocessing of data, there is a high likelihood of developing a biased model, which overall affects the accuracy of machine learning classifiers.

Data encoding and balancing

Class imbalance is one of the common issues in supervised ML-based anomaly detection, where rare or critical classes are ignored during model development. SMOTE solves this problem by generating new synthetic samples from the minority classes. In general, using SMOTE-generated minority samples greatly improves the Recall and F1 score without overfitting[42]. Improved and enhanced versions of SMOTE further optimize sample generation (close to original samples in the data), hence contributing to overall accuracy and robustness in the detection of rare events in real-world datasets[43].

Feature selection and engineering

Not all features in high-dimensional network traffic datasets aid in classification, and irrelevant or highly correlated features can increase computational costs and reduce model performance. Therefore, 10 features are carefully selected for training by considering prior domain knowledge, similar studies in the literature, and empirical feature importance to understand which features are most relevant. This feature engineering step improves the interpretability of the model and reduces processing time due to lower dimensionality. Feature importance was assessed based on two different methodologies, one using a Random Forest classifier and the second using permutation importance. Random Forest assigns greater importance to features based on the improvement in node purity during tree construction, while permutation importance measures a feature's contribution by assessing the effect on performance when its values are randomly shuffled[44,45]. The 10 x features selected are given as per Fig. 2.

Relevance of feature mapping to network layers for IDS development

The success of an intrusion detection system (IDS) in identifying threats depends on the ability to recognize features of relevant network traffic and understand their significance at particular points in the stack of network layers. This section correlates 'ten vital selected features' of network traffic to their corresponding layers in both the OSI and TCP/IP models, as well as describes how they can be used to detect a multitude of network intrusions[46–48]. An IDS can take advantage of certain features like 'Header_Length', 'rst_count', and 'Tot size' to give insight into certain layers of the network stack, such as Network and Transport Layers. These features can help identify threats such as TCP reset attacks, malformed packets, and unusual traffic. However, concentrating

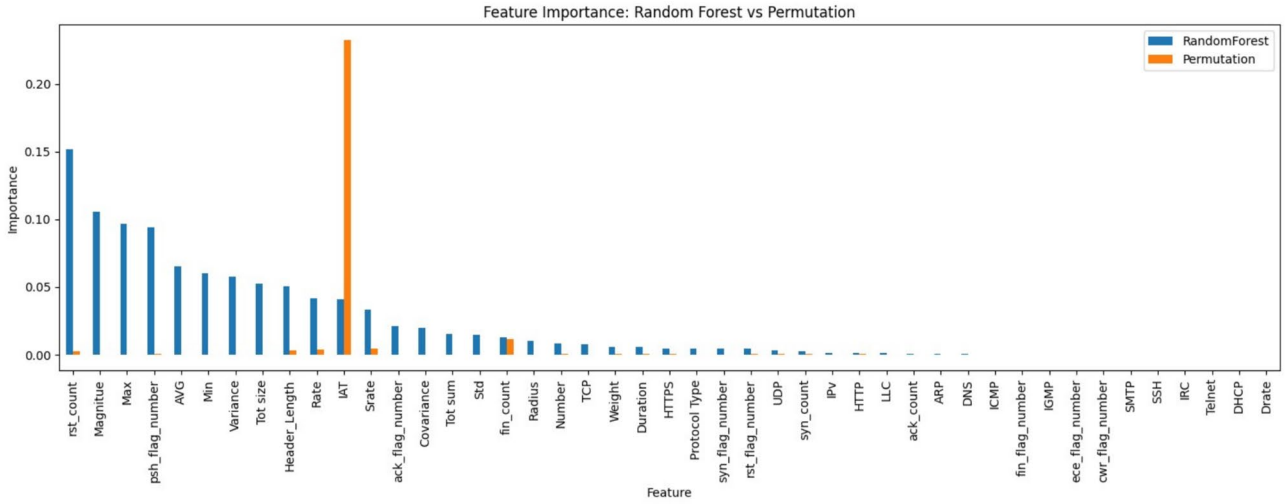


Fig. 2. Features selected after feature selection step.

Feature	Description	TCP/IP layers	OSI layers	Relevance to threats/malicious info
Srate	Source-to-destination packet rate (packets/sec).	Internet, Transport	Network (L3), Transport (L4)	Attacks like DoS, DDoS and port-scanning via anomaly in packet rates[46]
Rate	Flow rate including both directions.	Internet, Transport	Data Link (L2), Network (L3), Transport (L4)	Flagging abnormal data transfer, e.g., data exfiltration[49]
IAT	Inter-arrival time between packets.	Link, Internet, Transport	Physical (L1) to Transport (L4)	Detecting timing-related anomalies e.g., slow scans or stealthy floods[48,49]
Header_Length	Total length of packet headers.	Internet, Transport	Network (L3), Transport (L4)	Identifies malformed/altered headers which can indicate evasiveness or misuse[49,53]
rst_count	Number of TCP reset flags.	Transport	Transport (L4)	Detects port scans and TCP reset floods using RST flag spikes[51,54]
Duration	Total time duration of a flow.	Internet, Transport	Network (L3), Transport (L4)	Fast attacks like exfiltration activity detected through flow duration[46,47]
Protocol Type	Type of protocol used (e.g., TCP, UDP, ICMP).	Internet, Transport	Network (L3), Transport (L4), Application (L7)	Critical for identifying attacks and unauthorized protocols[51]
Max	Max value of selected attributes (e.g., packet size, duration).	Link to Application	Data Link (L2) to Application (L7)	Flags extreme outliers like too large a packet or excess delay[47]
Tot size	Total size of all packets in a flow.	Internet, Transport	Physical (L1) to Transport (L4)	Marker for large transfer, exfiltration, downloads or DoS volume[49]
Total sum	Sum of selected features to represent total activity.	All TCP/IP Layers	All OSI Layers (L1–L7)	Builds overall behavior profiles; detects significant behavioral shifts[54]

Table 4. Relevance of selected features to network layers.

on just a few layers can create problem areas of blindness, especially given that modern cyber attacks frequently happen across multiple layers from Application Layer exploitation to Transport Layer data exfiltration, and often take advantage of lower layer characteristics to hide from detection. Given the large data set, if the IDS cannot evaluate many layers at once, it will typically not see the full context of the attack[49,50] .

Correlation of data across all layers of the network stack is essential for effective threat detection. Features such as 'IAT' contain timing anomalies that are observable at the Physical/Link Layer, while 'Protocol Type' provides information about the Application Layer activity. Aggregated features such as 'Tot sum' can produce complete behavioral profiles that can reveal multi-stage attacks, which would otherwise remain hidden in associated data. Correspondingly, both the OSI and TCP/IP models can be utilized to organize the features, but it is essential to make certain that insights at different layers use data from each layer to provide comprehensive and accurate IDS capabilities[51,52]. Table 4 gives detailed relevance of selected features to network layers.

Train-test splitting

In efforts to objectively assess the performance of the models, the dataset was split into two sets, one for training and the other for testing, with a split of 80% training: 20% testing. Stratified sampling was employed to create the partitions with proportionate class distributions. This approach is especially important in imbalanced classification because the stratified sampling will account for the minority class instances being included in both training and testing sets[55].

Model training & selection
A comparative analysis employing six different machine learning classifiers in the quest to identify one with the best performing parameters of accuracy and latency time. This comparative analysis is a first step towards achieving the above performance representations. The thoughtful choice of such a wide-ranging set of machine learning models, from easily interpretable decision trees to the more complex and powerful ensemble methods and complicated neural networks, cemented a thorough comparative analysis.

Voting ensemble
This combines the predictions of several individual base models. The use of perspective coming from many diverse perspectives should improve the overall robustness and usually achieves better accuracy than any individual component's predictions in a Voting Ensemble[56].

Decision tree (C4.5)
The C4.5 algorithm for decision trees is an extension of the ID3 algorithm. It is a well-used and understood algorithm that can easily accommodate continuous and discrete attributes. It can also accommodate missing values and can prune the decision tree to limit overfitting[57,58].

XGBoost: Xtreme Gradient Boosting (XGBoost) is mentioned here primarily as a powerful, distributed, open-source machine learning library. It has the capability and performance to solve even large datasets. Moreover, XGBoost was designed with regularization as a built-in component[59].

Neural network/complex MLP
Multi-Layer Perceptrons (MLPs) are a type of artificial neural network that consists of multiple hidden layers and non-linear activation functions (Relu in this case), allowing them to learn and model complex, non-linear relationships in the data[60]. There are two versions of Neural Networks (NN) implemented in this study, one is a simple NN (Simple MLP with one hidden layer), and the second Complex NN (NN with three hidden layers). The existence of these two options indicates a belief in a highly complex relationship in the underlying data that may not be captured using other simpler models.

Random Forest
Random Forest is an ensemble learning technique that builds a set of decision trees while training, then outputs the mode of possible classes (for classification) or mean prediction (for regression). It uses bagging (bootstrapped aggregators) and also features randomness to help reduce the correlation between individual trees, leading to better robustness against overfitting, as well as to be able to work with more types of data[61,62].

One of the primary reasons for the selection of these classifiers is their previously documented success in intrusion detection and represent a diverse range of ML models[63]. All models were trained with SMOTE balanced training and tuned using either default or experimentally inferred hyper-parameters to mitigate the risk of overfitting the model.

Model evaluation
To evaluate the performance of the models accuracy, F1 score, Precision, Recall, and Log Loss Graph are used. These metrics were calculated for each of the three types of classification, i-e, binary, categorical & multi-class using a single evaluation function to ensure consistency of performance assessment when making comparisons across classifiers. In addition, training and inference times were also captured to assess computational viability in an IoMT context. Furthermore, we qualitatively compared models based on visual summaries such as confusion matrices, log loss plots, and Inference Time for binary classification.

Evaluation of inference time
Inference time was evaluated and compared for each model to assess their suitability in real-time deployment in the case of an IoMT. C4.5 and the Simple Neural Network achieved the shortest average inference times (9.64 ms and 25.24 ms, respectively), representing a best-case scenario for latency-sensitive environments. In contrast, the Voting Ensemble and Complex Neural Network required significantly longer inference times (exceeding 600 ms), limiting their real-time operational potential. The modelling results suggest the significance of balancing detection performance to computational expense when designing IDS within the resource constraints of healthcare systems.

Cross-validation and generalization
To further test that the models generalize to unseen data, k-fold cross-validation was employed, where k would equal 5. For each iterated test, the data set was equally partitioned into k parts, with k-1 parts for training and the last part for testing. The test was iterated k times, and the estimation of the model performance was achieved by averaging performance over the k runs[55]. Cross-validation has the effect of reducing bias in the estimation of model performance, and it allows for the detection of model overfitting, especially in cases where minority samples are much lower based on the total sample from the dataset. This was a particularly useful step when dealing with models that were based on neural networks, for example, since neural networks have an issue of overfitting data, especially related to sample quantity and imbalance across data levels in the dataset.

Phase II - Deep Q-Network (DQN) for Adaptive Detection and Response

The fundamental part of this proposed hybrid system is using a pre-trained C4.5 model in a Reinforcement Learning loop of Deep Q-Networks (DQN). This is very critical for the system to have adaptive capabilities.

Component	Definition	Rationale/connection to C4.5
State (S)	These are network traffic features (e.g., Srate, Rate, IAT, Header_Length, rst_count, Duration, Protocol Type, Max, Tot size, Tot sum) from CICIoMT2024, possibly with C4.5 outputs.	These features are selected for IoMT intrusion detection relevance[9]. C4.5 outputs help the DQN agent by adding informed cues or simplifying the state space.
Action (A)	Action space includes discrete values for each class label, ranging from 0 (Benign) to 13 (some specific kind of attack), making a total of 14 actions.	Reflects the IDS task of classifying traffic—binary or multi-class actions.
Reward Function (R)	Positive reward for correct classifications; penalty for misclassifications (False Positives/Negatives).	Encourages accurate detection while penalizing critical errors—important for safety and reliability in IoMT.
Policy (P)	Agent's behavior strategy.	Represents the IDS detection rule: how the agent classifies traffic given a particular state.
Agent	The one who takes action—the decision maker who learns iteratively and improves.	The IDS engine learns and detects threats by optimizing the policy given by the agent.
Environment	External system that an agent interacts with.	Represents the network traffic environment, including both benign and malicious patterns.

Table 5. Reinforcement learning components and their connection to C4.5.

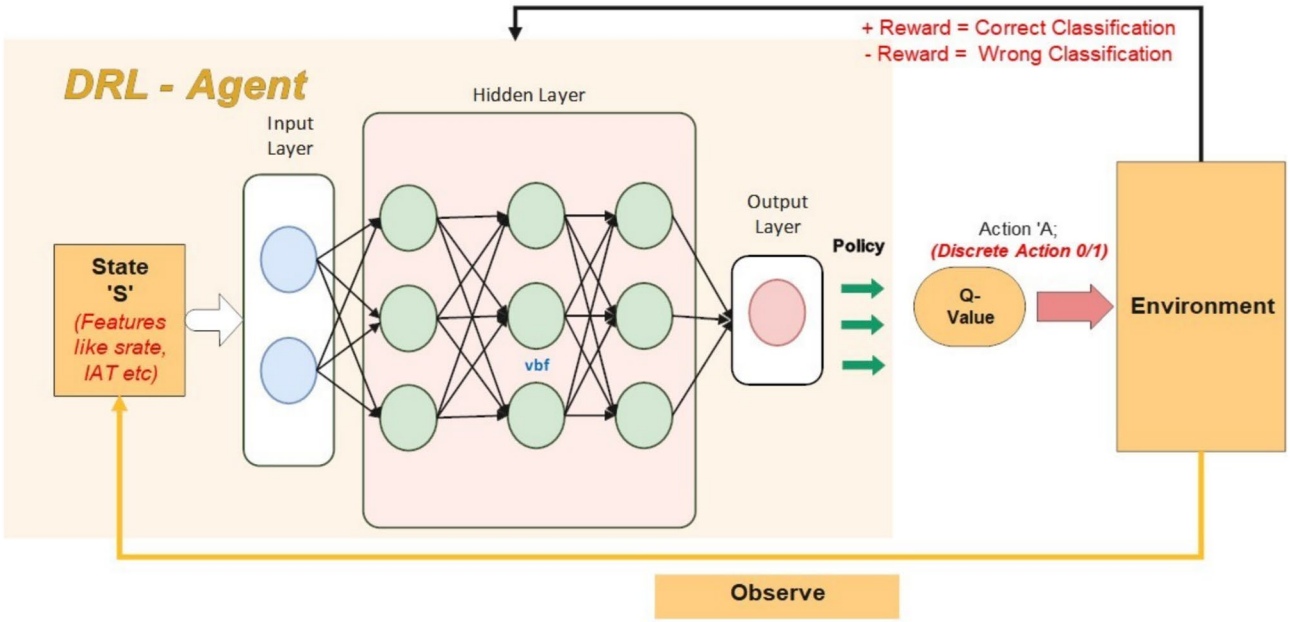


Fig. 3. Deep Q-network architecture.

Before explaining the hybrid framework in detail, it is vital to provide a formal definition of the DQN components for clarity and reproduction. The environment components are represented in Table 5:

The working of Deep Q-network is shown in Fig. 3. It explains the role of each of these components in training the model using DQN.

The Deep Q-Network (DQN) algorithm uses a neural network to approximate Q-values, the value represent the expected cumulative future reward for taking a specific action in a state. Key features of DQN include the use of experience replay (storing past interactions, i-e, state, action, reward, next state) in a replay buffer and sampled randomly during training. This technique will serve to limit correlations between successive entries, hence the stabilizing learning process[64]. DQN agent training parameters and architecture settings used for adaptive learning within the IoMT environment are given as per Table 6.For action selection, the epsilon-greedy exploration strategy was used. The initial exploration rate ("ϵ") from the agent's experience was randomly set to 1.0 and then decayed by 0.995 each episode until it reached minimum of 0.01. This decay schedule allows the agent to explore more of the different actions at the start, and then focus more on using the best behavior it had learned in the later episodes. This adaptive investigation is vital for the agent to constantly learn from the dynamic IoMT environment and identify evolving attack patterns[65].

The outputs of the pre-trained C4.5, such as predicted class labels, confidence scores, or even the feature vector after processing through C4.5's internal nodes, are used as part of the state representation for the DQN agent. The benefit of this hybrid learning through an RL agent is that it provides a more informed starting point than raw network traffic data alone, hence capitalizing on the learned patterns and efficiency of the supervised model. Recent works have started exploring the deployment of DQN in IoMT network security for better anomaly detection, showing its potential for adaptive threat detection[9]. The target Q-values are calculated using the Bellman equation as shown in Eq. 1 :

Parameter	Value/description	Explanation
Optimizer	Adam	–
Learning Rate	0.001	Default for Adam optimizer
Batch Size	16	Small batch size enables frequent updates
Episodes	1500	Suitable for stable convergence of rewards
Discount Factor (γ)	0.95	–
Epsilon Start	1.0	Encourages full exploration initially
Epsilon Minimum	0.01	–
Epsilon Decay	0.995	Suitable for smooth transition from exploration to exploitation
Replay Buffer Size	2000	Maintains recent transitions for stable learning
Loss Function	Mean Squared Error (MSE)	Standard for Q-value approximation in DQN
Framework	PyTorch	Supports modularity
Neural network architecture	2 fully connected layers: Input → 32 neurons (ReLU) → Output layer (2 actions)	Simplicity of architecture for fast inference for IoMT edge cases

Table 6. Hyper parameters used for training the DQN agent.

$$Q^*(s,a) = \mathbb{E}\left[r_{t+1} + \gamma \max_{a'} Q^*(s_{t+1}, a') \,\middle|\, s_t = s, a_t = a \right] \tag{1}$$

where $Q^*(s,a)$ is the optimal Q-value for taking action a in state s. $\mathbb{E}$ denotes the expected value of the total reward, and γ is the discount factor ($0 \leq \gamma \leq 1$), which determines the importance of future rewards in decision making.

Let the original dataset be denoted by S, the training set as S_{train}, and the testing set as S_{test}. The division is expressed in Eq. 2:

$$S = S_{\text{train}} \cup S_{\text{test}} \tag{2}$$

Where the ground truth label is the actual class label of the data point in the dataset, and $R_d(s,a)$ is the reward for taking action a in state s. Let A be the action taken by the RL agent, and A_{actual} be the actual class label of the data point[66]. Then, the reward R_d is given by Equation 3:

$$R_d = \begin{cases} +2, & \text{if } A = A_{\text{actual}} \text{ and } A = 1 \\ +1, & \text{if } A = A_{\text{actual}} \text{ and } A = 0 \\ -3, & \text{if } A = 0 \text{ and } A_{\text{actual}} = 1 \\ -1, & \text{if } A = 1 \text{ and } A_{\text{actual}} = 0 \end{cases} \tag{3}$$

This reward function takes into account the following factors: a +2 reward is given for correctly identifying an attack (true positive), and a +1 reward for correctly identifying benign activity (true negative). A more significant penalty of -3 is applied for missing a real attack (false negative), due to the possibility of serious consequences following for ineffective threat detection across healthcare systems. The penalty for a false alarm (false positive) is -1 as this is a lesser consequence in comparison to undetected threats.

Phase III—Generalization

This segment validates the generalization capability of the proposed hybrid model, whereby the model is tested on generalization aspects across multiple unseen datasets. Generalization is a critical aspect to validate model robustness and reliability in dynamic domains like cyber security, which is characterized by ever-evolving threats[66]. Compared to models that perform well only on training data, models capable of generalizing across domains are suited for real-world tasks[67]. Evaluating across diverse data sets enables higher external validity and limits the probability of overfitting, making it a critical process before any deployment occurs[68]. The implementation of the generalization aspects in this work is shown in Fig. 4.

Data preparation for generalization

A uniform preprocessing pipeline is applied to each dataset to ensure consistency across datasets. This includes:

- Data loading and feature/label separation: The dataset on which generalization aspects are to be evaluated is loaded, and then it is formatted in a manner that features and target labels are separated, enabling a consistent input structure[69].
- Missing value imputation: To assign values to missing values, an imputation technique (mean) was used while preserving input distributions[70].
- Train-test split (80/20): All datasets used in the generalization evaluation process used the same train-test split uniformly (80% train and 20% test). This was intended to enable the fairest possible baseline for comparing each model's performance.

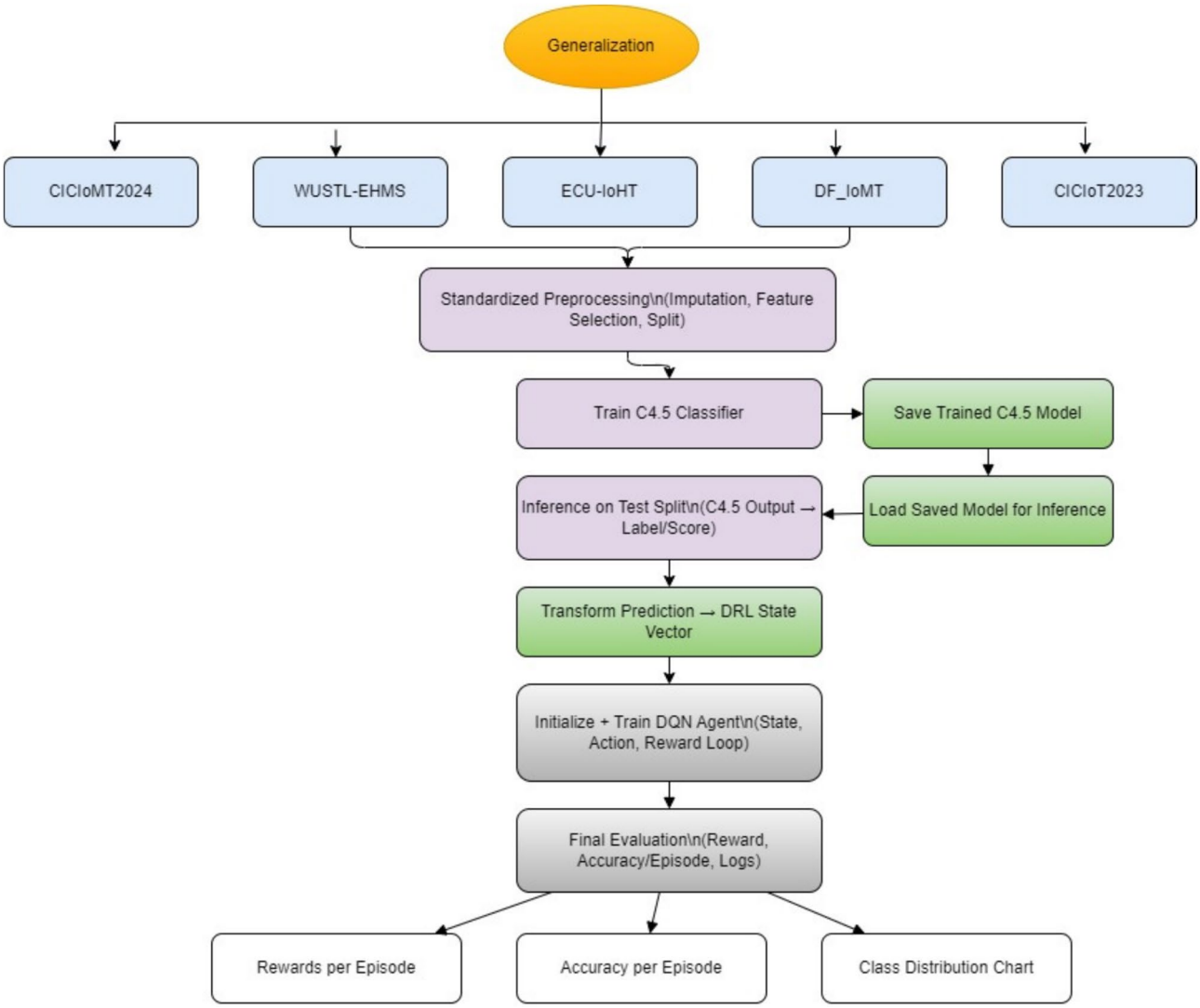


Fig. 4. Workflow—evaluation of generalization aspects.

Standardized pre-processing does not introduce bias associated with how datasets are handled and ensures that the performance of the model represents generalization, not variance from pre-processing[71].

Model application and training

The C4.5–DQN classifier is uniformly applied to all datasets to maintain consistency in the generalization process. SMOTE is used to balance class distributions, and Random Forest-based feature selection is done to ensure the inclusion of the most appropriate and relevant attributes in the classification process. The C4.5 model is then trained on each processed dataset and saved for further inference. Using the same model structure across datasets ensures comparability and ensures a generalization aspect in a real sense[72]. Additionally, research demonstrates that sustaining the same selection of models and feature selection transformation pipeline lowers variance in evaluations across datasets[73].

Model inference and DQN integration

The trained C4.5 models (as classifiers) are each considered to produce inference on their respective test sets and have their predictions used for the Deep Q-Network (DQN) framework. The DQN agent is framed in a way that it considers the classifier output to be the environmental state input. For every dataset, the RL environment is created, initialized, and then the agent interacts within the environment to learn detection-response behaviors. This allows us to merge static classification with adaptive policy learning, where the DQN agent can learn response patterns against unexpected attacks within a range of IoMT environments[74]. An additional use for DQN is that it is more robust to noisy or imbalanced test data; the agent only needs to learn a strategy for maximizing reward, which may not involve classification[75].

This staged hybrid design leverages both the bias–variance trade-off and state-space reduction: C4.5 (high bias, low variance) provides stable, explainable classifications, while DQN (low bias, high variance) fine-tunes

only ambiguous cases. Such gating reduces dimensionality and stochasticity for DRL, mitigating overfitting and aligning adaptive updates with interpretable Phase I boundaries[76,77]. Empirical studies confirm that these hybrid IDS architectures have higher performance metrics than standalone DRL and significantly improve convergence speed[78], making them well-suited for security-sensitive IoMT applications.

Evaluation metrics in DQN environment
The DQN agent is trained in a generalization environment. Cumulative Reward is calculated to indicate the capability of the system to detect and respond effectively. Accuracy per Episode approximates episode-level detection accuracy. Simulation logs demonstrates a qualitative view of agent behavior and learning trends. This shift in evaluation metrics from static parameters like F1-score, etc to dynamic & reward-based metrics accuracy per episode & 'reward' is in line with desired RL practices to test model capability and long-term strategy success[73]. The RL-driven evaluation also improves model operational performance in emergent IoMT environments[79].

Tools and experimental setup required
Various tools and frameworks were utilized for all experiments conducted under this study, The DRL component was trained and evaluated on a system with an Intel Core i5-8250U CPU, Intel UHD Graphics 620 (integrated GPU), 24 GB DDR4 RAM, and a 512 GB SK Hynix SATA SSD. Training the DQN phase took approximately 30 mins for 1500 episodes, while inference latency per packet flow averaged 23.35 ms.The C4.5 phase runs completely on the CPU with insignificant latency (< 10 ms). The hybrid model, despite being performed on modest, non-dedicated GPU hardware, runs in the compute envelope that is typical with IoMT edge servers, demonstrating its suitability in a lightweight and deployment ready environments.

Basic and advanced ensemble ML methods
Scikit-learn 1.6.1 was employed for traditional machine learning models. This included Random Forest, XGBoost, Decision Tree (C4.5), and Voting Ensemble (RF GB).

Deep learning/deep reinforcement learning models
PyTorch 2.7.1 was employed for designing and training the Multilayer Perceptron (MLP) architectures and the DQN neural network. 'Gym 0.26.2' library (which is a standard API for reinforcement learning, and a diverse collection of reference environments) was used for the RL environment[41]. Development and model experimentation were conducted in Visual Studio Code (VSCode) while using the Jupyter Notebook Extension for prototyping purposes on the local machine. For GPU-accelerated resources, the training of the Voting Ensemble and Complex MLP used resources provided via Google Colab on the cloud, which provided a significant speedup in model training and able to handle larger model architectures. For the working environment utilized, Python 3.12 was used, where all code and results were version-controlled and archived to ensure reproducibility and track experimental changes.

Results and analysis

This section offers a comprehensive study of the application of six machine learning classifiers to find the optimum model in terms of accuracy and latency time. The evaluation metrics involve classification performance, inference latency, model complexity, and trade-off identification. Each classifier is also evaluated in terms of real-world implementation feasibility, particularly in resource-constrained IoMT environments.

High accuracy performance across classifiers

The performance metrics of all models were evaluated. Ensemble-based methods, specifically the Voting Classifier and XGBoost, achieved the highest scores across all measures, with accuracies of approximately 99.0% and 98.8%, respectively. Notably, lightweight models such as C4.5 also produced strong results and may serve as a viable alternative in resource-constrained environments. Figure 5 depicts detailed results for each parameter (Accuracy, Precision, Recall & F1 Score) for binary classification. Similarly, Figs. 6 and 7 represent the same for Categorical and Multiclass, respectively.

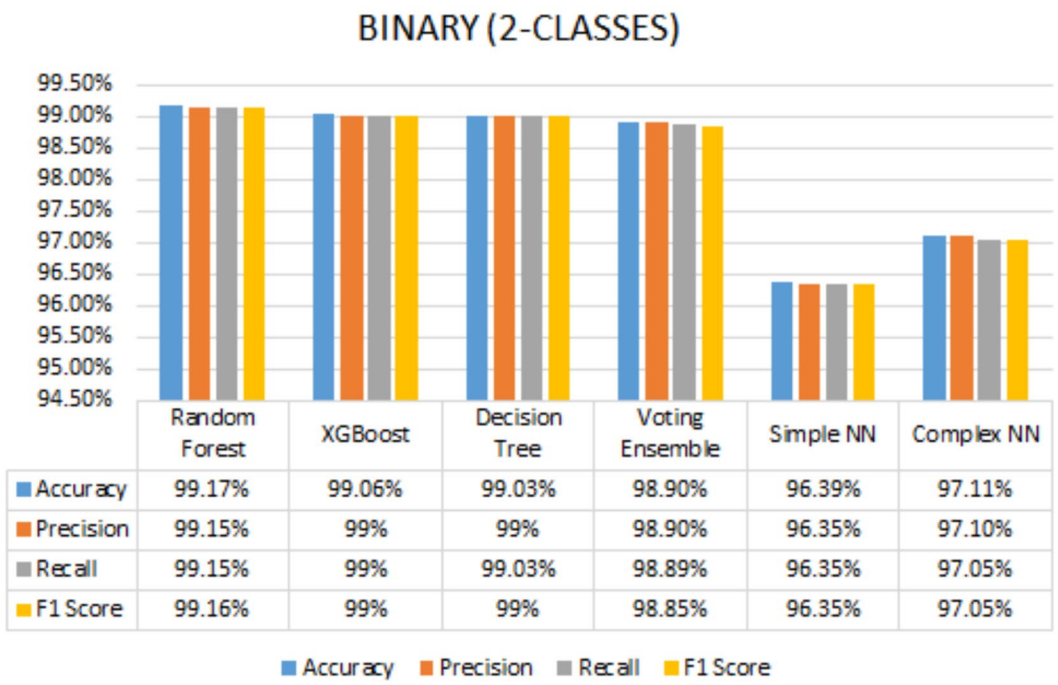
Inference latency evaluation for IoMT feasibility

The average inference times for each model across 200 prediction runs are calculated for binary classification only (2 x classes) and are shown in Fig. 8.

The lightweight models (C4.5 and Simple Neural Network) had the lowest latency of 9.64 ms and 25.24 ms, respectively, suggesting that these models would work best for real-time applications. Meanwhile, Ensemble (RF + GB), Complex Neural Network, and Random Forest had latency exceeding 600 ms, which may hinder adoption within latency-sensitive healthcare contexts. This Inference time benchmark is a critical measure for recognizing which classifiers meet the timing constraints of edge-based IoMT intrusion detection systems.

Performance of the reinforcement learning agent

The Total Reward per Episode plot shows the DQN agent's learning development in the IoMT intrusion detection environment. The starting rewards are on the negative side because the agent is exploring its environment. However, with increasing episode numbers, the rewards rise and settle around 750–780 at around 920 episodes as represented in Fig. 9, indicating some convergence to an optimal policy. This is also depicted in the area of stabilization, which suggests the agent has learned to separate attacks and benign behaviors successfully. Again, the high and stable total reward corresponds with plenty of correct classifications and few misclassifications,



	Random Forest	XGBoost	Decision Tree	Voting Ensemble	Simple NN	Complex NN
Accuracy	99.17%	99.06%	99.03%	98.90%	96.39%	97.11%
Precision	99.15%	99%	99%	98.90%	96.35%	97.10%
Recall	99.15%	99%	99.03%	98.89%	96.35%	97.05%
F1 Score	99.16%	99%	99%	98.85%	96.35%	97.05%

Fig. 5. Performance metrics for all classifiers—binary (2-classes).

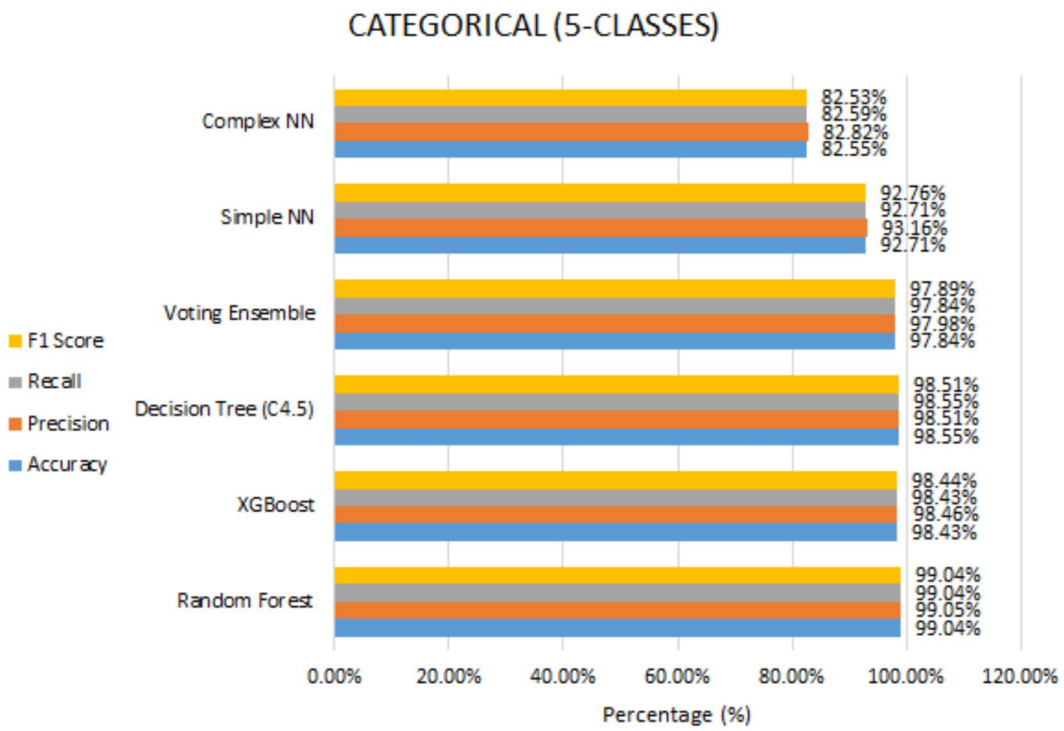
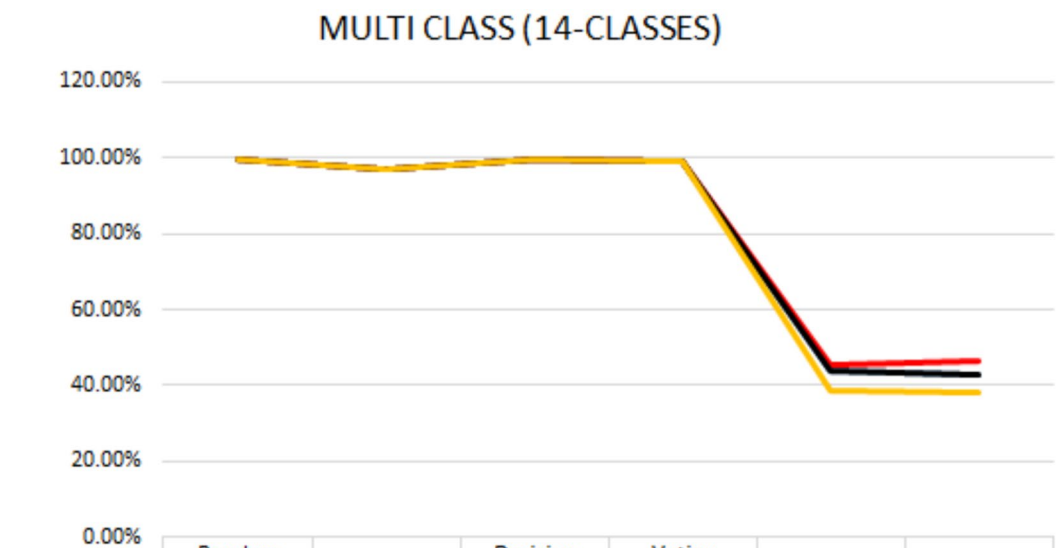


Fig. 6. Performance metrics for all classifiers—categorical (5-classes).



	Random Forest	XGBoost	Decision Tree (4.5)	Voting Ensemble	Simple NN	Complex NN
Accuracy	99.64%	97.01%	99.56%	99.07%	43.97%	42.79%
Precision	99.64%	97.02%	99.56%	99.15%	45.65%	46.57%
Recall	99.64%	97.01%	99.56%	99.07%	43.97%	42.79%
F1 Score	99.64%	97%	99.56%	99.11%	38.54%	38.06%

Fig. 7. Performance metrics for all classifiers—multi class (14-classes).

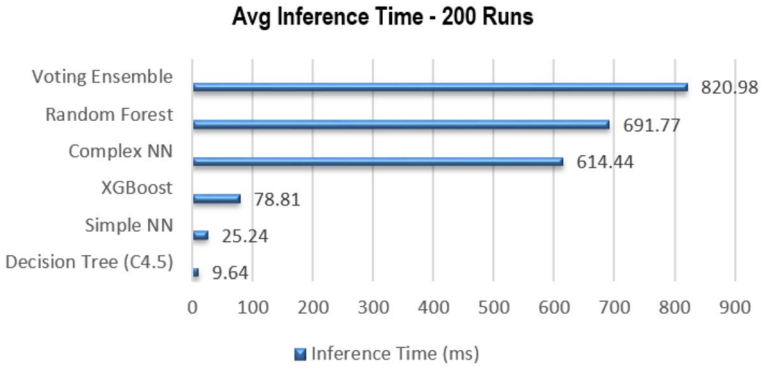


Fig. 8. Average inference time (binary classification)-each model.

which represents how effective the intrusion detection is in the real world. The maximum theoretical cumulative reward per episode depends on the Eq. 4.

$$R_{\max} = 2 \times \#\text{Attack Samples} + 1 \times \#\text{Benign Samples} \tag{4}$$

Max reward (Rmax) can go up to 1000 if nearly all 500 samples are attacks (but that's statistically unlikely). Since Max cumulative reward depends on the class distribution in the sampled 500 data points per episode. Given our balanced dataset and random sampling, most episodes have 250 attack samples (theoretically yielding 750 max reward). However, due to random variation, some episodes have slightly more attacks, resulting in a theoretical maximum reward up to 780, confirming that the agent reaches near-optimal performance.

This is consistent with similar findings in previous studies using DQN for IoT intrusion detection, where cumulative reward trends directly indicate improved learning and threat detection[80]. To provide additional context and strengthen the comparison, we have included two baseline policies: a random policy, represented as a red dashed line, and a static policy, represented by green dots. Both baselines give consistently low and highly variable rewards over all episodes, clearly showing that they cannot adapt to dynamic attack scenarios. The DQN agent learns incrementally and consistently outperforms both baselines. The value of the static policy remains constant at -652, while the value of a random policy bordered between -268 and -38 over the course of

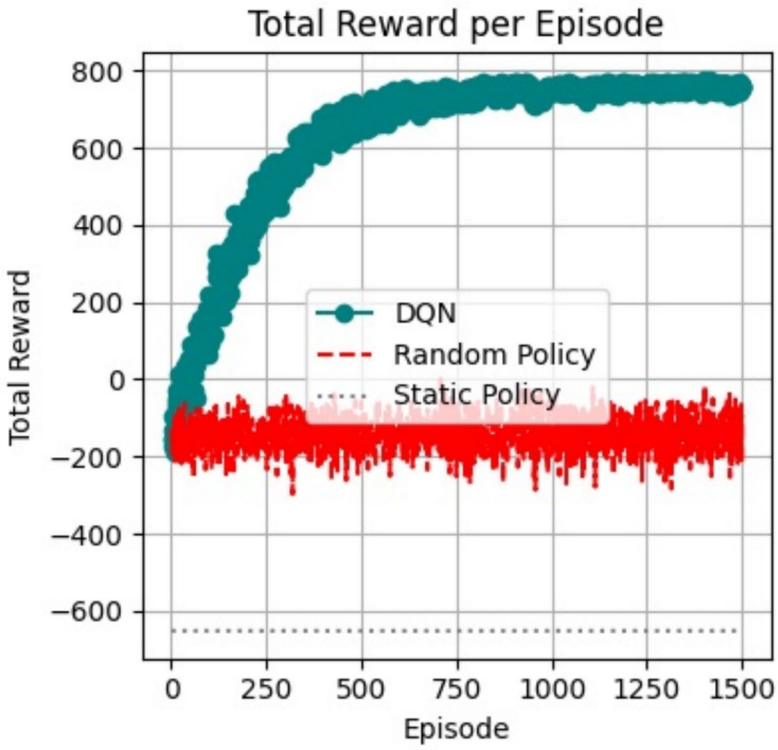


Fig. 9. Total reward administered by DQN.

1500 episodes.The stabilization in rewards supports the idea that DQN is indeed a self-improving approach to adaptive intrusion detection in a dynamic environment.

The "Accuracy per Episode (%)" plot indicates a sharp increase in performance from 50% to around 99% as illustrated in Figure 10, suggesting the agent became increasingly proficient in decision making. This high degree of accuracy is stable and near perfect, suggesting a generalization of the hybrid model to many different and ubiquitous types of threat patterns. Performance-wise, this model surpasses or performs similarly to static models, such as Voting Ensemble (99.0% accuracy) and XGBoost (98.8% accuracy), while continuing to learn. The dynamic nature of the training allows for new or changing threats to be detected, an important strength over traditional static models[81,82]. The accuracy curve suggests the long-term stability of the model as well as its fit for use in security-sensitive healthcare settings.

The "Action Distribution" chart denotes how many times the agent identifies inputs as either "Benign (0)" or "Attack (1)" in its training. A consistent and symmetrical distribution demonstrates that the agent discovered a reliable classification policy as shown in Fig. 11.

As long as the agent consistently detects "Attack (1)" cases and does not overfit on benign behavior, the agent demonstrates the ability to generalize. The action distributions align with other behavioral indicators (i.e., reward and accuracy) and suggest the model's efficacy for real world deployment. Similar distributions were observed in multi-agent reinforcement learning IDS studies that reported stable action policies once the model matured[81,83]. The observations confirm clear evidence of strong learning and decision making exhibited by the RL agent[84].

The amalgamation of the pre-trained C4.5 model and DQN is a unique solution contextualizing the trade-off between detection accuracy and deployment readiness. Although C4.5 alone is capable of good low latency (9.6 ms), but has static performance at a given point in time. Whereas, the ensemble models have achieved higher accuracy but at the cost of high inference time i-e (>600 ms). The proposed hybrid system (C4.5-DQN), effectively engages the efficiency of C4.5 in the first instance and then contributes to a low-to-moderate overall inference time for the adaptive footprint of the system. When efficiently deployed, the DQN algorithm provides both continuous learning and adaptation, so that the system can maintain high accuracy dynamically to detect zero-day attacks or evolving attacks that static systems cannot detect[64].

Trade-off between detection accuracy and deployment readiness

In IDS research, the factor of accuracy often overcomes the factor of real-time suitability; however, in some clinical IoMT situations, real-time effectiveness is equally important. Table 7 integrates both these factors. Models such as Voting Ensemble and Complex Neural Network have higher accuracy, but much greater inference times, and therefore will not be applicable for immediate threat detection. C4.5 and Simple Neural Network provide a good compromise on both accuracy and time. These results have reinforced that in the scope of suitable IDS for IoMT, fast static classifier (C4.5) coupled with dynamic threat detection ability of Deep Q Networks provides an

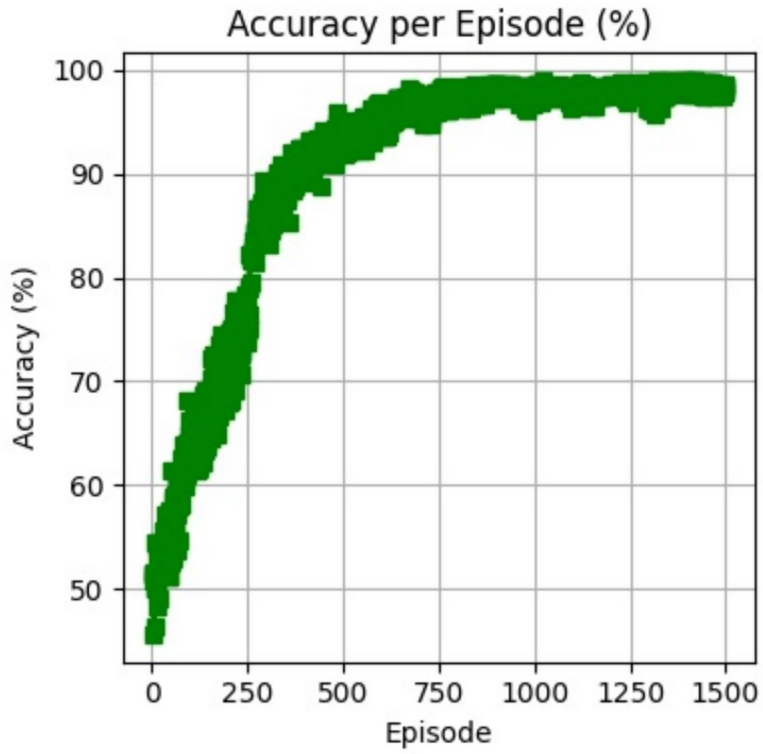


Fig. 10. Accuracy per episode graph-DQN.

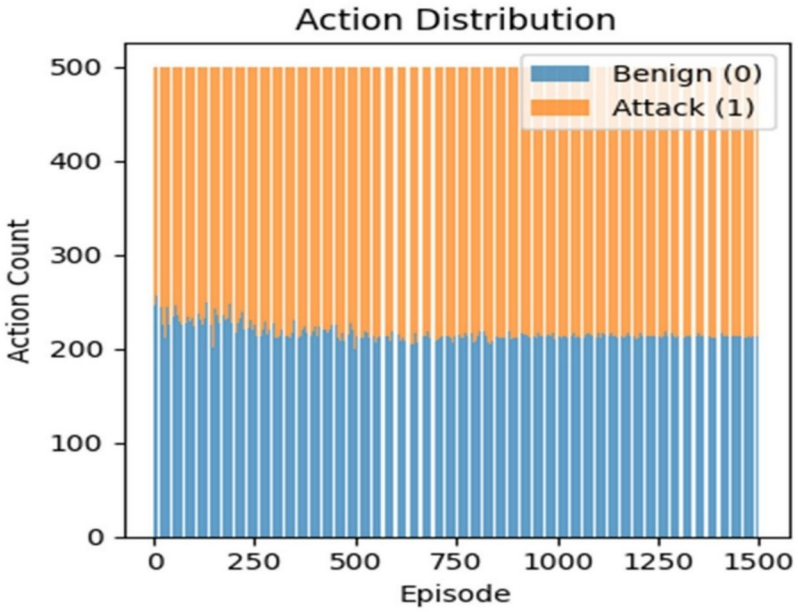


Fig. 11. Action distribution graph.

optimum trade-off but offers a wholesome approach for accurate and timely detection all types of threats on the go, which is critical in the case of real traffic routing through IoMT networks.

Cross-dataset generalization results

To validate the robustness and adaptability of the proposed hybrid C4.5-DQN model, the program was tested on a variety of real-world datasets taken from the IoT and IoMT domains. The goal was to examine whether the model could generalize to data, scenarios, and contexts beyond what was seen in the training set, and still perform

Model	Accuracy	Inference time (ms)	Real-time suitability
Voting Ensemble (RF + GB)	High	820.98	Limited—slow inference restricts real-time use
Complex Neural Network	High	614.44	Limited—inference delay affects real-time response
Random Forest	High	691.77	Limited—strong accuracy but latency is too high
XGBoost	High	78.81	Partially Suitable—efficient, but not ideal
Simple Neural Network	Moderate–High	25.24	Suitable—balances accuracy and latency
C4.5 (Pruned Entropy)	Moderate	9.64	Highly Suitable—fast and interpretable
Hybrid C4.5-DQN	High (Adaptive, Zero-Day)	575.18	C4.5's accuracy with DQN's adaptability for real-time, intelligent threat detection

Table 7. Comparison of models in terms of accuracy and real-time suitability.

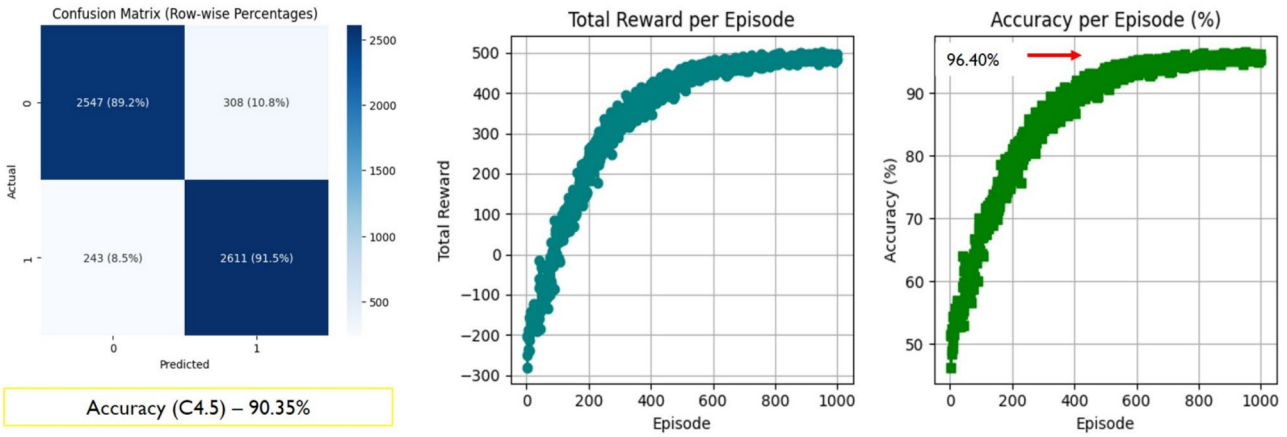


Fig. 12. C4.5-DQN results on WUSTL-EHMS dataset.

Fig. 13. C4.5–DQN Results–ECU-IoHT.

optimally in new heterogeneous environments. **WUSTL-EHMS**: Accuracy improved from 90.35 to 96.40% as illustrated in Fig. 12, verifying the model's capabilities to adapt in a smart healthcare environment.**ECU-IoHT (binary classification)**: A modest yet important improvement from 99.00% to 99.20% shows stability in precision-critical environments in Fig. 13.

DF-IoMT: The model retained its full 100% accuracy as represented in Fig. 14, which establishes resilience across cases due to unseen distributions of threat data.

CICIoT23: The model managed to gain performance from 94.23% to 99.20% in a solid degree of robustness across a modern and large-scale diverse attack dataset represented in Fig. 15. **ECU-IoHT (Multiclass)**: The hybrid system also confirmed multiclass classification with excellent performance at 99.35% from 97.28% demonstrating versatility across granular threat categories. These improvements across multiple, independent datasets support the stated system's ability to generalize. Unlike domain static models that operate well when trained under fixed conditions, the hybrid architecture maintains a dynamic learning backbone that adapts

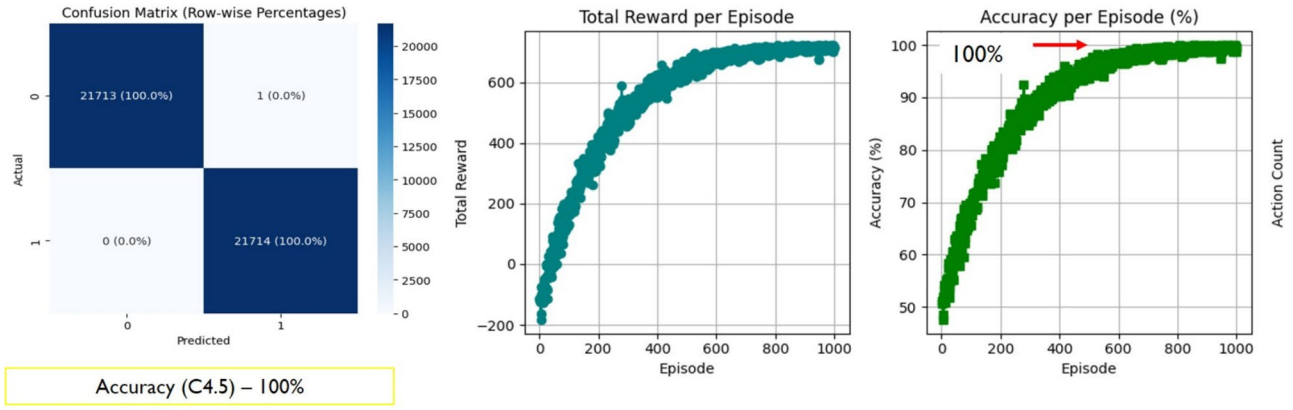


Fig. 14. C4.5–DQN Results–IoMT.

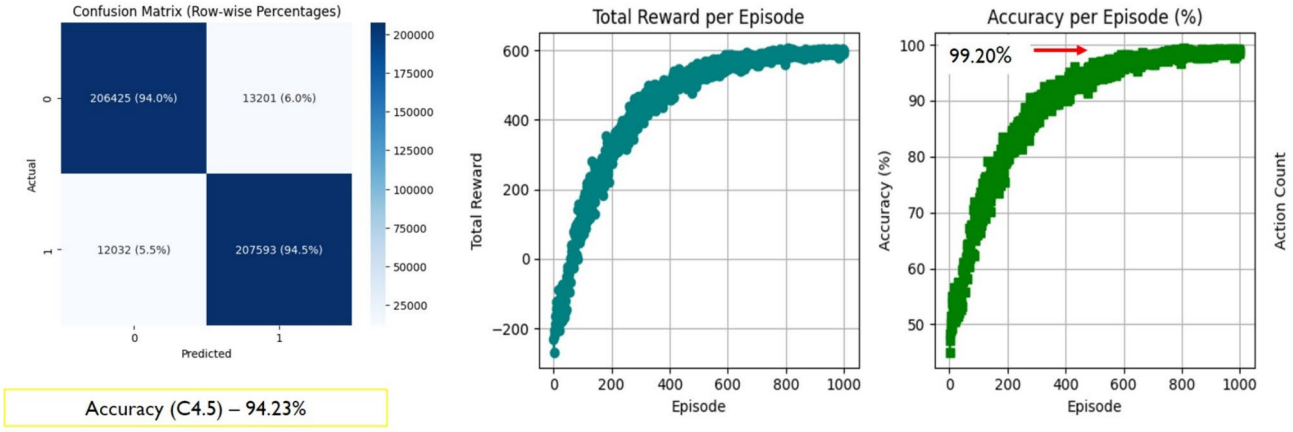


Fig. 15. C4.5 - DQN Results - CICIoT23.

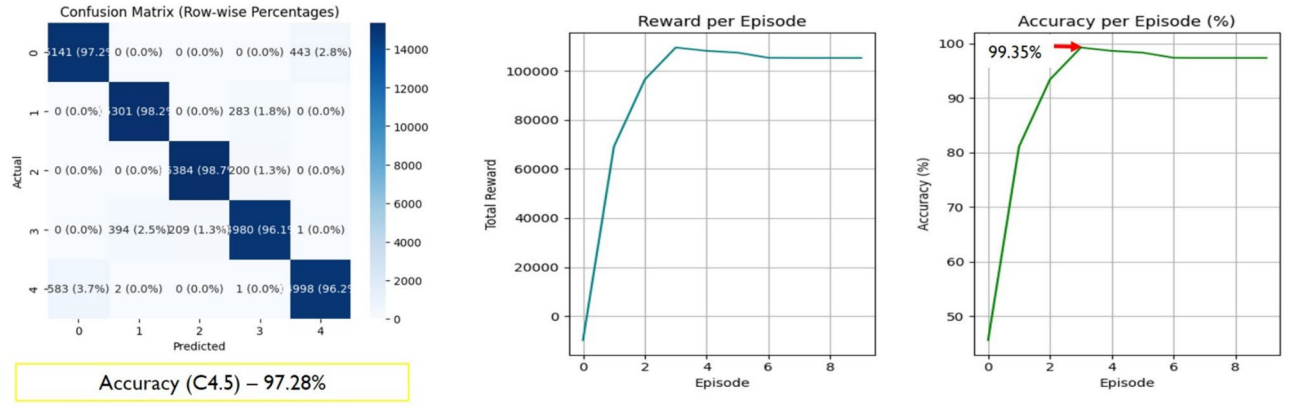


Fig. 16. C4.5–DQN Results–ECU-IoHT (MultiClass).

well to new attack patterns and data characteristics. This adaptability, coupled with low-latency and strong detection accuracy, renders the C4.5-DQN system a feasible and scalable system for real-time IoMT security implementations where resilience and generalizability matter. However, an equally important aspect of IDS performance in IoMT environments is the system's generalization across heterogeneous environments and new threat patterns[85] (Fig. 16).

Model	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Remarks
HIDS-RPL[86]	CIC-DDoS2019	99.87	98.50	98.64	98.54	Hybrid CNN + LSTM for RPL-based IoMT; optimized for low-power environments
Meta-Learning Ensemble IDS[87]	IoMT-specific datasets (various feature sizes)	99.50	99.40	99.60	99.50	Model dynamically reweights classifiers
PSO-AdaBoost IDS[88]	NSL-KDD	–	–	96.67	–	Particle Swarm Optimization for feature selection and AdaBoost for classification (Accuracy/Precision values not numerically reported)
HIDS (GA-DT)[89]	NSL-KDD	99.88	–	–	–	Genetic Algorithm (GA) – Decision Tree (DT) model
StandAlone DQN	CICIoMT-2024	99.70	98.31	98.81	98.60	Dynamic Learning ability of DQN
Hybrid C4.5–DQN (Proposed)	CICIoMT-2024	99.40	99.65	99.65	99.48	Combines the speed/efficiency of C4.5 with the adaptive learning of DQN

Table 8. Comparison of proposed Hybrid C4.5–DQN IDS with selected recent IDS approaches.

Comparison with recent Hybrid IDSs

To put the performance and novelty of the C4.5-DQN hybrid IDS into perspective, we expanded our review to assess not just standard classifiers, but also the latest hybrid and adaptive intrusion detection systems from existing literature. Detail given as per Table 8

Discussion and future recommendations

The findings this section indicate that when applying machine learning classifiers to intrusion detection in the Internet of Medical Things (IoMT) ecosystems, selecting the machine learning classifier is not simply an act of finding the most accurate detection; it is also the consideration of practical aspects of the deployment; and thus how usable the model will be in practice[11]. Each classifier produced varying levels of performance in terms of accuracy, inference time, and latency. The ensemble models such as voting Ensemble or XGBoost tended to have the most accuracy, but also included the biggest inference times, which is important when the application is a real-time medical application that integrates other IoMT systems in near real-time—i.e. the lower the latency in detecting the intrusion, the better it is for patient safety and care[15].

By comparison, Decision trees (C4.5) and the Simple Neural Network provided the greatest compromise, with accuracy that was high enough to be adequate, with much lower inference times of 9.64 ms and 25.24 ms, respectively. These attributes contribute to C4.5 and the Simple Neural Network being valuable options for real-world intrusion detection solutions on systems in a resource-constrained environment. In addition to this experimental paradigm, this work also examined hybrid modeling to improve adaptability. The proposed mixed C4.5—DQN intrusion detection system combined the rapid decisions afforded by a pre-trained C4.5 classifier with the adaptability and self-learning behaviors of Deep Q-Networks (DQN). The hybrid model continuously adjusted its detection policy based on the interaction observed with the environment, and over each episode observed increases in reward and accuracy. The self-adaptive quality of the model allows it to maintain effective performance under changing cyberattack conditions.

Furthermore, the results provide evidence that the proposed model produced meaningful performance gains over baseline mechanisms in cross-domain evaluations on the unseen datasets WUSTL-EHMS, CICIoT23, DF-IoMT and ECU-IoHT. The consistent performance across a variety of heterogeneous datasets further supports the framework's robustness and real-world applicability. These findings also validate recent observations showing how generalization continues to be among the most challenging aspects of IDS systems due to the heterogeneous nature of IoMT data and attack behavior volatility.

Future research recommendations include advancing the adaptability and robustness of IoMT intrusion detection systems by exploring more sophisticated Deep Reinforcement Learning (DRL) architectures, such as Policy Gradient and Actor-Critic models. Emphasis should also be placed on improving generalization to entirely new attack types through comprehensive stress testing and robustness evaluations. Additionally, conducting real-world pilot deployments across diverse IoMT network conditions and communication protocols will be essential for validating system performance and minimizing dataset-specific dependencies. Finally, incorporating formal statistical hypothesis testing using paired *t*-tests when normality assumptions are satisfied[90,91] or the Wilcoxon signed-rank test for non-normal or small-sample cases[92] will further enhance the reliability and rigor of model evaluation.

Limitations

Although this study offers a thorough comparative evaluation of machine learning classifiers used for intrusion detection in IoMT environments, there were several limitations identified through experimentation and system design

Long training time with complex models

One of the significant hurdles encountered throughout this study was the lengthy training times associated with some of the classifiers. The Voting Ensemble model, which is used for multi-class classification, it has taken roughly 1476 min (24.6 h) to complete a single training cycle, even where systems had strong computational power. This limitation is a glaring obstacle for real-time model retraining or updates on IoMT systems.

Protocol dependency and latency variation

IoMT networks can contain an array of heterogeneous devices communicating in different protocols (for example, Bluetooth, MQTT, WiFi), each of which has its latency characteristics and makes it more challenging for an IDS model to generalize its performance across the protocol layers. Recent work has shown that the heterogeneity of protocol leads to disparate data patterns and timing behaviors, which can reduce the consistency of the IDS detection given the same context derived from different device types[43].

Adaptability to unseen real-world attacks

While the proposed system establishes robust generalization across datasets, the most difficult hurdle remains directly adapting to novel or evolving attack strategies, especially in real-time operational environments. Most training datasets do not represent the full scale of diverse attack vectors related to IoMT ecosystems, which may impact adaptability upon deployment. Research has suggested that reinforcement learning agents are, in theory, potentially effective, but they require extensive live interaction with the system to genuinely generalize across non-stationary attack patterns[93,94].

Conclusion

This research presented a comprehensive comparative analysis of six different machine learning classifiers: Random Forest, XGBoost, C4.5, Voting Ensemble, Simple Neural Network, and Complex Neural Network for IoMT intrusion detection using the CICIoMT2024 dataset. While ensemble models such as Voting Ensemble and XGBoost achieved the highest classification accuracies of up to 99%. However, they were considerably slower and had a higher computational complexity compared to lightweight models. The lightweight models like C4.5 and Simple Neural Network demonstrated the next highest level of accuracy, but with considerably lower latencies (9.64 ms and 25.24 ms, respectively). The proposed hybrid C4.5–DQN model further enhanced adaptability, maintaining stable performance under evolving threat conditions. In summary, this study finds that the optimal IoMT intrusion detection systems will be those that combine accuracy, speed, and adaptability while also guaranteeing reliability and safety in healthcare settings with limited resources.

Data availability

The data sets analyzed during the current study are publicly available at https://www.unb.ca/cic/datasets/iomt-dataset-2024.html, https://www.cse.wustl.edu/jain/ehms/index.html, https://researchdata.edu.au/ecu-ioht/1714095, https://www.kaggle.com/datasets/faisalmalik/iot-healthcare-security-dataset/data and https://www.unb.ca/cic/datasets/iotdataset-2023.html.

Received: 6 July 2025; Accepted: 5 December 2025
Published online: 20 December 2025

References

1. Shanmugam, B. & Azam, S. Risk assessment of heterogeneous iomt devices: A review. *Technologies* **11**, 31 (2023).
2. Khaled, A. E. Internet of medical things (iomt): Overview, taxonomies, and classifications. *J. Computer Commun.* **10**, 64–89 (2022).
3. Berguiga, A., Harchay, A. & Massaoudi, A. HIDS-IoMT: A deep learning-based intelligent intrusion detection system for the internet of medical things. *IEEE Access* **13**, 32863–32882. https://doi.org/10.1109/ACCESS.2025.3543127 (2025).
4. Alalhareth, M. & Hong, S.-C. Enhancing the internet of medical things (iomt) security with meta-learning: A performance-driven approach for ensemble intrusion detection systems. *Sensors (Basel, Switzerland)* **24**, https://doi.org/10.3390/s24113519 (2024).
5. Kalakoti, R., Nomm, S. & Bahsi, H. Explainable transformer-based intrusion detection in internet of medical things (iomt) networks. In *2024 International Conference on Machine Learning and Applications (ICMLA)*, 1164–1169, https://doi.org/10.1109/ICMLA61862.2024.00179 (2024).
6. Shaikh, J. *et al.* A deep reinforcement learning-based robust intrusion detection system for securing iomt healthcare networks. *Front. Med.* **12**, https://doi.org/10.3389/fmed.2025.1524286 (2025).
7. Khan, I. A. et al. Fed-inforce-fusion: A federated reinforcement-based fusion model for security and privacy protection of iomt networks against cyber-attacks. *Information Fusion* **101**, 102002. https://doi.org/10.1016/j.inffus.2023.102002 (2023).
8. Angeline, L. *et al.* Alot-driven machine learning for anomaly detection in structural health monitoring. In *2024 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, 687–692 (IEEE, 2024).
9. Naeem, M. R. et al. Cyber security enhancements with reinforcement learning: A zero-day vulnerabilityu identification perspective. *PLoS One* **20**, e0324595 (2025).
10. Naseer, M., Ullah, F., Ahmad, J., Jhaveri, R. H. & Gadekallu, T. R. Adaptive iot defense with deep q-learning model for dns spoofing prevention in self-organizing networks. *IEEE Communications Standards Magazine* (2025).
11. Dadkhah, S. et al. Ciciomt 2024: A benchmark dataset for multi-protocol security assessment in iomt. *Internet Things* **28**, 101351 (2024).
12. Axelsson, S. Intrusion detection systems: A survey and taxonomy. *Tech Rep* (2000).
13. Balhareth, G. & Ilyas, M. Optimized intrusion detection for iomt networks with tree-based machine learning and filter-based feature selection. *Sensors* **24**, 5712 (2024).
14. Ekwueme, C. P., Adam, I. H., Dwivedi, A. *et al.* Lightweight cryptography for internet of things: A review. *EAI Endorsed Transactions on Internet of Things* **10** (2024).
15. Ashraf, J., Raza, G. M., Kim, B.-S., Wahid, A. & Kim, H.-Y. Making a real-time iot network intrusion-detection system (inids) using a realistic bot–iot dataset with multiple machine-learning classifiers. *Applied Sciences (2076-3417)* **15** (2025).
16. Ghubaish, A., Yang, Z. & Jain, R. Hdrl-ids: A hybrid deep reinforcement learning intrusion detection system for enhancing the security of medical applications in 5g networks. In *2024 International Conference on Smart Applications, Communications and Networking (SmartNets)*, 1–6, https://doi.org/10.1109/SmartNets61466.2024.10577692 (2024).
17. Ravi, S. et al. Deep learning-based network intrusion detection system for the internet of medical things (iomt). *Internet Things Manag.* https://doi.org/10.1109/IOTM.001.2300021 (2023).
18. Abu Daher, L. Towards secure iomt: Attack detection using deep q-learning in healthcare networks. In *Proceedings of the 16th International Conference on Developments in eSystems Engineering (DeSE)*, 407–412, https://doi.org/10.1109/DeSE60595.2023.10468942 (2023).

19. Attiya Khan, M. R., Bagdasar, O., Alabdulatif, A., Alamro, S. & Alnajim, A. Deep learning-driven anomaly detection for iomt-based smart healthcare systems. *Computer Modeling in Engineering & Sciences* **141**, 2121–2141, https://doi.org/10.32604/cmes.2024.054380 (2024).
20. Yahya, R. *et al.* Deep learning for enhanced iomt security: A gnn-bilstm intrusion detection system. In *2024 International Conference on Cybersecurity and Communication Systems (ICCSC)*, 1–6, https://doi.org/10.1109/ICCSC62074.2024.10616456 (2024).
21. Yang, W., Acuto, A., Zhou, Y. & Wojtczak, D. A survey for deep reinforcement learning based network intrusion detection (2024). arXiv:2410.07612.
22. Sharma, N. & Shambharkar, P. Multi-attention deepcrnn: an efficient and explainable intrusion detection framework for internet of medical things environments. *Knowledge and Information Systems* (2025).
23. Konatham, B. *A secure and efficient IIoT anomaly detection approach using a hybrid deep learning technique*. Ph.D. thesis, Wright State University (2023). Accessed: 2025-08-09.
24. Alfahaid, A., Alalwany, E., Almars, A., Alharbi, F. & Atlam, E. Machine learning-based security solutions for iot networks: A comprehensive survey. *Sensors* **25**, 3341. https://doi.org/10.3390/s25113341 (2025).
25. Yan, Z., Shukla, P., Shukla, P. & Thakur, K. Intrusion detection and mitigation method for the industrial internet of things using bidirectional convolutional long short-term memory and deep recurrent. *Wireless Personal Communications* (2025).
26. Karthikeyan, M., Brindha, R. & Vianny, M. Integration of metaheuristic based feature selection with ensemble representation learning models for privacy aware cyberattack detection in iot environments. *Sci. Rep.* **15**, 12345. https://doi.org/10.1038/s41598-025-05545-5 (2025).
27. Saheed, Y. & Chukwuere, J. Cps-iiot-p2attention: Explainable privacy-preserving with scaled dot-product attention in cyber physical system-industrial iot network. In *2025 IEEE International Conference on Industrial Technology (ICIT)* (IEEE, 2025).
28. Aflaki, A. *A Secure and Explainable AI-Based Framework for IIoT with Privacy-Prioritized Model Aggregation*. Ph.D. thesis, University of Calgary (2024). https://doi.org/10.11575/PRISM/47267.
29. Lazaros, K., Koumadorakis, D. E., Vrahatis, A. G. & Kotsiantis, S. Federated learning: Navigating the landscape of collaborative intelligence. *Electronics* **13**, 4744 (2024).
30. Rane, N., Choudhary, S. & Rane, J. Ensemble deep learning and machine learning: applications, opportunities, challenges, and future directions. *Smart Medical and Healthcare Systems* **1**, https://doi.org/10.48185/smhs.v1i2.1225 (2024).
31. Mienye, I. & Jere, N. Optimized ensemble learning approach with explainable ai for improved heart disease prediction. *Information* **15**, 394 (2024).
32. Murad, N., Hasan, M., Azam, M., & Yousuf, N. (A review of explainable deep learning healthcare techniques. IEEE Access, Unraveling the black box, 2024).
33. Khan, N., Nauman, M., Almadhor, A. & Akhtar, N. Guaranteeing correctness in black-box machine learning: A fusion of explainable ai and formal methods for healthcare decision-making. *IEEE Access* (2024).
34. Patil, S. et al. Explainable artificial intelligence for intrusion detection system. *Electronics* **11**, 3079 (2022).
35. Alalwany, E. et al. Stacking ensemble deep learning for real-time intrusion detection in iomt environments. *Sensors* **25**, 624 (2025).
36. Fatema, K. et al. Federated xai ids: An explainable and safeguarding privacy approach to detect intrusion combining federated learning and shap. *Future Internet* **17**, 234 (2025).
37. Sohail, F., Bhatti, M. A. M., Awais, M. & Iqtidar, A. Explainable boosting ensemble methods for intrusion detection in internet of medical things (iomt) applications. In *2024 4th International Conference on Digital Futures and Transformative Technologies (ICoDT2)*, 1–8 (IEEE, 2024).
38. Hady, A. A., Ghubaish, A., Salman, T., Unal, D. & Jain, R. Intrusion detection system for healthcare systems using medical and network data: A comparison study. *IEEE Access* **8**, 106576–106584 (2020).
39. Ahmed, M., Byreddy, S., Nutakki, A., Sikos, L. F. & Haskell-Dowland, P. Ecu-ioht: A dataset for analyzing cyberattacks in internet of health things. *Ad Hoc Networks* **122**, 102621 (2021).
40. Garg, N., Wazid, M., Singh, J., Singh, D. P. & Das, A. Security in IoMT-driven smart healthcare: A comprehensive review and open challenges. *Security and Privacy* **5**, 1–27. https://doi.org/10.1002/spy2.235 (2022).
41. Kumar, A. G., Rastogi, A. & Ranga, V. Evaluation of different machine learning classifiers on new iot dataset ciciot2023. In *2024 International Conference on Intelligent Systems for Cybersecurity (ISCS)*, 1–6 (IEEE, 2024).
42. Sun, S., Zhou, X., Wei, J., Xiao, Y. & Wang, J. An optimization of smote for anomaly detection based on high contribution sample screening. In *2023 China Automation Congress (CAC)*, 2010–2014 (IEEE, 2023).
43. Hussein, A. S., Li, T., Yohannese, C. W. & Bashir, K. A-smote: A new preprocessing approach for highly imbalanced datasets by improving smote. *Int. J. Comput. Intell. Syst.* **12**, 1412–1422 (2019).
44. Chen, R.-C., Dewi, C., Huang, S.-W. & Caraka, R. E. Selecting critical features for data classification based on machine learning methods. *J. Big Data* **7**, 52 (2020).
45. Altmann, A., Toloşi, L., Sander, O. & Lengauer, T. Permutation importance: A corrected feature importance measure. *Bioinformatics* **26**, 1340–1347 (2010).
46. Hwang, W.-J. & Ou, C.-M. Efficient header classification architecture for network intrusion detection. *J. Inform. Sci. Eng.* **25** (2009).
47. Shamsuddin, S. B. & Woodward, M. E. Modeling protocol based packet header anomaly detector for network and host intrusion detection systems. In *International Conference on Cryptology and Network Security*, 209–227 (Springer, 2007).
48. Gao, M., Zhang, K. & Lu, J. Efficient packet matching for gigabit network intrusion detection using tcams. In *20th International Conference on Advanced Information Networking and Applications-Volume 1 (AINA'06)*, vol. 1, 6–pp (IEEE, 2006).
49. Saraswathy, V., Kasthuri, N. & Ramyadevi, I. Multi-granularity approach for enhancing the performance of network intrusion detection with supervised learning. In *2016 10th International Conference on Intelligent Systems and Control (ISCO)*, 1–7 (IEEE, 2016).
50. Xu, C., Sun, W. & Li, M. Dtt: A dual-domain transformer model for network intrusion detection. *EAI Endorsed Transactions on Scalable Information Systems* **11** (2024).
51. Jiang, J., Wang, Q., Shi, Z., Lv, B. & Qi, B. Rst-rf: A hybrid model based on rough set theory and random forest for network intrusion detection. In *Proceedings of the 2nd International Conference on Cryptography, Security and Privacy*, 77–81 (2018).
52. Rodda, S. & Erothi, U. S. A roughset based ensemble framework for network intrusion detection system. *Int. J. Rough Sets Data Anal. (IJRSDA)* **5**, 71–88 (2018).
53. Jia, X. et al. A prediction-based anomaly detection method for traffic flow data with multi-domain feature extraction. *App. Sci.* **15**, 3234 (2025).
54. Zhao, R. et al. Application-layer anomaly detection leveraging time-series physical semantics in can-fd vehicle networks. *Electronics* **13**, 377 (2024).
55. Xu, Y. & Goodacre, R. On splitting training and validation set: A comparative study of cross-validation, bootstrap and systematic sampling for estimating the generalization performance of supervised learning. *J. Anal. Testing* **2**, 249–262 (2018).
56. Chung, Y.-S., Hsu, D. F. & Tang, C. Y. On the diversity-performance relationship for majority voting in classifier ensembles. In *Multiple Classifier Systems: 7th International Workshop, MCS 2007, Prague, Czech Republic, May 23-25, 2007. Proceedings 7*, 407–420 (Springer, 2007).
57. Kim, H., Kim, H., Moon, H. & Ahn, H. A weight-adjusted voting algorithm for ensembles of classifiers. *J. Korean Stat. Soc.* **40**, 437–449 (2011).
58. Patidar, P. & Tiwari, A. Handling missing value in decision tree algorithm. *Int. J. Computer Appl.* **70** (2013).

59. Chen, T. & Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794 (2016).
60. Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M. & Suganthan, P. N. Ensemble deep learning: A review. *Eng. Appl. Artif. Intell.* **115**, 105151 (2022).
61. Salman, R., Alzaatreh, A., Sulieman, H. & Faisal, S. A bootstrap framework for aggregating within and between feature selection methods. *Entropy* **23**, 200 (2021).
62. Breiman, L. Random forests. *Machine Learning* **45**, 5–32 (2001).
63. Hasan, M. A. M., Nasser, M., Pal, B. & Ahmad, S. Support vector machine and random forest modeling for intrusion detection system (ids). *J. Intell. Learn. Syst. Appl.* **2014** (2014).
64. Ramaswamy, A. & Hüllermeier, E. Deep q-learning: Theoretical insights from an asymptotic analysis. *IEEE Trans. Artif. Intell.* **3**, 139–151 (2021).
65. Mnih, V. *et al.* Human-level control through deep reinforcement learning. *Nature* **518**, 529–533 (2015).
66. Wang, J. et al. Generalizing to unseen domains: A survey on domain generalization. *IEEE Trans. Knowl. Data Eng.* **35**, 8052–8072 (2022).
67. Chen, K., Zhuang, D. & Chang, J. M. Discriminative adversarial domain generalization with meta-learning based cross-domain validation. *Neurocomputing* **467**, 418–426 (2022).
68. Yeung, D. S., Ng, W. W., Wang, D., Tsang, E. C. & Wang, X.-Z. Localized generalization error model and its application to architecture selection for radial basis function neural network. *IEEE Trans. Neural Netw.* **18**, 1294–1305 (2007).
69. Wujek, B., Hall, P. & Günes, F. Best practices for machine learning applications. *SAS Institute Inc* 3 (2016).
70. Liu, C.-H., Tsai, C.-F., Sue, K.-L. & Huang, M.-W. The feature selection effect on missing value imputation of medical datasets. *Appl. Sci.* **10**, 2344 (2020).
71. Li, G. et al. A classification method for incomplete mixed data using imputation and feature selection. *Appl. Sci.* **14**, 5993 (2024).
72. Packer, C. *et al.* Assessing generalization in deep reinforcement learning. *arXiv preprint* arXiv:1810.12282 (2018).
73. Zhang, J., Hao, J. & Fogelman-Soulié, F. Cross-data automatic feature engineering via meta-learning and reinforcement learning. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 818–829 (Springer, 2020).
74. Bertran, M., Martinez, N., Phielipp, M. & Sapiro, G. Instance-based generalization in reinforcement learning. *Adv. Neural Inform. Process. Syst.* **33**, 11333–11344 (2020).
75. Chen, J. Z. Reinforcement learning generalization with surprise minimization. *arXiv preprint* arXiv:2004.12399 (2020).
76. Qiu, L., Xu, Z., Lin, L., Zheng, J. & Su, J. Design and optimization of hybrid cnn-dt model-based network intrusion detection algorithm using deep reinforcement learning. *Mathematics* **13**, 1459. https://doi.org/10.3390/math13091459 (2025).
77. Alsaffar, A., Nouri-Baygi, M. & Zolbanin, H. Shielding networks: Enhancing intrusion detection with hybrid feature selection and stack ensemble learning. *J. Big Data* **11**, 64. https://doi.org/10.1186/s40537-024-00994-7 (2024).
78. Lucas, T., De Figueiredo, I. & Tojeiro, C. A comprehensive survey on ensemble learning-based intrusion detection approaches in computer networks. *IEEE Access* **11**, 104872–104899. https://doi.org/10.1109/ACCESS.2023.3318297 (2023).
79. Olisah, C. C., Smith, L. & Smith, M. Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective. *Computer Methods Programs Biomed.* **220**, 106773. https://doi.org/10.1016/j.cmpb.2022.106773 (2022).
80. Rookard, C. & Khojandi, A. Rriot: Recurrent reinforcement learning for cyber threat detection on iot devices. *Computers Security* **140**, 103786 (2024).
81. Tellache, A., Mokhtari, A., Korba, A. A. & Ghamri-Doudane, Y. Multi-agent reinforcement learning-based network intrusion detection system. In *NOMS 2024-2024 IEEE Network Operations and Management Symposium*, 1–9 (IEEE, 2024).
82. Sangoleye, F., Johnson, J. & Tsiropoulou, E. E. Intrusion detection in industrial control systems based on deep reinforcement learning. *IEEE Access* (2024).
83. Yu, K., Jin, K. & Deng, X. Review of deep reinforcement learning. In *2022 IEEE 5th advanced information management, communicates, electronic and automation control conference (IMCEC)*, vol. 5, 41–48 (IEEE, 2022).
84. Korkmaz, E. A survey analyzing generalization in deep reinforcement learning. *arXiv preprint* arXiv:2401.02349 (2024).
85. Fan, J., Wang, Z., Xie, Y. & Yang, Z. A theoretical analysis of deep q-learning. In *Learning for dynamics and control*, 486–489 (PMLR, 2020).
86. Berguiga, A., Harchay, A. & Massaoudi, A. Hids-rpl: A hybrid deep learning-based intrusion detection system for rpl in internet of medical thing networks. *IEEE Access* (2025).
87. Alalhareth, M. & Hong, S. Enhancing the internet of medical things (iomt) security with meta-learning: A performance-driven approach for ensemble intrusion detection systems. *Sensors* **24**, 3519 (2024).
88. Sun, Z., An, G., Yang, Y. & Liu, Y. Optimized machine learning enabled intrusion detection system for internet of medical things. *Health Information Science and Systems* (2024).
89. Saif, S., Das, P., Biswas, S. & Khari, M. Hiids: Hybrid intelligent intrusion detection system empowered with machine learning and metaheuristic algorithms for application in iot based healthcare. *Computer Commun.* **195**, 136–145 (2022).
90. Imam, A., Usman, M. & Chiawa, M. On consistency and limitation of paired t-test, sign and wilcoxon sign rank test. *IOSR J. Math.* **10**, 01–06. https://doi.org/10.9790/5728-10140106 (2014).
91. Rietveld, T. & van Hout, R. The paired t test and beyond: Recommendations for testing the central tendencies of two paired samples in research on speech, language and hearing pathology. *J. Commun. Disorders* **69**, 44–57. https://doi.org/10.1016/j.jcomdis.2017.07.002 (2015).
92. Smucker, M. D., Allan, J. & Carterette, B. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management (CIKM)*, 623–632, https://doi.org/10.1145/1321440.1321528 (2007).
93. Daher, L. A. Towards secure iomt: Attack detection using deep q-learning in healthcare networks. In *2023 16th International Conference on Developments in eSystems Engineering (DeSE)*, 407–412 (IEEE, 2023).
94. Rookard, C. & Khojandi, A. Applying deep reinforcement learning for detection of internet-of-things cyber attacks. In *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, 0389–0395 (IEEE, 2023).

Author contributions

Conceptualization, H.S, M.N. and A.R.; methodology, H.S. and M.N.; software, A.A., F.U., Y.Z; validation, A.R, F.A., G.A. ; formal analysis, H.S., H.N. Y.Z; investigation, H.S., M.N; resources, A.A., F.A, G.A, F.U.; data curation, H.S, F.U.; writing original draft preparation, H.S., M.N., A.R; visualization, H.S., F.U, Y.Z, H.N. All authors have read and agreed to the published version of the manuscript.

Funding

No funding was received for this work.

Declarations

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to Y.Z.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.